

Simulador para Entrenamiento en Gestión de Proyectos Basado en Inteligencia Artificial

Profesora: Dra. Marcela Samela

Alumno: Juan Pablo Sasso

Trabajo Final de Carrera presentado para obtener el título de

Lic. en Gestión de Tecnología Informática

Marzo 2026

Resumen

La formación en gestión de proyectos presenta una brecha crítica entre conocimientos teóricos y habilidades prácticas aplicables en contextos profesionales, las propuestas educativas actuales se centran en la transmisión de marcos metodológicos como PMBOK, Scrum o SAFe, pero sin ofrecer instancias realistas donde los estudiantes puedan ejercitar la toma de decisiones bajo presión, la gestión de conflictos interpersonales o la negociación con múltiples interesados.

Este trabajo propone el diseño arquitectónico completo de un simulador interactivo basado en inteligencia artificial que recrea dinámicas auténticas de gestión de proyectos tecnológicos. El sistema genera escenarios complejos que incluyen desafíos técnicos imprevistos, conflictos de equipo, cambios de alcance y presiones de clientes, obligando al usuario a tomar decisiones estratégicas cuyas consecuencias se reflejan en las variables del proyecto (tiempo, presupuesto, moral del equipo, satisfacción del cliente). La retroalimentación automática y personalizada, generada mediante procesamiento de lenguaje natural y algoritmos de aprendizaje por refuerzo, permite desarrollar competencias críticas en un entorno seguro donde el error es formativo.

El documento especifica el modelo dinámico central del simulador (la representación del conocimiento, la generación de eventos ponderados y la propagación de consecuencias), la arquitectura del agente de aprendizaje (el modelo bayesiano de usuario, la adaptación pedagógica y la evaluación multidimensional), los mecanismos de interacción mediante lenguaje natural y los lineamientos de implementación técnica. Aunque no incluye desarrollo de software funcional, establece las bases conceptuales completas para su futura materialización, aportando un recurso innovador que transforma la enseñanza tradicional de gestión de proyectos en una experiencia inmersiva y adaptativa.

Palabras clave: aprendizaje experiencial, gestión de proyectos, inteligencia artificial, simulación, tutorización inteligente

Abstract

Training in project management shows a critical gap between theoretical knowledge and practical skills applicable in professional contexts. Current educational approaches focus on transmitting methodological frameworks such as PMBOK, Scrum, or SAFe, but they fail to provide realistic instances where students can practice decision-making under pressure, interpersonal conflict management, or negotiation with multiple stakeholders.

This work proposes the complete architectural design of an AI-based interactive simulator that recreates authentic dynamics of technological project management. The system generates complex scenarios that include unforeseen technical challenges, team conflicts, scope changes, and client pressure, forcing the user to make strategic decisions whose consequences are reflected in project variables (time, budget, team morale, client satisfaction). Automatic and personalized feedback (generated through natural language processing and reinforcement learning algorithms) enables the development of critical competencies in a safe environment where making mistakes is part of the learning process.

The document specifies the simulator's core dynamic model (knowledge representation, weighted event generation, and consequence propagation), the architecture of the learning agent (the user's Bayesian model, pedagogical adaptation, and multidimensional evaluation), the interaction mechanisms via natural language, and the technical implementation guidelines. Although it does not include functional software development, it establishes the complete conceptual foundations for its future materialization, providing an innovative resource that transforms traditional project-management education into an immersive and adaptive experience.

Keywords: artificial intelligence, experiential learning, intelligent tutoring, project management, simulation

Dedicatoria

A mis padres, por su apoyo impulsándome siempre a dedicarme a este proyecto que es mi carrera en sistemas, con esfuerzo y constancia.

A Marianella Cabrera Ahumada, por acompañarme desde el primer día, aportando su conocimiento, su visión y su paciencia en cada etapa del camino. Este trabajo no habría sido el mismo sin su apoyo constante y su compromiso genuino.

A mis amigos y colegas, por su compañía, comprensión y estímulo durante este proceso tan importante para mí.

Reconocimientos

A la Dra. Marcela Rosalba Samela, profesora del Trabajo Final de Carrera, por su guía, compromiso y paciencia a lo largo de todo el proceso de realización de este trabajo final de carrera. Su orientación académica y humana fue fundamental para alcanzar los objetivos propuestos.

A los docentes y autoridades de la Universidad Abierta Interamericana, por su dedicación y por brindar las herramientas necesarias para mi formación profesional.

A todos los que, de una u otra manera, colaboraron con su apoyo, sus ideas o su tiempo para hacer posible este trabajo.

A todos ellos, nuevamente, ¡muchas gracias!

Índice

Resumen 1

Abstract 2

Dedicatoria 3

Reconocimientos 4

Capítulo 1 - Introducción 14

1.1 Problemas y Soluciones 14

1.2 Nuestra Propuesta 15

1.3 Marco Institucional 16

1.4 Objetivos del Trabajo Final 17

1.4.1 **Objetivo general** 17

1.4.2 **Objetivos específicos** 17

1.5 Contribuciones Principales 17

1.6 Estructura General del Trabajo Final de Carrera 18

Capítulo 2 - Estado del Arte 21

2.1 El Aprendizaje Experiencial y las Metodologías Activas en la Educación Superior 21

2.2 La Simulación como Instrumento Didáctico 22

2.2.1 ***Plataformas y Ecosistemas Educativos Digitales*** 24

2.2.2 ***Sobre la Integración de Simulaciones en la Educación*** 25

2.3 Integración de la Inteligencia Artificial en la Educación 26

2.4 La Inteligencia Artificial en el Aprendizaje por Proyectos y Desafíos 29

2.5 Simuladores para la Gestión de Proyectos 31

2.5.1 ***Integración de Inteligencia Artificial*** 32

2.6 Justificación de la Propuesta 33

Capítulo 3 – Marco Teórico 35

3.1 Tecnologías de la Información y Comunicación en la Educación 35

3.1.1 ***Evolución de las TIC en Contextos Educativos*** 35

3.1.2 ***Metodologías Activas Mediadas por TIC*** 36

3.1.3 Evaluación Digital de Competencias 39

3.2 Inteligencia Artificial 40

3.2.1 Fundamentos de la Inteligencia Artificial 40

3.2.2 Aprendizaje por Refuerzo 44

3.2.3 Aprendizaje Automático, Redes Neuronales y Aprendizaje Profundo 45

3.2.4 Agentes Inteligentes y Sistemas Expertos 48

3.2.5 La IA Aplicada a la Educación 49

3.2.6 Consideraciones Éticas, Desafíos y Limitaciones 55

3.3 Procesamiento de Lenguaje Natural 56

3.3.1 La Evolución del PLN 57

3.3.2 Fundamentos Matemáticos que Sustentan el PLN 58

3.3.3 El Desafío del Lenguaje Humano 61

3.3.4 El Papel del Aprendizaje Automático y el Aprendizaje Profundo 63

3.3.5 Modelos de Lenguaje de Gran Tamaño 64

3.3.6 Arquitectura Transformer y Mecanismos de Atención 65

3.3.7 Aportes de BERT, GPT, T5 y BART al PLN Moderno 67

3.3.8 Estructura del Subsistema de Comunicación 69

3.3.9 El Futuro de los Modelos Multimodales 70

3.3.10 Cómo Contribuyen los PLN en la Educación 71

3.4 Simulaciones 72

3.4.1 Modelos de Simulación 72

3.4.2 Simulaciones Educativas 74

3.4.3 Las Simulaciones y las Teorías del Aprendizaje Experiencial 76

3.4.4 Simulaciones Tradicionales y Simulaciones con IA Incorporada 77

3.4.5 Cómo la IA Mejora la Calidad de una Simulación 79

3.4.6 Técnicas de IA en la Simulación Educativa 81

3.4.7 El Futuro de las Simulaciones Educativas 82

3.4.8 Evaluación de la Efectividad del Aprendizaje en Simulaciones Educativas 83

3.4.9 Desafíos y Riesgos de las Simulaciones con IA 86

3.5 Gestión de Proyectos de Software y Liderazgo de Equipos 89

3.5.1 Los Grupos de Procesos del PMBOK 90

3.5.2 Competencias Críticas en Gestión de Proyectos 92

3.5.3 La Gestión de Proyectos de Software como Contexto Específico 93

3.5.4 Desafíos en la Formación de Gestores de Proyectos 95

3.6 Síntesis e Integración Conceptual 96

3.6.1 Convergencia de Tecnologías Educativas 96

3.6.2 Simulación Inteligente para Gestión de Proyectos 97

3.6.3 Marco Conceptual Integrado 98

3.6.4 Delimitación del Sistema Propuesto 99

Capítulo 4 – Diseño de Simulador. Fase I: Definición y Adquisición del Conocimiento 103

4.1 Definición del Propósito y Objetivos 103

4.2 Adquisición y Análisis Cognitivo del Experto 105

4.3 Documentación del Diseño 106

4.3.1 Biblia de la Historia 107

4.3.2 Guion Gráfico 107

4.3.3 Documento de Diseño Técnico 108

4.4 Evaluación Tecnológica Preliminar 109

Capítulo 5 – Diseño de Simulador. Fase II: Diseño de la Arquitectura y Representación 112

5.1 Definición de la Arquitectura de Sistemas 112

5.2 Arquitectura General del Sistema 112

5.3 Modelo de Agente Inteligente 114

5.4 Diseño del Modelo Dinámico Central 116

5.4.1 Finalidad Pedagógica del Modelo 120

5.5 Ontología y Representación de Conocimiento 120

5.5.1 Tipos de Conocimiento Representado 121

5.5.2 Ontología del Simulador 122

5.5.3 Representación del Conocimiento y Razonamiento Contextual 123

5.5.4 Finalidad y Valor Educativo 125

5.6 Tipos de Contenido y Flujo de Información 125

5.6.1 Contenido Lineal 125

5.6.2 Contenido Cíclico 126

5.6.3 Contenido Abierto 126

5.6.4 Flujo de Información Entre Módulos 126

5.7 Alcance Funcional del Sistema Propuesto 127

Capítulo 6 – Diseño de Simulador. Fase III: Modelado Lógico, Dinámica de Eventos y Procesamiento de Lenguaje Natural 131

6.1 Formulación del Problema Simbólico 131

6.1.1 Formalización del Sistema 131

6.2 Operadores y Encadenamiento de Eventos 137

6.2.1 **Los Operadores como Acciones Simbólicas** 138

6.2.2 Encadenamiento de Eventos 138

6.3 Generación de Eventos Ponderados 139

6.3.1 El Método Monte Carlo 140

6.3.2 Mecanismo de Generación Pseudoaleatoria Guiada 142

6.3.3 Incorporación de Lógica Difusa 144

6.3.4 Integración con el Agente Planificador 145

6.4 Entrada de Lenguaje Natural 145

6.4.1 Restricción de Dominio 146

6.4.2 Proceso de Interpretación 146

6.4.3 Tipificación de Perceptos y Reducción de Ambigüedad 147

6.5 Procesamiento Semántico y Sintáctico 148

6.5.1 Capas de Procesamiento 148

6.5.2 Integración con el Modelo de Conocimiento 149

6.5.3 Arquitectura y Relación con Sistemas Expertos 150

Capítulo 7 – Diseño de Simulador. Fase IV: Diseño del Sistema de Inteligencia Artificial 151

7.1 Arquitectura del Agente de Aprendizaje 151

7.1.1 Estructura Dual: Elementos de Desempeño y Aprendizaje 151

7.1.2 Ciclo Operativo del Agente 153

7.2 Función Objetivo del Sistema 156

7.2.1 Naturaleza Multidimensional de la Función Objetivo 156

7.2.2 Descomposición de la Función Objetivo. 156

7.3 Modelo de Usuario 159

7.3.1 Componentes del Modelo 159

7.3.2 Técnicas de Modelado 161

7.3.3 Funcionamiento del Sistema de Inferencia Bayesiana 163

7.4 Experiencia de Entrenamiento 168

7.5 Detección de Patrones 169

7.5.1 Patrones Predictivos y Anticipación 169

7.5.2 Minería de Datos Educacionales 171

7.5.3 Clustering, Clasificación e Identificación de Arquetipos de Usuario 172

7.6 Módulo de Adaptación y Tutoría 173

7.6.1 Arquitectura del Módulo Tutor 173

7.6.2 Tipos de Intervenciones Tutoriales 174

7.6.3 Estrategias Pedagógicas 176

7.6.4 Reglas de Acción ante Respuestas del Usuario 177

7.6.5 Personalización de la Tutoría 178

7.6.6 Proactividad y Comportamiento Social 178

7.6.7 Resumen y Síntesis de Sesión 179

7.7 Algoritmos de Aprendizaje por Refuerzo 179

7.7.1 Formalización como Proceso de Decisión de Markov 179

7.7.2 Algoritmo Actor-Critic 180

7.7.3 Balance Exploración-Explotación 181

7.7.4 Aprendizaje por Currículo 182

7.7.5 Aprendizaje por Transferencia 182

7.7.6 Adaptación a la No Estacionariedad 182

7.8 Integración Sistémica y Consideraciones Finales 183

7.8.1 Arquitectura de Integración Funcional 183

7.8.2 Propiedades Emergentes de la Integración 186

Capítulo 8 – Diseño de Simulador. Fase V: Consideraciones Teóricas de Despliegue y Validación 189

8.1 Diseño de la Metodología de Validación 189

8.1.1 **Validación en Dos Fases** 189

8.1.2 Validación del Modelo de Usuario 190

8.1.3 Validación del Componente PLN 190

8.1.4 Métricas Cuantitativas de Éxito 190

8.2 Arquitectura de Despliegue 191

8.2.1 Arquitectura Cliente-Servidor Estratificada 191

8.2.2 Escalabilidad Horizontal mediante Replicación 193

8.2.3 Integración con Ecosistemas Educativos Institucionales 194

8.3 Especificación de Plataforma de Procesamiento 195

8.3.1 Disparidad entre Procesamiento Tradicional y Procesamiento de IA 195

8.3.2 Requisitos de Almacenamiento y Base de Datos 196

8.3.3 Latencia, Tasa de Procesamiento y Experiencia del Usuario 197

8.3.4 Estimación de Capacidad para Diferentes Volúmenes 198

8.3.5 Gestión de Riesgos y Contingencia 199

Capítulo 9 – Plan de Implementación 201

9.1 Recursos Humanos 201

9.2 Presupuesto 203

9.2.1 Costo Total Estimado del Equipo Humano 203

9.2.2 Recursos Tecnológicos y Costos de Infraestructura 204

9.3 Cronograma Tentativo 208

9.4 Plan de Control y Seguimiento del Proyecto 209

Factores Externos Condicionantes 211

Conclusiones 213

Líneas futuras de investigación 214

Limitaciones del Trabajo Actual 214

Simulaciones Colaborativas y Modelado de Conocimiento Distribuido 214

Modelado Emocional y Motivacional del Estudiante 215

Reutilización de Componentes en Otros Dominios Educativos 215

Transparencia, Explicabilidad y Consideraciones Éticas 216

Acrónimos 217

Referencias 220

Índice de figuras

Figura 1: Diagrama de bloques de aprendizaje con un tutor 43

Figura 2: Arquitectura de un Transformer 67

Figura 3: Representación en forma de Árbol de Decisiones del Espacio de Estados del Simulador 136

Figura 4: Ciclo operativo del agente del simulador 155

Figura 5: Diagrama de Flujo del Proceso de Inferencia de Necesidad de Asistencia 165

Figura 6: Diagrama de Flujo del Proceso de Actualización Bayesiana de Creencias 166

Figura 7: Ciclo Completo de retroalimentación del Sistema de Aprendizaje Adaptativo 167

Figura 8: Diagrama integrador completo 187

Figura 9: Diagrama de Gantt Tentativo 209

Índice de tablas

Tabla 1: Métricas utilizadas para medir el éxito del sistema 191
Tabla 2: Distribución aproximada del presupuesto 203
Tabla 3: Costo estimado de los recursos de Hardware y Licencias de Software 205
Tabla 4: Costos estimados de los servicios de la nube a contratar 206
Tabla 5: Costos estimados de los servicios complementarios 207
Tabla 6: Costos totales estimados de implementación del sistema 208

Capítulo 1 - Introducción

1.1 Problemas y Soluciones

En la formación en gestión de proyectos, ya sea en cursos cortos, capacitaciones laborales, diplomaturas, tecnicaturas o carreras universitarias, se observa una brecha crítica entre contenidos teóricos y habilidades prácticas requeridas en contextos reales. A pesar de la difusión de marcos metodológicos como PMBOK, Scrum o SAFe, las instancias formativas ofrecen principalmente conocimiento teórico sin oportunidades significativas de aplicación en escenarios dinámicos, bajo presión o con incertidumbre.

Los egresados poseen sólidos fundamentos teóricos pero carecen de experiencia en enfrentar cambios imprevistos de requerimientos, conflictos interpersonales, fallas técnicas críticas o presiones de las partes interesadas. Esta carencia limita su capacidad para tomar decisiones bajo presión y ejercer un liderazgo efectivo de equipos. Desde la perspectiva docente, resulta difícil evaluar competencias estratégicas mediante instancias tradicionales como exámenes o trabajos escritos.

Actualmente, la única forma de adquirir experiencia práctica es participar directamente en proyectos reales, frecuentemente en contextos productivos de alta exigencia. Esta situación representa un riesgo para las organizaciones, que deben asumir el costo de errores críticos, desviaciones de alcance y pérdida potencial de clientes. Las empresas no siempre están dispuestas a este costo de aprendizaje, generando una paradoja: se requiere experiencia para gestionar, pero no existen espacios controlados donde adquirirla sin consecuencias reales.

Frente a este diagnóstico, se propone el diseño de un simulador que permita recrear dinámicas típicas de proyectos reales dentro de un entorno seguro y controlado, integrando eventos aleatorios, restricciones de tiempo, conflictos internos y variables externas que obliguen al usuario a tomar decisiones estratégicas. A través de retroalimentación automática basada en inteligencia artificial, la herramienta facilitará el aprendizaje activo y promoverá el desarrollo de habilidades críticas como la priorización, la negociación, la resolución de problemas y la comunicación efectiva.

1.2 Nuestra Propuesta

La propuesta consiste en el diseño arquitectónico integral de un simulador interactivo orientado al entrenamiento de habilidades en gestión de proyectos. Se trata de una especificación conceptual y estructural de alto nivel, cuyo propósito es definir los componentes funcionales, pedagógicos y tecnológicos necesarios para construir una herramienta capaz de emular situaciones reales enfrentadas por un gerente de proyectos. El sistema permitirá experimentar la dinámica completa de un proyecto, desde su planificación inicial hasta su cierre, enfrentando imprevistos, restricciones y decisiones críticas.

El simulador se estructura mediante escenarios predefinidos y generados dinámicamente, que incluyen eventos típicos como cambios en el alcance, demoras en entregables, conflictos de equipo, bloqueos técnicos, limitaciones presupuestarias y presiones de interesados; cada situación exige una respuesta estratégica del usuario, evaluada mediante criterios definidos en el modelo dinámico central. La retroalimentación, inmediata o diferida según el modo de simulación, se expresa cuantitativamente, mediante puntos e indicadores, y cualitativamente, con recomendaciones y análisis de impacto.

Este documento no contempla el desarrollo técnico del software, sino la especificación funcional, arquitectónica y pedagógica completa que permitirá su futura implementación, e incluye al estructura general del simulador, la lógica de interacción y tipología de eventos, los mecanismos de evaluación mediante inteligencia artificial, cálculo de métricas de desempeño y arquitectura del agente de aprendizaje, integración de procesamiento de lenguaje natural para interacciones conversacionales, y un diseño pedagógico con progresión de dificultad y personalización de escenarios. Dada la complejidad técnica de algunos de los módulos propuestos, como el modelado probabilístico del usuario, los algoritmos de aprendizaje por refuerzo o la generación dinámica de contenido conversacional, su implementación práctica requerirá un desarrollo progresivo por etapas. En un escenario realista, el sistema debe comenzar con un prototipo funcional mínimo (MVP) basado en reglas determinísticas y evaluación estructurada, incorporando gradualmente mecanismos avanzados de adaptación y aprendizaje automático conforme se disponga de validación empírica, recursos técnicos y presupuesto suficiente.

El diseño se apoya en enfoques modernos de simulación, modelado de comportamiento y evaluación adaptativa, tomando como referencia arquitecturas de sistemas multi-agente, motores de reglas basados en lógica difusa y algoritmos de aprendizaje por refuerzo. Para la

futura implementación técnica, se recomienda el uso de marcos de trabajo web modernos, como React para interfaz y Python para backend, bases de datos híbridas con PostgreSQL para estructuras formales y MongoDB para eventos dinámicos, y modelos de lenguaje natural de OpenAI o Anthropic para simular interacciones con interesados del proyecto.

El objetivo pedagógico trasciende la respuesta correcta, se busca desarrollar el razonamiento estratégico propio de un líder real, considerando simultáneamente las dimensiones técnicas, humanas y organizacionales de la gestión. Esta propuesta busca combinar los beneficios de la simulación con el potencial de la inteligencia artificial para construir una experiencia de aprendizaje significativa, inmersiva y replicable, que contribuya al desarrollo de competencias prácticas en gestión de proyectos.

1.3 Marco Institucional

Este trabajo está destinado principalmente a instituciones de educación superior que imparten formación en gestión de proyectos tecnológicos, tales como universidades, institutos técnicos y centros de capacitación profesional, que enfrentan la dificultad de ofrecer experiencias prácticas realistas sin los riesgos asociados a proyectos reales en entornos productivos.

El simulador propuesto responde a la necesidad de estas instituciones de complementar la formación teórica en marcos metodológicos (PMBOK, Scrum, SAFe) con instancias de entrenamiento controlado que permitan a los estudiantes desarrollar competencias estratégicas y de liderazgo antes de su inserción laboral. Particularmente, se orienta a programas académicos de carreras de grado en ingeniería de software, tecnología informática y administración de empresas, así como a tecnicaturas, diplomaturas y cursos de actualización profesional en gestión de proyectos.

El sistema se integra como recurso didáctico complementario en arquitecturas pedagógicas existentes, mediante estándares de interoperabilidad, permitiendo su adopción gradual en diferentes niveles formativos sin requerir modificaciones estructurales de los planes de estudio vigentes. De este modo, las instituciones pueden incorporar experiencias prácticas documentables y evaluables sin necesidad de establecer convenios empresariales, gestionar proyectos reales con clientes externos o exponer a estudiantes novatos a contextos de alto riesgo operativo.

1.4 Objetivos del Trabajo Final

1.4.1 Objetivo general

Diseñar un simulador interactivo basado en inteligencia artificial que permita entrenar habilidades prácticas en gestión de proyectos, mediante escenarios realistas que simulen dinámicas propias de entornos educativos.

1.4.2 Objetivos específicos

- Diseñar un sistema de ponderación y retroalimentación que permita evaluar las decisiones del usuario en función de competencias clave en gestión de proyectos.
- Estructurar una arquitectura conceptual del simulador que garantice escalabilidad, flexibilidad y adaptabilidad a distintos escenarios educativos.
- Proponer lineamientos de integración curricular que permitan aplicar el simulador como herramienta complementaria en programas de formación en gestión de proyectos.

1.5 Contribuciones Principales

Esta propuesta plantea un enfoque alternativo para la formación de gestores de proyectos, desplazando el eje desde la memorización de marcos metodológicos hacia el desarrollo de capacidades estratégicas mediante experiencia controlada. La contribución fundamental radica en abordar parcialmente la brecha existente entre la formación académica y la empleabilidad, proporcionando un espacio de entrenamiento que actualmente solo existe mediante exposición directa en contextos laborales de alto riesgo. Para las instituciones educativas, el simulador representa una herramienta diferenciadora dentro de la oferta educativa actual, que en muchos casos continúa centrada principalmente en contenidos teóricos; Esto permite ofrecer experiencias prácticas documentables y evaluables sin requerir convenios empresariales específicos ni infraestructura física costosa. Asimismo, evita exponer a los estudiantes a riesgos asociados a entornos productivos reales; la herramienta se integra naturalmente en arquitecturas curriculares existentes, complementando asignaturas tradicionales con ejercitación práctica que puede incorporarse gradualmente según recursos disponibles.

Para los estudiantes, constituye un entorno de aprendizaje sin consecuencias donde el error es formativo, sino que constituye una instancia de aprendizaje, a diferencia de casos escritos o juegos de rol presenciales limitados en tiempo, el simulador permite iteración

ilimitada, experimentación con estrategias alternativas y retroalimentación inmediata sobre decisiones complejas. Desarrolla competencias críticas difíciles de evaluar mediante exámenes tradicionales: gestión bajo presión, negociación con múltiples interesados, priorización de recursos escasos y adaptación ante imprevistos. Los egresados pueden acceder al mercado laboral con experiencia demostrable en gestión de crisis, lo que puede contribuir a reducir su curva de aprendizaje inicial.

Para el cuerpo docente, la herramienta funciona como un recurso pedagógico complementario que libera tiempo de tareas repetitivas, como corrección manual, diseño de casos o seguimiento individual, para concentrarse en mentoría estratégica de alto valor; ya que puede facilitar la observación de patrones de razonamiento que suelen resultar difíciles de identificar en evaluaciones escritas tradicionales, identificar fortalezas y debilidades específicas mediante analítica automática, y diseñar intervenciones pedagógicas precisas. El sistema no reemplaza al instructor sino que transforma su rol: de transmisor de contenidos a diseñador de experiencias y facilitador de reflexión metacognitiva.

Finalmente, desde una perspectiva sistémica, la propuesta busca contribuir a ampliar el acceso a formación de calidad en gestión, reduciendo la dependencia de redes de contactos personales o ubicación geográfica privilegiada para acceder a experiencias prácticas. El modelo es escalable, replicable y adaptable a diferentes contextos culturales y metodológicos, permitiendo que instituciones con recursos limitados puedan ofrecer experiencias formativas más cercanas a las disponibles en programas académicos con mayor infraestructura.

1.6 Estructura General del Trabajo Final de Carrera

La estructura general de este trabajo está compuesta por tres capítulos principales y cuatro secciones complementarias que desarrollan los aspectos principales del trabajo.

Capítulo 1 – Introducción: Presenta el problema profesional abordado, los objetivos del trabajo, el marco institucional que lo contextualiza, la propuesta general planteada y las principales contribuciones esperadas.

Capítulo 2 – Estado del Arte: Se revisan los principales aportes sobre la aplicación de la inteligencia artificial en educación, destacando su impacto en la personalización del aprendizaje, en metodologías activas y en la gestión institucional.

Capítulo 3 – Marco Teórico: Fundamenta conceptualmente la propuesta a partir de teorías sobre aprendizaje experiencial, simulación educativa, inteligencia artificial en contextos formativos y evaluación adaptativa.

Capítulo 4 – Definición y Adquisición del Conocimiento: Describe el proceso de identificación del dominio, objetivos de aprendizaje y tareas del simulador. Incluye la adquisición del conocimiento experto, el análisis cognitivo de tareas y la definición de escenarios base que guiarán el modelado pedagógico y la interacción del sistema.

Capítulo 5 – Diseño de la Arquitectura y Representación: Desarrolla la estructura general del simulador, definiendo los módulos, flujos de información y el método de representación del conocimiento. Establece la arquitectura general y la comunicación entre los componentes que articularán la simulación, la pedagogía y la inteligencia artificial.

Capítulo 6 – Modelado Lógico, Dinámica de Eventos y Procesamiento de Lenguaje Natural: Aborda la construcción del motor lógico del simulador. Describe cómo se generan, ponderan y procesan los eventos pseudoaleatorios, y cómo se integra el procesamiento del lenguaje natural para interpretar y responder al usuario dentro de entornos dinámicos y realistas.

Capítulo 7 – Diseño del Sistema de Inteligencia Artificial: Expone el diseño del componente inteligente, basado en aprendizaje automático y por refuerzo. Define cómo el simulador analiza decisiones, adapta su dificultad y retroalimentación, y modela el comportamiento del usuario mediante técnicas bayesianas y redes neuronales.

Capítulo 8 – Consideraciones Teóricas de Despliegue y Validación: Presenta las consideraciones teóricas y prácticas para la validación del simulador. Incluye los métodos de evaluación, métricas de rendimiento, aspectos éticos, pedagógicos y de accesibilidad, así como lineamientos para su implementación y mantenimiento futuro.

Capítulo 9 - Plan de Implementación: Detalla la implementación del simulador inteligente, definiendo el equipo, presupuesto, recursos tecnológicos, cronograma y mecanismos de control necesarios para garantizar un desarrollo ordenado, eficiente y coherente con los objetivos del proyecto.

Factores Externos Condicionantes: Identifica variables contextuales, institucionales o tecnológicas que podrían limitar o modificar el alcance del simulador propuesto.

Conclusiones: Resume los hallazgos principales y reflexiona sobre el aporte del trabajo dentro del campo profesional.

Líneas Futuras de Investigación: Propone caminos para extender, complementar o mejorar la propuesta presentada, considerando sus limitaciones y nuevas oportunidades.

Capítulo 2 - Estado del Arte

En las últimas décadas, la inteligencia artificial (IA) ha adquirido una creciente relevancia en el ámbito educativo, posicionándose como una herramienta con potencial para transformar procesos de enseñanza, gestión institucional y aprendizaje personalizado. Esta incorporación no ha sido homogénea ni carente de tensiones, pero múltiples investigaciones recientes coinciden en destacar su potencial de generar cambios significativos cuando se articula con enfoques pedagógicos innovadores. En este capítulo se presenta una revisión de antecedentes significativos que permiten enmarcar conceptual y metodológicamente la propuesta desarrollada en este trabajo. Se destacan especialmente aquellos aportes que vinculan la IA con entornos virtuales de aprendizaje, metodologías basadas en proyectos y dispositivos institucionales orientados a la mejora educativa. La educación superior contemporánea se encuentra en un proceso de transformación continua, impulsada por la necesidad de adaptar los métodos de enseñanza-aprendizaje a las demandas de un mundo globalizado, digital y en constante evolución. Este cambio de paradigma busca ir más allá de la mera transmisión de conocimientos teóricos, priorizando el desarrollo de habilidades, competencias y actitudes que preparen a los estudiantes para los desafíos profesionales y sociales del siglo XXI. En este contexto, el aprendizaje experiencial y las metodologías activas han adquirido un rol cada vez más relevante.

2.1 El Aprendizaje Experiencial y las Metodologías Activas en la Educación Superior

Los métodos tradicionales de enseñanza, que históricamente se han basado en la repetición y memorización de conceptos, están siendo progresivamente complementados por enfoques de aprendizaje experiencial, el cual se centra en habilidades de pensamiento crítico, como el análisis y resolución de problemas y la creatividad (Armstrong, 2010; Maksimenko et al., 2023). Este cambio de paradigma significa que el alumno asume un rol más activo en su proceso de aprendizaje, en un entorno educativo diseñado para fomentar la participación activa y el desarrollo autónomo. Diversas investigaciones, basadas en los aportes de Dewey (1938) y Kolb (2015), muestran la relación entre la experiencia, el saber, el trabajo e incluso crecer como persona, junto con la importancia de la experiencia en la educación. Permitir a los estudiantes "aprender haciendo y siendo", impulsa el desarrollo de habilidades esenciales para el siglo XXI, como la comunicación asertiva, el liderazgo, la planificación, el trabajo en equipo, la reflexión, y el pensamiento crítico y analítico. Esto prepara a los alumnos para los complejos desafíos del ámbito laboral. "La combinación de conocimientos prácticos y teóricos prepara a los

estudiantes para los desafíos futuros en los campos del desarrollo, la innovación y la colaboración" (Jiménez García y Velasco Quintana, 2024, pág. 36). Actividades que recrean situaciones profesionales cercanas a la vida real, elevan la motivación estudiantil y la relevancia de su aprendizaje.

Las metodologías activas son procesos de enseñanza promovidos por el Espacio Europeo de Educación Superior (EEES), que buscan una participación activa del alumno, disminuyendo la importancia de la clase magistral y concediendo mayor valor al trabajo independiente y al uso de tecnologías de la información y comunicación. Su meta es no solo cultivar conocimientos teóricos, sino también habilidades y competencias solicitadas por la sociedad global e interconectada (Crespo et al., 2024). En la carrera de Psicología, se han puesto en marcha actividades, por ejemplo, empleando el "eye-tracking" para estudiar procesos cognitivos; el "juego del bien común" para entender las conductas prosociales, el "experimento de Asch" que sirve para reflexionar sobre la presión del grupo y además, simulaciones de mediación de conflictos. Según los resultados reportados, estas experiencias han mostrado niveles elevados de satisfacción en asistencia, aprendizaje y la conexión entre teoría y práctica (Jiménez García y Velasco Quintana, 2024).

Se investigó cómo integrar la inteligencia artificial en el Aprendizaje Basado en Proyectos en la educación superior, principalmente en las ingenierías. Los autores identificaron un uso inicial y focalizado de la IA, principalmente a través de herramientas tipo chatbot, apoyando tareas docentes, entregando retroalimentación personalizada y respondiendo consultas de alumnos, sobre todo en clases online. Pese a la llegada de herramientas como ChatGPT, la integración de la IA en entornos educativos aún presenta un grado limitado de adopción, lo que evidencia un campo de investigación con amplio potencial de desarrollo en los próximos años (Lavado-Anguera et al., 2024).

2.2 La Simulación como Instrumento Didáctico

La simulación constituye una estrategia didáctica orientada a recrear aspectos relevantes del mundo real mediante escenarios guiados e inmersivos. Este enfoque permite representar situaciones complejas de manera interactiva y puede aplicarse en múltiples disciplinas y contextos educativos. El objetivo es permitir a los estudiantes ejercitar habilidades críticas, tomar decisiones y cometer errores sin enfrentar las consecuencias propias de situaciones reales; algo crucial en materias que demandan una gran fusión de saberes teóricos y prácticos (Blackstock et al, 2013, como se citó en Jiménez García y Velasco Quintana, 2024).

Este enfoque permite desarrollar aprendizajes mediante la práctica activa en un entorno controlado. Esta metodología ha sido ampliamente utilizada en ámbitos como la aviación, el ámbito militar, la energía nuclear y la exploración espacial, así como en diversas especialidades médicas, entre ellas anestesiología, cuidados intensivos, obstetricia y medicina de urgencias; a través de escenarios realistas y equipo especial, facilitando la práctica repetida y el reentrenamiento hasta dominar procesos o habilidades. La práctica deliberada en entornos seguros, acompañada de retroalimentación sistemática, contribuye significativamente al desarrollo de competencias profesionales. La simulación es una metodología pedagógica que ha evolucionado considerablemente, logrando reconstruir situaciones de la vida cotidiana dentro de un ámbito seguro y controlado (Lateef, 2010).

Se han desarrollado simulaciones con altos niveles de fidelidad, lo que incrementa la confiabilidad de las actividades formativas y mejorando el desempeño de los alumnos. La simulación de alta fidelidad, por ejemplo, posibilita un acercamiento a contextos clínicos reales de forma segura y reflexiva, lo cual resulta especialmente relevante para el desarrollo de capacidades prácticas y emocionales en el ámbito médico, donde la falta de práctica puede generar consecuencias significativas. Favorece una comprensión más profunda de los contenidos académicos y su aplicación práctica, preparando mejor al profesional (Jiménez García y Velasco Quintana, 2024). “Experiencia y aprendizaje son conceptos inseparables, lo que abre camino al aprendizaje experiencial en el proceso educativo, persiguiendo la obtención de conocimientos y significados mediante la vivencia de experiencias realistas y su consiguiente reflexión posterior, con efectos permanentes” (Jiménez García y Velasco Quintana, 2024, pág. 148).

El aprendizaje con simulaciones promueve el liderazgo, el pensamiento crítico y la reflexión. "A pesar del acuerdo sobre la importancia del pensamiento crítico, la reflexividad, y la experimentación, aún no hay evidencia concreta de como este tipo de aprendizaje práctico desarrolla liderazgo" (Barbaroux, 2022, p. 2). La simulación no solo ocurre en espacios reales, mucho después de la pandemia COVID-19 de 2019 y 2020, la simulación virtual creció y brindó experiencias inmersivas, prácticas y sin limitaciones. La metodología de simulación, ampliamente utilizada en disciplinas como la medicina y la ingeniería, ha demostrado ser una herramienta efectiva para la formación de competencias prácticas y la mejora de la retención del conocimiento (Lateef, 2010).

Al abordar este simulador, la IA jugará un papel fundamental integrándose para mejorar el aprendizaje, incorporando modelos de lenguaje natural, para que los algoritmos interpreten y generen texto con coherencia, para simular interacciones reales con Interesados, como clientes, desarrolladores o patrocinadores (Cunuhay Cuchipe et al., 2022). Esto dará a los usuarios la oportunidad de practicar habilidades de conversación y negociación en un entorno seguro, mediante herramientas como chatbots, duplicando roles símil docentes, dando retroalimentación individual y gestionando consultas de estudiantes. La IA puede examinar grandes cantidades de datos sobre el comportamiento y las cualidades de los estudiantes, adaptando el contenido y los métodos de enseñanza a sus necesidades únicas. Dicha capacidad de la IA permite personalizar el aprendizaje y el crecimiento de habilidades, para que los estudiantes se preparen para desafíos complicados. Juntas, estas aplicaciones apuntan a impulsar el crecimiento de destrezas prácticas, en una forma inmersiva y replicable (Cunuhay Cuchipe et al., 2022).

2.2.1 Plataformas y Ecosistemas Educativos Digitales

Un Sistema de Gestión del Aprendizaje (SGA) es una plataforma de software que orquesta la administración y distribución digital de cursos, Su finalidad es apoyar los procesos de enseñanza, aprendizaje, comunicación y gestión del conocimiento, así como facilitar el seguimiento del proceso formativo de los estudiantes. Dichos sistemas constituyen la columna vertebral del aprendizaje electrónico (Cataldi y Lage, 2009). Múltiples SGA incorporan funciones interactivas para optimizar la calidad de la enseñanza con retroalimentación y tutoría personalizadas (Sánchez Rodríguez, 2005). Los Sistemas Integrados de Gestión Académica ofrecen una amplia variedad de herramientas como Entornos Virtuales de Aprendizaje (EVA), Sistemas de Gestión de Cursos (SGC), Entornos de Aprendizaje Gestionados (EAG) y Sistemas Integrados de Aprendizaje (SIA). La noción de "aula virtual" o "campus virtual" abarca estas posibilidades, con el establecimiento de un entorno enseñanza-aprendizaje mediado por tecnologías digitales. El análisis de estas plataformas suele centrarse en la calidad del entorno virtual, sus funcionalidades y su escala de implementación, que puede variar desde el soporte a un curso específico hasta el funcionamiento de toda una institución educativa (Adell y Girbert, 1997).

La analítica del aprendizaje emplea técnicas de IA con el objetivo preciso de medir y examinar el propio proceso y las actividades inherentes al aprendizaje, obteniendo datos relevantes sobre la experiencia del alumno, incluyendo, entre otros, el tiempo dedicado al

estudio y el desempeño en las pruebas (León y Viña Brito, 2017). Dichos análisis permiten desarrollar modelos de aprendizaje personalizados y estimar el rendimiento académico, facilitando la toma de decisiones pedagógicas y el diseño de intervenciones educativas. De esta manera, la IA posibilita un análisis más detallado de los datos del estudiante, permitiendo comprender mejor su perfil, aptitudes, requerimientos e intereses; esto a su vez facilita a los sistemas de aprendizaje en línea, la sugerencia de trayectorias educativas personalizadas, o la conformación de equipos de trabajo con perfiles complementarios, todo pensando en el aprendizaje colaborativo (Park Woolf, 2009). La recopilación y posterior análisis de datos, provenientes de las interacciones de los estudiantes en plataformas digitales de aprendizaje, como los sistemas de gestión del aprendizaje (*Learning Management Systems*, por sus siglas en inglés, LMS) o los sistemas adaptativos, resulta fundamental para identificar tendencias y adaptar las prácticas de enseñanza a las necesidades de cada individuo (Jara y Ochoa, 2020). Esta cuestión adquiere particular significancia en el ámbito de la Minería de Datos Educacional, donde convergen los sistemas tutores inteligentes y las técnicas de aprendizaje automático; esto con el objetivo de desentrañar patrones ocultos en las bases de datos educativas (Arana, 2021).

2.2.2 Sobre la Integración de Simulaciones en la Educación

Diversos estudios recientes han analizado el uso de simulaciones educativas en distintos contextos formativos, evidenciando su impacto positivo en el desarrollo de competencias prácticas y cognitivas. En el ámbito de la Fisioterapia, un estudio evaluó la implementación de una simulación compleja en estudiantes de primer año de la asignatura Terapia Manual Básica. Los resultados mostraron una mejora significativa en las habilidades comunicativas, el razonamiento clínico y la toma de decisiones vinculadas al cuidado del paciente. Los participantes valoraron especialmente el primer contacto con una situación clínica simulada y la posibilidad de aplicar los conocimientos adquiridos en un entorno que percibieron como realista. Asimismo, el retroalimentación inmediato y las instancias de discusión posteriores a la simulación fueron identificadas como elementos clave para consolidar el aprendizaje (Galán Hidalgo et al., 2024).

En una línea similar, Jiménez García y Velasco Quintana (2024) analizaron la aplicación de la simulación en entornos virtuales a través de un curso dirigido a docentes de la Universidad Europea de Madrid. El estudio evidenció un alto nivel de satisfacción por parte de los participantes, con una media de 4,69 sobre 5, así como una percepción de elevada utilidad

práctica, alcanzando una media de 4,80. Estos resultados confirman que la simulación virtual constituye una metodología efectiva y pertinente en el ámbito educativo. No obstante, los autores también señalan la existencia de desafíos relevantes, entre los que se destacan la necesidad de contar con infraestructura tecnológica adecuada y la formación específica del cuerpo docente para su correcta implementación.

Por su parte, Barbaroux (2022) aborda el aprendizaje del liderazgo desde una perspectiva crítica, subrayando la escasez de propuestas prácticas centradas en la simulación de escenarios reales de gestión educativa. El estudio pone de manifiesto la carencia de espacios donde los estudiantes puedan ejercitar habilidades como la coordinación de equipos, la gestión de conflictos y la toma de decisiones estratégicas. Si bien existe un consenso generalizado respecto a la importancia de la simulación en la formación del liderazgo, el autor destaca la falta de evidencia empírica que permita comprender en profundidad cómo los procesos prácticos fortalecen estas competencias y de qué manera los distintos mecanismos de apoyo influyen en dicho aprendizaje.

Las investigaciones revisadas respaldan la validez y el potencial de la simulación como herramienta para el desarrollo de competencias prácticas en áreas críticas, como la salud, la educación y la psicología. Sin embargo, también revelan la existencia de vacíos en relación con la simulación aplicada a contextos de gestión complejos. La evidencia disponible constituye así el punto de partida del presente trabajo, que se propone abordar dicha laguna mediante el diseño de un simulador educativo altamente realista orientado a la gestión de proyectos. En este entorno, el estudiante deberá enfrentarse a dinámicas, eventos y procesos de evaluación que emulan de manera fiel las situaciones que un gerente de proyectos encuentra en su práctica profesional.

2.3 Integración de la Inteligencia Artificial en la Educación

La integración de la inteligencia artificial (IA) en la educación implica la incorporación de un conjunto amplio de tecnologías orientadas a transformar los procesos de enseñanza, aprendizaje y gestión institucional. A través de disciplinas como el Aprendizaje Automático, el Aprendizaje Profundo y el Procesamiento del Lenguaje Natural, los sistemas basados en IA pueden aprender de manera autónoma, procesar grandes volúmenes de información y automatizar tareas complejas, generando resultados concretos y medibles en el ámbito educativo (Cunuhay Cuchipe et al., 2022).

Uno de los aportes más significativos de la IA en este contexto es su capacidad para habilitar modelos de aprendizaje adaptativo a gran escala. Estas tecnologías permiten diseñar sistemas que ajustan dinámicamente los recorridos formativos en función de las características, el desempeño y las interacciones de cada estudiante. Esto se traduce en evaluaciones personalizadas, recomendaciones de contenido ajustadas y retroalimentación automática en tiempo real. Asimismo, el análisis avanzado de datos educativos facilita la evaluación de la satisfacción estudiantil y de la calidad de la enseñanza, aportando información valiosa para la toma de decisiones basadas en evidencia dentro de las instituciones educativas (Cunuhay Cuchipe et al., 2022).

En este marco, el uso de modelos generativos como ChatGPT ha cobrado especial relevancia en la enseñanza de programación y análisis de datos. Un estudio realizado con estudiantes de la carrera de Administración y Dirección de Empresas en la Universidad Europea de Valencia evidenció que el 92 % de los participantes logró generar código complejo en Python, interpretar resultados y representar gráficamente grandes volúmenes de datos mediante el uso de esta herramienta. Estos resultados refuerzan el potencial de ChatGPT como recurso educativo, particularmente eficaz para estudiantes principiantes. No obstante, el estudio también advierte sobre el riesgo de generar dependencia si la herramienta se utiliza exclusivamente para producir código sin promover la comprensión del razonamiento subyacente al proceso de desarrollo (García-Ull, 2024).

Otro aspecto clave de la integración de la IA en la educación se relaciona con la optimización del trabajo docente y la gestión administrativa. La automatización de tareas como la corrección de evaluaciones, la revisión de trabajos prácticos y la gestión de información académica permite a los docentes liberar tiempo para actividades de mayor valor pedagógico, como el diseño de nuevas estrategias didácticas, la investigación educativa y la atención personalizada de los estudiantes. A nivel organizacional, la IA también puede emplearse para automatizar procesos administrativos y desarrollar estrategias de retención estudiantil basadas en modelos predictivos y análisis de datos (Moreno Padilla, 2022).

La diversificación de herramientas educativas basadas en IA ha ampliado significativamente las posibilidades de intervención pedagógica. Estas aplicaciones abarcan desde la generación automática de contenidos adaptados para exámenes o actividades de lectura, hasta el análisis de datos para detectar dificultades de aprendizaje, la toma de asistencia virtual y el entrenamiento en habilidades técnicas específicas (Moreno Padilla, 2019). En este

escenario, el rol del docente se ve necesariamente transformado. Herramientas como ChatGPT están redefiniendo la función del profesor, que deja de ser un transmisor exclusivo de conocimientos para asumir un papel de facilitador estratégico del aprendizaje. La automatización de tareas como la corrección de ensayos, la creación de evaluaciones y la tutoría personalizada hace posible la implementación efectiva del aprendizaje adaptativo. Esto permite que los educadores concentren sus esfuerzos en dimensiones humanas y emocionales del proceso educativo, fortaleciendo los vínculos interpersonales y el acompañamiento pedagógico. Sin embargo, diversos autores coinciden en que la interacción humana es insustituible, ya que la IA carece de comprensión real del contexto, de la experiencia subjetiva y de la complejidad de las relaciones humanas. Por ello, la IA debe concebirse como una herramienta de apoyo y complemento a la enseñanza, y no como un reemplazo del rol docente (Baidoo-Anu y Owusu Ansah, 2023).

En paralelo, surge la necesidad de promover una alfabetización digital crítica que permita a los estudiantes comprender, cuestionar y utilizar la IA de manera reflexiva y consciente. Este enfoque busca evitar una adopción acrítica de estas tecnologías, fomentando una actitud activa frente a sus limitaciones, sesgos y alcances reales (Morduchowicz, 2023). En este sentido, las universidades han comenzado a implementar normativas específicas para regular el uso ético y transparente de herramientas de IA, exigiendo que los estudiantes declaren qué sistemas utilizaron y de qué manera formularon las instrucciones o prompts correspondientes (Jiménez García y Velasco Quintana, 2024).

La investigación reciente también da cuenta del desarrollo de herramientas específicas basadas en IA para apoyar los procesos de evaluación. Un ejemplo de ello es un proyecto que utiliza técnicas de Procesamiento del Lenguaje Natural para transcribir grabaciones de clases, analizar documentos cargados por los docentes e identificar ideas principales. A partir de este análisis, el sistema genera preguntas de opción múltiple con una respuesta correcta y distractores plausibles, facilitando la creación de evaluaciones, el almacenamiento histórico de resultados y la provisión de retroalimentación personalizada a los estudiantes (Iborra Gómez et al., 2024). Este tipo de desarrollos ofrece una visión integral del papel actual de la IA en la educación, destacando su capacidad para transformar tanto la enseñanza como la gestión institucional y el aprendizaje personalizado.

La integración de la inteligencia artificial en la educación también ha impulsado la formulación de marcos éticos y regulatorios orientados a guiar su implementación. Organismos

internacionales como la UNESCO han promovido la elaboración de estándares globales destinados a garantizar un uso responsable y humanista de estas tecnologías, enfatizando principios como la equidad, la transparencia y el bienestar del alumnado. Entre las iniciativas más relevantes se encuentran el Acuerdo de Beijing sobre inteligencia artificial y educación (2019) y la Recomendación sobre la Ética de la inteligencia artificial (2021) (Jiménez García y Velasco Quintana, 2024; UNESCO, 2021). En el ámbito regulatorio, la Unión Europea ha desarrollado un marco normativo específico mediante la aprobación del Artificial Intelligence Act en 2024, que introduce un enfoque basado en niveles de riesgo para los sistemas de IA y establece restricciones para determinadas aplicaciones, como el uso de reconocimiento biométrico en espacios públicos, además de exigir mecanismos de transparencia para los contenidos generados por sistemas automatizados.

En paralelo, la investigación reciente ha comenzado a examinar de forma sistemática las aplicaciones de la IA en los procesos de evaluación educativa. El estudio “Buenas prácticas de la inteligencia artificial en el proceso de evaluación en la educación superior” analizó 130 investigaciones y detectó un incremento notable de publicaciones sobre IA aplicada a la evaluación a partir de 2023, fenómeno asociado en gran medida a la expansión de herramientas generativas como ChatGPT. Entre los principales ejes temáticos identificados se encuentran la retroalimentación automatizada, presente en el 14,6 % de los estudios, y la predicción del desempeño académico mediante análisis de datos, abordada en el 12,3 % de los casos. Estos resultados sugieren que la IA puede contribuir al fortalecimiento de los procesos de aprendizaje y evaluación, superando aplicaciones más limitadas centradas exclusivamente en la detección de plagio (Cotton et al., 2023).

En este contexto, el presente trabajo se alinea con los avances recientes en la integración de la IA en la educación, concibiéndola como un asistente en entornos virtuales capaz de generar diagnósticos personalizados y ofrecer retroalimentación automatizada. Esta perspectiva refuerza el cambio de rol del docente hacia una función de facilitador estratégico, en consonancia con los nuevos modelos pedagógicos emergentes.

2.4 La Inteligencia Artificial en el Aprendizaje por Proyectos y Desafíos

Al integrar la IA con estas metodologías, se obtiene una amplia gama de posibilidades, como monitoreo del progreso estudiantil y la provisión de retroalimentación automática; mediante el monitoreo del progreso estudiantil, genera sugerencias automáticas para mejorar las entregas y ajustar en tiempo real las actividades grupales o individuales. Esta capacidad de

respuesta inmediata alienta la autonomía y la autorregulación del alumno. El rol de la IA en el Aprendizaje Basado en Problemas se manifiesta principalmente mediante sistemas conversacionales, como los chatbots, que pueden desempeñar funciones de asistencia docente, respondiendo consultas y proporcionando retroalimentación personalizada (Jiménez García y Velasco Quintana, 2024).

La capacidad generativa de la IA puede utilizarse para la generación de ideas, escenarios simulados para prácticas profesionales, generar código de programación, representar gráficamente datos, y analizar sentimientos. La integración del Aprendizaje Basado en Proyectos y el Aprendizaje Basado en Problemas con aproximaciones transdisciplinarias y orientadas a los datos contribuye a fortalecer los procesos de aprendizaje y preparar a los estudiantes para contextos profesionales complejos, favoreciendo la toma de decisiones basadas en datos. Considerando que se está generando un puente entre teoría y práctica de la aplicación de las IA en educación, estas metodologías impulsadas por la IA contribuyen a reducir la brecha existente entre la formación universitaria y las demandas del entorno profesional, exigiendo que se apliquen los conocimientos de manera práctica (Jiménez García y Velasco Quintana, 2024).

Un análisis de la incorporación de la inteligencia artificial en el Aprendizaje Basado en Proyectos (ABP) en el ámbito de la ingeniería universitaria permite identificar un nivel de adopción todavía incipiente, aunque con proyecciones de desarrollo relevantes. En particular, el estudio de Lavado-Anguera et al. (2024) señala que las aplicaciones relevadas se concentran principalmente en usos acotados, como la asistencia mediante chatbots para la resolución de consultas y la provisión de retroalimentación personalizada, especialmente en contextos de docencia en línea. Los autores también indican que la expansión de herramientas como ChatGPT ha contribuido a mejorar la accesibilidad a este tipo de tecnologías, aunque su integración en propuestas educativas aún no se encuentra extendida de manera homogénea.

En una línea complementaria, el Aprendizaje Basado en Retos (Challenge-Based Learning, CBL) desarrollado en equipos multidisciplinarios también ofrece evidencia relevante sobre el valor de las metodologías activas en contextos de formación superior. En este sentido, Levi Alfarovi y Villagrasa Mejía (2024) describen una experiencia en la que participaron 48 estudiantes de Business Analytics, Administración y Dirección de Empresas e Ingeniería Matemática de la Universidad Europea de Madrid, en colaboración con la consultora tecnológica Nfq Advisory SL, quienes abordaron un desafío real de consultoría. Según los

resultados reportados, la experiencia se asoció con altos niveles de satisfacción y con mejoras percibidas en competencias técnicas, trabajo colaborativo y capacidad de resolución de problemas complejos. Asimismo, los autores destacan que la interacción entre estudiantes de distintas disciplinas favoreció el intercambio de perspectivas y el desarrollo de habilidades transversales vinculadas con la empleabilidad.

Un ejemplo de integración entre metodologías activas e inteligencia artificial es el proyecto PBL-CBS, orientado al desarrollo de un modelo predictivo de accidentes laborales mediante técnicas de inteligencia artificial. Estudiantes de Derecho e Ingeniería colaboraron en el desarrollo de un modelo predictivo, orientado a la clasificara los accidentes laborales en la Comunidad Valenciana, usando IA y aprendizaje automático. Al tratarse de un proyecto interdisciplinario, la experiencia permitió a los estudiantes desarrollar competencias como el trabajo colaborativo, el análisis crítico, la planificación estratégica y el liderazgo, Asimismo, al abordar problemas reales, los participantes pudieron comprender las implicaciones legales y técnicas asociadas al uso de inteligencia artificial (Fonseca Díaz y García Cuenca, 2024)

El presente trabajo identifica una convergencia clave entre la IA y las metodologías activas, como el ABP, que han adquirido creciente relevancia en el ámbito de la ingeniería, relevantes para el desarrollo de competencias profesionales. Explica cómo las plataformas inteligentes en el ABP monitorean el progreso del estudiante, generan recomendaciones automáticas y permiten ajustar actividades en tiempo real, favoreciendo la autonomía del estudiante. En este sentido, el trabajo de Lavado-Anguera et al. (2024) presenta una revisión de la aplicación de inteligencia artificial en el Aprendizaje Basado en Proyectos en el ámbito de la ingeniería universitaria. A partir de estos antecedentes, la presente propuesta plantea la integración de inteligencia artificial en un simulador de gestión de proyectos, aprovechando su capacidad para generar escenarios, personalizar el aprendizaje y dar retroalimentación, elementos fundamentales para el funcionamiento de metodologías como el Aprendizaje Basado en Proyectos (ABP) y el Challenge-Based Learning (CBL).

2.5 Simuladores para la Gestión de Proyectos

Los simuladores para la gestión de proyectos son herramientas creadas para reducir la brecha entre teoría y la práctica en la formación de gerentes de proyectos; recrean las dinámicas típicas de proyectos reales en un entorno seguro y controlado e incorporan elementos complejos como eventos aleatorios, restricciones de tiempo, conflictos de equipo internos, fallas técnicas y presiones de los interesados. Su objetivo primordial, es llevar a los usuarios a tomar

decisiones estratégicas al instante, bajo mucha presión, replicando con fidelidad las exigencias del ejercicio profesional.

En este sentido, los simuladores de gestión de proyectos cumplen un rol fundamental en el entrenamiento de habilidades críticas. A través de un enfoque de aprendizaje activo, favorecen el desarrollo de competencias clave como la priorización de tareas, la negociación, la resolución de problemas complejos y la comunicación efectiva. Este tipo de habilidades, comúnmente denominadas competencias blandas o estratégicas, resultan particularmente difíciles de evaluar y entrenar mediante métodos tradicionales de enseñanza, lo que posiciona a los simuladores como una herramienta especialmente valiosa para su adquisición y fortalecimiento.

Desde el punto de vista tecnológico, estos simuladores se apoyan en enfoques de simulación modernos que permiten modelar comportamientos complejos y realizar evaluaciones adaptativas del desempeño del usuario. Para la generación de escenarios dinámicos y realistas, suelen emplearse tecnologías como motores de reglas, árboles de decisión y sistemas multi-agente, los cuales posibilitan la evolución del entorno simulado en función de las decisiones tomadas. Esta capacidad de adaptación contribuye a ofrecer experiencias de aprendizaje personalizadas y coherentes con la diversidad de situaciones que un gerente de proyectos puede enfrentar en la práctica profesional.

2.5.1 Integración de Inteligencia Artificial

La integración de inteligencia artificial en los simuladores de gestión de proyectos constituye un factor clave para incrementar el realismo, la adaptabilidad y el valor formativo de estas herramientas. En particular, el uso de modelos de Procesamiento del Lenguaje Natural, como los desarrollados por OpenAI o Cohere, permite simular interacciones complejas con los distintos agentes involucrados en un proyecto, tales como clientes, desarrolladores y patrocinadores. Estas tecnologías hacen posible que el usuario practique habilidades de comunicación, negociación y manejo de expectativas en un entorno controlado, pero con un alto grado de verosimilitud respecto de las dinámicas reales del ámbito profesional. Para escenarios de mayor complejidad, se contempla la incorporación de técnicas de Aprendizaje por Refuerzo, orientadas a calibrar dinámicamente la dificultad del simulador en función del desempeño del usuario. Este enfoque permite construir una curva de aprendizaje verdaderamente adaptativa, ajustando los desafíos propuestos según las fortalezas y debilidades detectadas durante la interacción. De este modo, el entrenamiento puede focalizarse

en aspectos específicos del perfil del participante, como la gestión de conflictos, la toma de decisiones bajo presión o la coordinación de equipos, potenciando la personalización del proceso formativo (Barbaroux, 2022).

Desde una perspectiva pedagógica, estos simuladores no se conciben como un reemplazo del rol docente, sino como un complemento que amplifica y fortalece su función. Su integración en programas académicos permite utilizarlos tanto con fines formativos como evaluativos, enriqueciendo las propuestas curriculares existentes y ofreciendo nuevas herramientas para observar y analizar el desempeño de los estudiantes en contextos complejos y dinámicos (Barbaroux, 2022).

El documento de Barbaroux (2022), si bien no menciona un simulador particular, pone en evidencia la escasa evidencia empírica sobre la forma en que el aprendizaje pragmatista favorece el desarrollo de habilidades de liderazgo usando la simulación. Este hallazgo subraya la pertinencia de propuestas como la presentada en este trabajo, orientadas precisamente a generar evidencia empírica y a ofrecer formación práctica en este ámbito.

Este trabajo proporciona una propuesta de intervención en el campo profesional que responde a la ausencia de ofertas concretas para simular contextos reales de gestión educativa, donde se entrenen habilidades de coordinación, solución de conflictos y toma de decisiones bajo presión. En respuesta a ese diagnóstico, el presente trabajo propone un simulador educativo realista orientado a formar y evaluar competencias de liderazgo y gestión de proyectos, contribuyendo a reducir la brecha identificada entre la formación académica y el ejercicio profesional.

2.6 Justificación de la Propuesta

El análisis del estado del arte evidencia una brecha persistente entre la enseñanza teórica de la gestión de proyectos y el desarrollo efectivo de competencias estratégicas en contextos dinámicos. Si bien la literatura reconoce la importancia de habilidades como la coordinación, la resolución de conflictos y la toma de decisiones bajo presión, son escasas las propuestas que modelan escenarios auténticos donde estas capacidades puedan ejercitarse de forma sistemática y evaluable. Incluso en programas orientados al liderazgo, persiste una limitada evidencia empírica sobre el impacto de los procesos prácticos en el desarrollo real de habilidades complejas (Barbaroux, 2022).

En contraste, el aprendizaje experiencial ha demostrado ser un enfoque sólido para integrar teoría y práctica, promoviendo una comprensión profunda y transferencia efectiva al

ámbito profesional. Diversas iniciativas académicas como la Clínica de Sostenibilidad, UEM Solar o experiencias de inmersión disciplinar confirman que la exposición a situaciones reales o simuladas favorece la adquisición de competencias aplicadas (Jiménez García y Velasco Quintana, 2024). En este marco, metodologías activas como el Aprendizaje Basado en Problemas (PBL) y el Challenge-Based Learning (CBL) fortalecen el pensamiento crítico y la capacidad de resolución en entornos controlados. Las simulaciones, particularmente en campos como medicina o psicología, han mostrado resultados consistentes en la mejora del desempeño práctico y en la satisfacción docente.

La incorporación de inteligencia artificial amplifica este potencial pedagógico. Los sistemas adaptativos permiten personalizar la experiencia formativa, ajustar niveles de dificultad y generar retroalimentación automática basada en el desempeño. Asimismo, las herramientas de IA generativa facilitan la construcción de escenarios complejos, análisis de decisiones y simulación de interacciones, enriqueciendo el proceso formativo (Cunuhay Cuchipe et al., 2022). Este avance se enmarca en lineamientos éticos y regulatorios impulsados por organismos internacionales, que promueven un uso responsable, transparente y centrado en la persona (Macías Lara, 2023; UNESCO, 2021).

Finalmente, la transformación del mercado laboral exige profesionales capaces de liderar proyectos en entornos interdisciplinarios, inciertos y tecnológicamente complejos. Las tendencias en educación superior europea refuerzan la necesidad de modelos curriculares integrados, orientados a la experiencia y al desarrollo de competencias digitales avanzadas (Morduchowicz, 2023; Moreno Padilla, 2019). En este contexto, y en comparación con las soluciones actualmente disponibles, la presente propuesta incorpora un enfoque de simulación inteligente y adaptativa articulado con técnicas de inteligencia artificial. Mientras que las alternativas tradicionales se centran en la resolución de situaciones fijas con evaluación principalmente cualitativa o manual, el modelo aquí planteado introduce generación dinámica de eventos, retroalimentación automática basada en métricas de desempeño y ajuste progresivo de la dificultad en función del comportamiento del usuario. Su aporte específico radica en integrar simulación contextual, analítica formativa y adaptación personalizada en un entorno seguro, escalable e integrable curricularmente, orientado no solo a verificar conocimiento declarativo sino a entrenar competencias estratégicas bajo condiciones de incertidumbre controlada, aportando un modelo sistemático de evaluación del desempeño que actualmente no se encuentra integrado de forma estructurada en las soluciones existentes.

Capítulo 3 – Marco Teórico

3.1 Tecnologías de la Información y Comunicación en la Educación

Las tecnologías de la información y la comunicación (TIC) constituyen un componente fundamental de la sociedad de la información y de la era digital, cuya relevancia ha aumentado significativamente en los nuevos modelos de enseñanza y aprendizaje. La incorporación de la tecnología en la educación no implica solamente el uso de equipos sofisticados, sino que implica una transformación del proceso educativo y su integración en las distintas etapas del proceso didáctico (Guzmán et al, s.f.). Estas nuevas tecnologías son incorporadas principalmente en la educación multimedia, donde se busca brindar a los alumnos las destrezas y cualidades para comunicarse efectivamente y con un razonamiento crítico en un mundo donde la innovación tecnológica es constante y exponencial (Gutiérrez, 1997).

3.1.1 Evolución de las TIC en Contextos Educativos

La evolución de las TIC en entornos educativos ha experimentado un proceso de transformación significativo, modificando de forma significativa, los métodos de enseñanza y los roles de los agentes educativos. A finales del siglo XX, el avance de las TIC alteró de manera sustancial los saberes básicos, trasladando el enfoque de la información en sí misma a su tratamiento y procesamiento (Gutiérrez, 1997). Aunque la educación formal no siempre ha incorporado los nuevos medios con la velocidad de su desarrollo, su omnipresencia y carácter lúdico han generado un cambio en la función socializadora y educativa. Las redes informáticas fortalecieron la interacción entre educadores y estudiantes, además de hacer más flexible y personalizada la educación a distancia para los alumnos, que históricamente era pasiva y ofrecía niveles limitados de interacción (Adell y Sales, s.f.).

Con el avance de estas tecnologías, emergió la inteligencia artificial (IA), inicialmente considerada una tecnología utópica, dada la dificultad de simular el comportamiento humano, pero actualmente se reconoce como una herramienta con un potencial significativo para mejorar los procesos educativos (Castillo Herrera, 2023). La inteligencia artificial en educación persigue una comprensión más profunda, buscando la optimización del aprendizaje. En los años ochenta, la innovación cristalizó en los Sistemas Tutores Inteligentes (STI), derivando de la instrucción asistida por computadora (*Computer-Assisted Instruction*, por sus siglas en inglés, CAI). Estos aspiraban a replicar el comportamiento de un tutor humano, adaptando la enseñanza a las preferencias y al estilo de aprendizaje particular del discente. Simultáneamente, nacieron los micromundos, ambientes de software interactivos con los cuales se capacita a los

estudiantes para explorar ideas y construir conocimientos a través del descubrimiento activo, distanciándose de la simple transmisión pasiva de datos (Gross, 1992; Guzmán et al., s.f.).

Hoy en día, la irrupción de las herramientas de IA generativa ha impulsado cambios relevantes, facilitando a los educadores la producción de contenidos innovadores y dinámicos. Del mismo modo, proporciona tutorías y trayectorias de aprendizaje personalizadas, todo ello en tiempos significativamente reducidos. (Castillo Herrera, 2023; Cunuhay Cuchipe et al., 2022). Dichas tecnologías, además, permiten la automatización de tareas administrativas cotidianas de los docentes, proporcionando libertad para dedicarse a tareas pedagógicas como la investigación, el diseño de nuevas metodologías educativas y la atención individualizada a cada alumno. Los chatbots colaboran con esto último en aclaraciones y gestión de trámites administrativos, optimizando la flexibilidad y la administración del conocimiento (Cunuhay Cuchipe et al., 2022).

Aun así, la integración de la IA en la educación enfrenta retos considerables. Entre ellos, la brecha digital y la costosa infraestructura, además de la formación continua de profesores y estudiantes en habilidades digitales y de ética. Existe preocupación respecto a que el uso inadecuado de la IA pueda mecanizar ciertos procesos educativos o reemplazar la interacción personal, se hace hincapié en que la IA debería colaborar con el docente, no sustituirlo; y le permite enfocarse en aspectos humanos como la empatía, la creatividad y el pensamiento crítico (Baidoo-Anu y Owusu Ansah, 2023; Moreno Padilla, 2019). La función del educador cambia hacia la de ser un facilitador, guía y diseñador de experiencias de aprendizaje; necesitando un balance entre la eficiencia tecnológica y los principios pedagógicos tradicionales para asegurar un uso responsable, inclusivo y equitativo de la IA (Jara y Ochoa, 2020; Moreno Padilla, 2019).

3.1.2 Metodologías Activas Mediadas por TIC

Aprendizaje Basado en Proyectos Digitales. El Aprendizaje Basado en Proyectos (ABP) se presenta como una estrategia educativa relevante para afrontar los desafíos del mercado laboral del siglo XXI, ayudando a desarrollar habilidades como la creatividad, el trabajo en equipo, la autonomía y el pensamiento lógico. Tal enfoque se vincula estrechamente con la educación STEAM (Ciencia, Tecnología, Ingeniería, Artes y Matemáticas) promoviendo un proceso de aprendizaje multidisciplinar. Se fundamenta en proyectos auténticos, que reproducen escenarios de la vida diaria, a menudo vinculados a la robótica y la inteligencia artificial, utilizando herramientas que favorecen la familiarización de los estudiantes con

tecnologías emergentes como la inteligencia artificial así como desarrollar un uso más crítico y competente de las herramientas tecnológicas disponibles.

A través del ABP, los estudiantes desarrollan habilidades clave para su formación científico-tecnológica. Entre ellas se encuentran habilidades como la programación, el pensamiento computacional, la lógica y los algoritmos, además de las actitudes científicas esenciales, como la curiosidad y la investigación. También se promueve el interés en la cultura tecnológica, estimulando la creatividad y la innovación al permitirse la exploración de múltiples soluciones ante los problemas (Chevalier et al., 2025). Este enfoque establece un vínculo entre la enseñanza tradicional y el desarrollo de tecnologías capaces de interactuar con los usuarios, lo cual contribuye al desarrollo de competencias tecnológicas e informáticas. La IA puede usarse como apoyo en la realización de actividades e investigaciones, así como el uso de robótica educativa facilita la elaboración de proyectos colaborativos; en los que los estudiantes idean y editan máquinas básicas, impulsando la integración curricular y también la retroalimentación (Moreno Padilla, 2019).

Gamificación y Entornos Virtuales. La gamificación es una metodología que incorpora dinámicas lúdicas para incrementar la motivación y el compromiso estudiantil (Barbay, 2021). El aprendizaje basado en juegos, o mejor dicho los “juegos serios” en ámbito educativo, aprovecha el potencial de los videojuegos para captar la atención y promover el desarrollo de nuevas habilidades y conocimientos. La IA se integra a los elementos de juego en aplicaciones educativas, alentando habilidades sociales, de comunicación, trabajo colaborativo, persistencia, creatividad, autoeficacia y resolución de problemas en entornos recreativos y controlados (Bahreini et al., 2020; Lama Penin y Sánchez Vila, 2007).

Un elemento clave en la gamificación son los sistemas de recompensa y progresión; aplicaciones como Duolingo implementan la gamificación y la repetición espaciada para adaptar el orden y dificultad de los ejercicios del estudiante (Barbay, 2021). Aplicaciones como Brain Training de Nintendo se buscan mejorar determinadas habilidades cognitivas, a través de prácticas que guían a una evolución, promoviendo la mejora de determinadas capacidades cognitivas (Lama Penin y Sánchez Vila, 2007). La IA generativa ha demostrado potencial para incrementar la motivación y el entusiasmo en el estudiante, particularmente al crear materiales y contenidos educativos (Castillo Herrera, 2023); así como los sistemas adaptativos, alimentados con IA, que se vinculan con una mayor autonomía, dedicación, y proactividad del

alumno en su proceso personal de aprendizaje (Horizon Report, 2019, como se citó en Jara y Ochoa, 2020).

Simulaciones y Entornos Virtuales de Aprendizaje. Las simulaciones son programas informáticos que replican aspectos del mundo real, permitiendo a los estudiantes modificar parámetros y observar los resultados de sus acciones; fomentan la interactividad y la participación del estudiante, facilitando la exploración de conceptos y la construcción del conocimiento, a través de la interacción con la máquina (Gross, 1992; León y Viña Brito, 2017). El dominio de estos modelos computacionales resulta clave para el aprendizaje experiencial de teorías científicas formalizadas (Vilches Márquez, 2005). Los micromundos, por su parte, son entornos de software interactivos donde los estudiantes pueden crear objetos, explorar relaciones entre diferentes entidades y desarrollar habilidades de descubrimiento y formulación de hipótesis (Guzmán et al, s.f.). Los simuladores y los entornos virtuales proporcionan entornos controlados para la práctica y el ensayo de escenarios que, en el mundo tangible, serían impracticables debido a costes, tiempo, infraestructura o riesgo (Lama Penin y Sánchez Vila, 2007), posibilitan modelar y simular interacciones sociales o profesionales, creando espacios seguros donde los docentes pueden experimentar con distintas estrategias pedagógicas (Vilches Márquez, 2005). En cuanto a las plataformas colaborativas, herramientas como VirtualProfe favorecen el aprendizaje colectivo por medio de chats y foros, posibilitando el debate, propuestas y el diseño colaborativo de proyectos educativos (Buzon-Garcia, 2005). El aprendizaje cooperativo asistido por ordenador mejora el aprendizaje en tareas grupales complejas, respaldado por agentes de software que median y colaboran en la interacción (Guzmán et al., s.f.; Lama Penín y Sánchez Vila, 2007).

En relación con la recreación de contextos profesionales, los Sistemas Tutores Inteligentes, o STI, se emplean para recrear situaciones complejas en áreas como medicina, como COMET y VIPER, así como en programación, mediante sistemas como Beetle II System y EER-Tutor, lo cual facilita el aprendizaje práctico en contextos que simulan entornos profesionales (Arana, 2021), además de permitir el entrenamiento con equipamiento costoso o para el desarrollo de nuevas aptitudes por personal ubicado a distancia, ahorrando así desplazamientos, En consecuencia, las simulaciones resultan valiosas en el adiestramiento profesional, industrial y militar (Park Woolf, 2009).

Los agentes virtuales inteligentes exhiben autonomía, trabajo en equipo y aprendizaje. Estos pueden funcionar como tutores virtuales, aprendices simulados o colaboradores en el

aprendizaje, apoyando a los estudiantes en su proceso de aprendizaje. (Guzmán et al., s.f.). Los tutores virtuales, además de calificar trabajos, identifican fallos frecuentes y otorgan realimentación exacta e instantánea. El desarrollo de este tipo de software ha avanzado rápidamente, como una solución pronta a las urgencias educativas, investigaciones recientes han evidenciado su capacidad, un hecho comprobado, para generar interacciones.

> El desarrollo de este tipo de software se ha convertido en una solución rápida a las necesidades educativas de lo atemporal de la educación virtual está herramienta va evolucionando muy rápidamente y, en trabajos recientes, se ha comprobado su utilidad para generar conversaciones que se están empezando a usar para realizar evaluaciones automáticas y adaptativas de respuestas incluso con textos abiertos. (Moreno Padilla, 2019, pág. 264).

Estos agentes, capaces de modificar su conducta, pueden colaborar o competir con el estudiante, ofreciendo consejo y aprendiendo en el proceso, una dinámica recíproca. El proyecto EVA, por su parte, ideó un entorno multi-agente para asesoramiento y asistencia personal inteligente. Esto incorpora agentes de búsqueda, filtrado de información, síntesis, evaluación e interpretación de datos, una composición compleja (Guzmán et al., s.f.).

Herramientas de Seguimiento y Evaluación. Actualmente las plataformas digitales emplean sistemas que rastrean y evalúan el avance académico de los estudiantes; además, los algoritmos de la IA poseen la capacidad de discernir tendencias en el comportamiento estudiantil, lo que permite identificar posibles dificultades de aprendizaje, al mismo tiempo que brindan datos importantes para informar a los profesores cuando intervenir, así como maneras de adaptar el contenido, personalizándolo (Arana, 2021; Cunuhay Cuchipe et al., 2022).

3.1.3 Evaluación Digital de Competencias

La evaluación en el ámbito de la tecnología educativa no se limita a medir el desempeño del software o los resultados del aprendizaje, sino que implica analizar cada componente, su utilización y su impacto en los procesos de aprendizaje; se procura garantizar la calidad y funcionalidad del software en escenarios académicos auténticos (Park Woolf, 2009).

La IA generativa también puede utilizarse para ofrecer evaluaciones objetivas y una retroalimentación personalizada, mediante el uso de métricas de desempeño en tiempo real, como ocurre en los Sistemas Tutores Inteligentes. Estos sistemas proporcionan instrucción y

retroalimentación personalizadas., aumentando la eficiencia del aprendizaje; utilizando algoritmos, la IA puede realizar evaluaciones automáticas e identificar estudiantes con dificultades, lo cual proporciona a los docentes información valiosa, permitiéndoles dedicar más tiempo al diseño de contenidos adaptados (Arana, 2021; Cunuhay Cuchipe et al., 2022). Por medio de la analítica del aprendizaje, se recopilan y analizan datos como el tiempo de estudio y la evaluación de pruebas, con el objetivo de emular el comportamiento de un tutor humano, prediciendo los resultados e informando la planificación educativa. (Arana, 2021). Los STI proporcionan una instrucción adaptada al alumno, modificando su contenido y forma, de manera tal que buscan imitar a un profesor para evitar las restricciones del software educativo tradicional (Gross, 1992). Estos sistemas presentan contenido según el estilo de aprendizaje del estudiante, ofrecen asesoría sobre métodos de estudio, ayudan en el logro de objetivos, colaboran en el trabajo en grupo, así como diagnostican el desempeño y proveen herramientas de mejora (Cataldi y Lage, 2009).

Los Sistemas de Evaluación Automática (SEA) se usan para examinar fortalezas y áreas a mejorar en los estudiantes mediante pruebas, estos sistemas pueden calificar por sí solos exámenes de selección múltiple, redacciones e incluso generar preguntas para las pruebas. Ejemplos como ToL (Test on Line) en física y CELLA (Comprehensive English Language Learning Assessment) en inglés demuestran la capacidad para evaluar habilidades y el progreso (Jara y Ochoa, 2020; Lama Penín y Sánchez Vila, 2007). Iniciativas tales como TENCompetence, emplean enfoques de IA para analizar el perfil estudiantil así como las tareas de aprendizaje implementadas, con el objetivo de configurar planes de aprendizaje personalizados, una cualidad clave en el perfeccionamiento constante de aptitudes profesionales (Lama Penin y Sánchez Vila, 2007).

3.2 Inteligencia Artificial

3.2.1 Fundamentos de la Inteligencia Artificial

“La historia de la inteligencia artificial es una colección de mitos, fantasías, mecanismos, teorías y ensayos” (Banda, 2014, p. 2).

El origen de la inteligencia artificial se remonta a los intentos por reproducir el razonamiento humano mediante reglas formales, aunque pronto se reconoció la necesidad de sistemas capaces de aprender por sí mismos (Murphy, 2022). Desde Aristóteles, con su lógica silogística, hasta Descartes, Pascal y Babbage, distintos pensadores imaginaron máquinas que pudieran razonar o calcular. En 1950, Alan Turing formuló la pregunta “¿Pueden pensar las

máquinas?" y propuso la Prueba de Turing como criterio de inteligencia (Cevallos, 2008). El nacimiento formal de la IA moderna se ubica en la Conferencia de Dartmouth (1956), donde John McCarthy, Marvin Minsky, Claude Shannon y otros acuñaron el término y sentaron sus bases (Ganascia, 2018; Banda, 2014).

Como disciplina, la IA integra aportes de la filosofía, matemáticas, psicología, lingüística e ingeniería informática, integrando enfoques relacionados con el razonamiento, la probabilidad, el lenguaje y la computación (Norvig y Russell, 1995). A lo largo de su evolución ha atravesado ciclos de auge y estancamiento, pero ha producido avances significativos en aprendizaje automático, procesamiento del lenguaje natural y robótica, destacándose sus aplicaciones educativas, como los tutores inteligentes, que ofrecen aprendizaje personalizado y adaptable (Bialik et al., 2019; Jara y Ochoa, 2020).

IA Débil vs IA Fuerte. El concepto IA débil o IA estrecha, alude a sistemas de inteligencia artificial diseñados para resolver tareas específicas., permitiendo a los sistemas computacionales resolver problemas y razonar únicamente en un campo restringido. La IA débil busca simular determinadas capacidades cognitivas humanas en contextos específicos, lo cual beneficia a la sociedad automatizando acciones y procesos. Esta perspectiva sostiene que los sistemas inteligentes no necesitan replicar de forma exacta los procesos cognitivos humanos. La mayor parte de la investigación efectuada en la segunda mitad del siglo XX se ubica en esta definición (Banda, 2014; García-Peñalvo, 2005).

En cambio, cuando hablamos de la IA fuerte, o general, se refiere a sistemas hipotéticos capaces de replicar plenamente las capacidades cognitivas humanas y tener otros estados cognitivos. La premisa básica de la IA fuerte es que los procesos realizados por una computadora serían equivalentes a los procesos cognitivos humanos., implicando que, si el cerebro genera conciencia, las computadoras también deberían ser conscientes. Sin embargo, se argumenta que las aplicaciones de IA tratan únicamente con la sintaxis, no con la semántica, y por lo tanto un computador no puede explicar la intencionalidad o significado de lo que procesa. actualmente se considera un objetivo teórico aún no alcanzado, aunque el campo de la robótica ha impulsado su estudio, se han evidenciado las dificultades de generar modelos de procesos mentales que aún no se comprenden a nivel suficiente (Banda, 2014).

Paradigmas de IA: Simbólica, Conexionista, e Híbrida. La inteligencia artificial simbólica se refiere a sistemas que operan mediante símbolos abstractos, tales como ideas, conceptos o representaciones formales, este enfoque representa el modelo tradicional de la

inteligencia artificial.; se basa en la formalización de los problemas mediante representaciones simbólicas del conocimiento, como redes semánticas, y en su posterior procesamiento mediante algoritmos de búsqueda, utilizando algoritmos que permiten explorar posibles soluciones (Banda, 2014). La computación simbólica es la disciplina tecnológica que crea aplicaciones inspiradas en modelos clásicos de IA, un ejemplo es el lenguaje Logo, usado en la enseñanza (Cevallos, 2008).

El segundo paradigma, el paradigma conexionista, es representado principalmente por las redes neuronales artificiales (RNA), sistemas subsimbólicos que se caracterizan por su capacidad de aprender de forma iterativa a partir de datos, basándose en datos experimentales. Entre los ejemplos de sistemas inteligentes computacionales sub-simbólicos están la computación neuronal y la computación evolutiva; "Estos sistemas trabajan bajo conceptos fuertemente relacionados como la autonomía, el aprendizaje y la adaptación" (Banda, 2014).

El tercer y último paradigma es el de la IA Híbrida, donde se adoptan aproximaciones híbridas incorporando las mejores teorías, modelos y tecnologías, tanto tradicionales como modernas. Las tecnologías hibridas han emergido con la habilidad de combinar las representaciones simbólicas del conocimiento con otras tecnologías que se asemejan más al comportamiento humano y demuestran gran flexibilidad (Baradó et al., 2013).

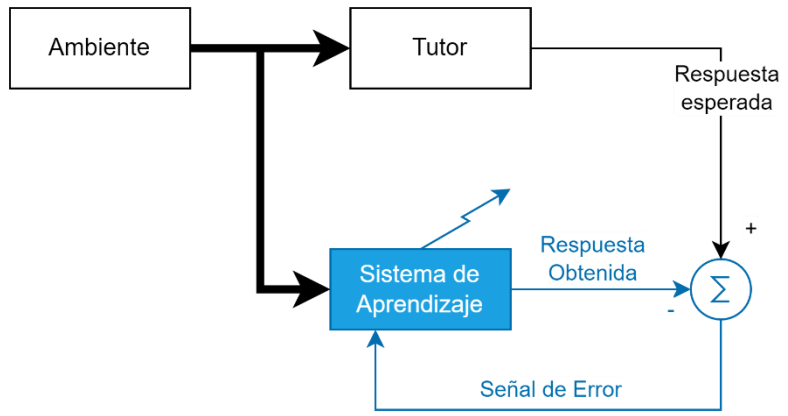
El Aprendizaje de la IA. El aprendizaje de las máquinas opera en un marco formal compuesto de tres elementos, que modelan la interacción entre el agente y su entorno: (a) agente y entorno, donde el agente es quien aprende y toma decisiones, mientras que el entorno abarca todo lo demás con lo que el agente interactúa de forma continua; el límite entre ambos representa la frontera de control del agente; (b) meta y recompensa, en la que el objetivo se formaliza mediante una señal de recompensa (Rt) enviada desde el entorno al agente, quien busca maximizar la recompensa acumulada a largo plazo, conforme a la denominada hipótesis de la recompensa; y (c) estados y acciones, dado que la interacción se basa en acciones específicas seleccionadas por el agente en función de los estados que las condicionan, definiendo así el proceso de toma de decisiones (Barto y Sutton, 2018). Según Bishop (2006) los sistemas de IA contemporáneos se categorizan dependiendo del tipo de supervisión que reciben durante su adiestramiento:

Aprendizaje supervisado. El aprendizaje supervisado (*Supervised Learning*, por sus siglas en inglés, SL) implica la existencia de un mecanismo que supervisa el proceso de aprendizaje de aprendizaje; en este sistema, una computadora adquiere la capacidad de

relacionar entradas con salidas utilizando un conjunto de entrenamiento con etiquetas, esto significa que, para cada dato de entrada, se le asigna la salida esperada o "señal guía" (Goldberg, 2017). Esta metodología resulta apropiada para aquellos escenarios donde las respuestas deseadas son discernibles, o donde se dispone de ejemplos que demuestran el comportamiento deseado, incluso en casos en los que el procedimiento exacto que conduce a la solución no es conocido por los humanos. En la Figura 1 (Haykin, 2009) se ilustra como un sistema de aprendizaje supervisado compara su salida generada con una respuesta dada por el tutor, y utiliza una señal de error para modificar su comportamiento.

Figura 1

Diagrama de bloques de aprendizaje con un tutor.



Nota. El diagrama representa el flujo fundamental de un sistema de aprendizaje supervisado: el *tutor* proporciona la respuesta esperada, el *sistema de aprendizaje* genera una respuesta obtenida y compara ambas para producir una señal de error, que luego retroalimenta el proceso para ajustar el comportamiento futuro. Adaptado de *Neural Networks and Learning Machines*, de S. Haykin, 2009, Pearson Prentice Hall.

Aprendizaje no supervisado. El aprendizaje no supervisado (*Unsupervised Learning*, por sus siglas en inglés, UL) ocurre en ausencia de un instructor o supervisor externo que dicte las respuestas correctas. El sistema aprende al explorar y reorganizar la información recibida, con el fin de descubrir de forma autónoma las características o patrones presentes en los datos (Cevallos, 2008). Estos métodos permiten el aprendizaje mediante la exploración de los datos, sin requerir datos previamente etiquetados. A pesar de esto, requieren una elevada cantidad de recursos computacionales, tanto en tiempo de procesamiento como en memoria, lo que a veces provoca que el sistema no alcance los objetivos deseados de aprendizaje. Un tipo de técnicas de aprendizaje no supervisado son las redes neuronales autoorganizadas, como los Mapas

Autoorganizativos de Kohonen, y la Teoría de la Resonancia Adaptativa, se sitúan como ejemplos destacados, así como distintos métodos de reducción de dimensionalidad (Cevallos, 2008).

3.2.2 Aprendizaje por Refuerzo

El Aprendizaje por Refuerzo (*Reinforcement Learning*, por sus siglas en inglés, RL) estudia como un agente puede aprender a actuar en un entorno, es decir, establecer una relación entre estados del entorno y acciones, con el objetivo de maximizar una señal numérica de recompensa. El agente tiene el objetivo de aumentar al máximo el retorno esperado, el cual representa la ganancia acumulada que obtiene en el curso del tiempo (Barto y Sutton, 2018). Esta metodología se inspira en la forma en que los animales y los humanos aprenden, a través del ensayo y error, recibiendo recompensas o castigos por sus acciones (Haykin, 2009). Los sistemas de RL integran por cuatro componentes clave:

Política (π): Gobierna el comportamiento del agente, consiste en un mapeo entre los estados del entorno y las acciones que el agente debe ejecutar en tales estados (Barto y Sutton, 2018).

Señal de Recompensa: Indica la calidad de una acción instantánea, cuantificando si una acción resultó beneficiosa (Feit et al., 2024).

Función de Valor: Anticipa las recompensas futuras. El valor de un estado es la suma esperada de recompensas obtenidas, entre el instante presente y la finalización de la secuencia. Estimar la función de valor es fundamental para la búsqueda eficiente en el vasto espacio de las políticas (Barto y Sutton, 2018).

Modelo: De ser provisto, simula las dinámicas del mundo exterior, lo cual posibilita deducir comportamientos futuros (Barto y Sutton, 2018).

La optimización de decisiones secuenciales es el foco del aprendizaje por refuerzo. El agente, en un proceso de aprendizaje, aprende una relación entre estados y acciones a través de la interacción constante con el entorno, buscando maximizar una función de rendimiento. Esencialmente, el sistema no es instruido directamente acerca de las acciones correctas, sino que aprende de una recompensa o penalización diferida., con el fin de elegir secuencias de acciones que proporcionen la mayor recompensa acumulada (Haykin, 2009). Los algoritmos de aprendizaje por refuerzo guardan relación con los algoritmos de programación dinámica, herramienta útil para solucionar problemas de optimización. Se aplica en ámbitos tan diversos

como la conducción de robots móviles, la optimización de los procesos industriales, o el aprendizaje de juegos de mesa (Mitchell, 1997).

El aprendizaje por refuerzo se sustenta de manera intrínseca en la adaptación fundamentada en retroalimentación; A diferencia del aprendizaje supervisado, el sistema no recibe las acciones correctas de manera directa, se le otorga una señal, una recompensa o un castigo, para saber cuán favorable es el estado resultante de sus acciones. Esta retroalimentación es clave para el agente, puesto que le permite aprender una función que relacione estados y acciones exitosamente. Esto posibilita que el sistema se ajuste y perfeccione su conducta a lo largo del tiempo, evaluando su propio proceso de aprendizaje, así como las estrategias pedagógicas que emplea. El RL tiene el potencial de hacer que un tutor sea más adaptable y flexible, permitiendo así la modificación dinámica de las estrategias pedagógicas, todo para que se ajusten a los grupos heterogéneos de estudiantes (Park Woolf, 2009).

Una dificultad central en el Aprendizaje por Refuerzo, que no aflora del mismo modo en el aprendizaje supervisado o no supervisado en su forma más básica, es la necesidad de equilibrar la exploración y la explotación. La explotación se enfoca en llevar a cabo acciones ya probadas, que gracias a la experiencia previa, han demostrado generar una recompensa significativamente alta. Mientras que, en la exploración, el agente se concentra en elegir acciones destinadas a recopilar más datos para su entrenamiento; eso capacita al agente para tomar mejores decisiones más adelante, aun cuando implique una menor recompensa a corto plazo (Barto y Sutton, 2018; Bengio et al., 2016).

Los métodos de aprendizaje por refuerzo suelen clasificarse en dos grupos principales, según si emplean de forma explícita un modelo del entorno: Métodos Basados en Modelos y Métodos Libres de Modelo. Los métodos basados en modelos (*Model-based methods*), emplean un modelo del entorno para la planificación. Este modelo se aprende a partir de la experiencia real y produce experiencia simulada para la planificación, ejemplo de esto es la arquitectura Dyna-Q. Por otro lado, los métodos libres de modelo (*Model-free methods*) se basan en la experiencia directa sin un modelo explícito del entorno, sin un modelo de planificación inherente (Barto y Sutton, 2018).

3.2.3 Aprendizaje Automático, Redes Neuronales y Aprendizaje Profundo

El Aprendizaje Automático (*Machine Learning*, ML) constituye una de las ramas fundamentales de la inteligencia artificial y se orienta al desarrollo de métodos que permiten a los sistemas aprender a partir de la experiencia y mejorar su desempeño sin intervención

humana directa. En términos generales, un programa se considera capaz de aprender cuando, frente a una tarea determinada y una medida de rendimiento específica, su desempeño mejora como resultado de la experiencia acumulada (Norvig y Russell, 1995). Este enfoque permite identificar patrones en los datos y utilizarlos para predecir resultados o tomar decisiones en situaciones nuevas, construyendo modelos capaces de generalizar a partir de información previa (Mitchell, 1997).

Desde una perspectiva estructural, Deisenroth et al. (2020) señalan que todo sistema de aprendizaje automático se apoya en tres pilares fundamentales. En primer lugar, los datos constituyen la base empírica del aprendizaje, ya que proporcionan la información a partir de la cual el sistema extrae regularidades. En segundo término, el modelo representa la función predictiva que da sentido a dichos datos y permite realizar inferencias. Finalmente, el proceso de aprendizaje se encarga de ajustar los parámetros del modelo mediante técnicas de optimización, con el objetivo de minimizar el error y maximizar la capacidad de generalización frente a nuevos casos.

En cuanto a las tareas que aborda el Aprendizaje Automático, se destacan algunas especialmente relevantes. La clasificación permite asignar una categoría a una entrada determinada, como ocurre en el reconocimiento de objetos dentro de imágenes, siendo un ejemplo clásico el reconocimiento óptico de caracteres (*Optical Character Recognition*, OCR), donde imágenes de texto se transforman en caracteres digitales. La regresión, por su parte, se orienta a la predicción de valores numéricos continuos, como ocurre en la estimación de precios, temperaturas o probabilidades. Finalmente, los sistemas de recomendación utilizan el comportamiento previo del usuario para ofrecer contenidos personalizados, una aplicación particularmente valiosa en el ámbito educativo, donde resulta posible adaptar materiales, ejercicios y planes de estudio a las necesidades individuales de los estudiantes (Géron, 2022).

Las Redes Neuronales Artificiales (RNA) constituyen uno de los enfoques más influyentes del aprendizaje automático moderno y el fundamento del Aprendizaje Profundo. Inspiradas en el funcionamiento del cerebro humano, las RNA están formadas por nodos o neuronas artificiales organizadas en capas interconectadas. Estas neuronas transforman las entradas mediante pesos, sesgos y funciones de activación no lineales, lo que permite modelar relaciones complejas entre los datos (Mitchell, 1997; Bishop, 2006).

La unidad básica de estos modelos es la neurona artificial, también conocida en sus primeras formulaciones como unidad de lógica de umbral (*Threshold Logic Unit*, por sus siglas

en inglés, TLU), que combina las señales de entrada y activa una salida cuando se supera un determinado umbral, emulando de forma simplificada el comportamiento de una neurona biológica. Esta arquitectura posibilita la aproximación de funciones no lineales y el reconocimiento de patrones complejos, lo que resulta especialmente útil en entornos caracterizados por la incertidumbre. En el ámbito educativo, las redes neuronales pueden emplearse para modelar procesos cognitivos, detectar errores conceptuales y predecir el progreso del estudiante, contribuyendo de manera directa a la personalización del aprendizaje.

El Aprendizaje Profundo (Deep Learning, DL) representa la evolución contemporánea de las redes neuronales artificiales y se basa en arquitecturas de múltiples capas, conocidas como redes neuronales profundas, capaces de extraer representaciones progresivamente más abstractas de los datos (Bengio et al., 2016). Mientras que las capas iniciales capturan rasgos simples, como bordes o frecuencias, las capas superiores combinan estas características para identificar estructuras de mayor nivel, tales como rostros, gestos o conceptos complejos.

Este procesamiento jerárquico permite al Aprendizaje Profundo superar muchas de las limitaciones de los algoritmos tradicionales, abordando con eficacia problemas de alta dimensionalidad y gran complejidad. Durante el entrenamiento, el modelo compara sus predicciones con los valores esperados y ajusta sus pesos mediante algoritmos de optimización como el descenso por gradiente (*Gradient Descent*), reduciendo progresivamente el error de predicción. Cada capa actúa como un filtro que refina la información entrante hasta alcanzar una representación eficiente del conocimiento subyacente. Este flujo unidireccional de información, que va desde la entrada hasta la salida del modelo, se conoce como propagación hacia adelante (*forward propagation*).

Desde una perspectiva aplicada, las técnicas de Aprendizaje Automático, Redes Neuronales y Aprendizaje Profundo han generado un avance significativo en la comprensión y personalización de los procesos de aprendizaje. En el contexto de los Sistemas Tutores Inteligentes, estas tecnologías permiten modelar el conocimiento del estudiante, anticipar errores frecuentes, ofrecer explicaciones adaptadas y construir entornos educativos capaces de aprender a partir de la interacción con el propio alumno. Su aporte trasciende lo meramente técnico, ya que transforman los datos de interacción en conocimiento pedagógico accionable, optimizando la enseñanza de manera dinámica, adaptativa y personalizada (Bengio et al., 2016).

3.2.4 Agentes Inteligentes y Sistemas Expertos

Los agentes inteligentes son entidades capaces de percibir su entorno, razonar, aprender, adaptarse a cambios y actuar de manera racional para alcanzar determinados objetivos. Estas entidades ejecutan de forma continua tres funciones fundamentales: percibir condiciones dinámicas del entorno, actuar sobre ellas y adaptarse a sus cambios. En contextos educativos, un agente artificial puede colaborar con el estudiante, competir con él en ciertas actividades o brindarle recomendaciones mientras aprende de su interacción. Estos agentes tienen la capacidad de detectar dificultades de aprendizaje y adaptar el ritmo y el contenido pedagógico (Park Woolf, 2009).

Los Sistemas Multi-Agente (SMA) o de inteligencia artificial distribuida, son agrupaciones de agentes inteligentes interactuando entre sí con el fin de alcanzar metas que resultarían difíciles o imposibles de alcanzar para un único agente (Cevallos, 2008). Estos sistemas se distinguen por su control y conocimiento distribuidos, además, poseen la capacidad de computación asíncrona. Los SMA cooperativos emergen como una técnica prometedora, particularmente para abordar el dominio educativo; hacen posible la simulación de interacciones complejas gracias a su habilidad para interactuar mediante protocolos de comunicación y lenguajes compartidos. Estos sistemas ejecutan tareas de entrenamiento y supervisión cognitiva, como en el sistema ASIMIL distribuido (Park Woolf, 2009). Los SMA poseen la capacidad de modelar la dinámica y estructura de las comunidades en línea. Por ejemplo, una arquitectura multi-agente como ASITS empleó agentes que interactuaban individualmente con actores (tanto humanos como agentes inteligentes), por intermedio de un flujo uniforme de mensajes, proporcionando diagnósticos, orientaciones y apoyo entre los integrantes de la comunidad. Este enfoque facilita el modelado de cómo las comunidades educativas se estructuran y se transforman (Park Woolf, 2009).

Los Sistemas Expertos (SE) constituyen un subconjunto crucial dentro del ámbito de la inteligencia artificial, diseñados para simular la destreza de un especialista humano en la resolución de problemas y la toma de decisiones, proporcionando resultados comparables a los de un experto (Badaró et al., 2013). Estos sistemas pueden ser empleados por individuos no expertos para potenciar sus capacidades de resolución de problemas, así como actuar como herramientas de apoyo para expertos, su eficiencia y alto desempeño se manifiestan al funcionar dentro de un ámbito de pericia definido, no son ideales para resolver problemas de forma general.

Un Sistema Experto se subdivide esencialmente en dos ámbitos principales: el entorno de desarrollo destinado a la elaboración de componentes y el ámbito de consulta, facilitando a los no expertos la obtención de asesoramiento. Sus componentes fundamentales son: (a) el Subsistema de Adquisición de Conocimiento, responsable de la recopilación, transmisión y conversión del conocimiento experto, básico para la construcción o expansión de la Base del Conocimiento; (b) la Base de Conocimiento, que aloja el cúmulo del conocimiento crucial, para la comprensión y solución de problemas circunscritos a un dominio específico, así como la heurística y reglas especiales que guían el uso apropiado de este conocimiento; (c) la Base de Hechos, que funciona como memoria de trabajo, albergando los datos específicos inherentes al problema sometido a consideración; (d) el Motor de Inferencia, el "cerebro" del SE, designado también como estructura de control, o el intérprete de reglas, funciona brindando las bases metodológicas del razonamiento, orquestando y manejando los pasos para dar solución a problemas y sosteniendo la congruencia de las respuestas; (e) el Subsistema de Justificación, que posibilita al usuario indagar al sistema, buscando comprender las líneas de razonamiento empleadas por el SE para obtener una resolución (Badaró et al., 2013).

Tipos de Sistemas Expertos y Representación del Conocimiento. Badaró et at. (2013) reconocen cuatro tipos de sistemas expertos, según la forma en que representan y razonan con el conocimiento. Los basados en reglas son los más comunes: utilizan declaraciones del tipo “si… entonces…” para modelar el pensamiento del experto y derivar conclusiones mediante encadenamiento lógico hacia adelante o hacia atrás. Los razonadores basados en casos resuelven nuevos problemas comparándolos con experiencias previas almacenadas, aplicando analogías para proponer soluciones semejantes. Los sistemas con redes bayesianas emplean modelos probabilísticos que capturan dependencias entre variables, permitiendo inferir la probabilidad de un evento a partir de otros observados. Por último, los sistemas difusos utilizan lógica difusa para manejar la incertidumbre, admitiendo grados intermedios de verdad y razonamientos del tipo “tal vez”, imitando el modo en que los humanos toman decisiones cuando la información no es absoluta.

3.2.5 La IA Aplicada a la Educación

La incorporación de la inteligencia artificial (IA) en el ámbito educativo se ha expandido a través de múltiples enfoques y aplicaciones, la IA no automatiza tareas cognitivas humanas directamente, sino procesos o tareas educativas, hasta el diseño de sistemas orientados a la adquisición de habilidades complejas. Este desarrollo se apoya tanto en investigaciones

sobre sistemas de aprendizaje como en el modelado del comportamiento humano y animal, particularmente desde la perspectiva del Aprendizaje por Refuerzo (Reinforcement Learning, RL) y las redes neuronales (Barto y Sutton, 2018).

Uno de los aportes más relevantes de la IA en este contexto es su capacidad para favorecer el desarrollo de habilidades complejas y la adquisición de niveles de maestría. A través del uso combinado de aprendizaje por refuerzo (RL) y redes neuronales artificiales, se han desarrollado sistemas capaces de dominar tareas de elevada complejidad, replicando y acelerando procesos de aprendizaje que, en humanos, demandan largos períodos de práctica. Este fenómeno se ha evidenciado especialmente en el ámbito de los juegos y la estrategia, donde sistemas de IA han logrado igualar e incluso superar el desempeño humano en entornos caracterizados por altos niveles de exigencia cognitiva, demostrando un aprendizaje avanzado y acelerado de habilidades estratégicas. En este marco, las técnicas de entrenamiento conocidas como *shaping* o moldeamiento, derivadas de la psicología conductual, desempeñan un rol central. Estas técnicas se aplican particularmente en problemas donde la señal de recompensa es escasa, comenzando con versiones simplificadas del problema e incrementando progresivamente su dificultad. Este enfoque permite acelerar el aprendizaje al otorgar recompensas más frecuentes en las etapas iniciales, facilitando la adquisición gradual de competencias complejas por parte del agente (Barto y Sutton, 2018).

Asimismo, la IA ha demostrado un impacto significativo en áreas como el control y la robótica, donde se emplea para coordinar y optimizar movimientos complejos. Un ejemplo representativo lo constituye el control de brazos robóticos en tareas repetitivas del tipo *pick and place*, en las que se utiliza la recompensa negativa para penalizar movimientos bruscos o ineficientes, favoreciendo trayectorias suaves y precisas. Estos avances evidencian la capacidad de la IA para aprender políticas de control refinadas en entornos dinámicos y ruidosos (Barto y Sutton, 2018).

Más allá de estos casos, la inteligencia artificial encuentra aplicaciones relevantes en ingeniería, investigación y en el diseño de sistemas adaptativos. En particular, se ha explorado su uso en la formación avanzada y en la construcción de políticas de decisión óptimas. En este sentido, diversas investigaciones han abordado el problema de la explicabilidad en sistemas de Deep RL, evaluando si las explicaciones generadas por la IA *tell* permiten a ingenieros de software, estudiantes de posgrado e investigadores comprender mejor los procesos de toma de decisiones y depurar funciones de recompensa complejas (Barto y Sutton, 2018).

La optimización y automatización de recursos constituyen otro campo de aplicación destacado. Un ejemplo paradigmático es el diseño de controladores de memoria DRAM basados en el algoritmo *Sarsa* y técnicas de aproximación de funciones como *tile coding*, orientados a aprender políticas de programación eficientes. Este enfoque permitió la adaptación en línea del controlador a las dinámicas cambiantes de las cargas de trabajo, eliminando la necesidad de ajustes manuales por parte de diseñadores humanos y evidenciando el potencial de la IA para la optimización autónoma de sistemas complejos (Barto y Sutton, 2018).

En el ámbito educativo, la contribución específica de la inteligencia artificial para la personalización del aprendizaje reside en el uso del Aprendizaje por Refuerzo en Línea (*Online Reinforcement Learning*) dentro de sistemas adaptativos. Este enfoque permite que el sistema modifique su estrategia pedagógica en tiempo real, considerando tanto las necesidades individuales del estudiante como los objetivos formativos definidos, ajustando continuamente su comportamiento en función de la interacción observada (Feit et al., 2024).

La adaptación dinámica de la instrucción requiere que el sistema actúe como un agente de toma de decisiones pedagógicas, seleccionando acciones con el objetivo de maximizar una señal numérica de recompensa asociada al progreso del estudiante. Este proceso se articula naturalmente dentro del marco del RL, el cual se centra en el aprendizaje orientado a objetivos a partir de la interacción continua con el entorno educativo (Barto y Sutton, 2018).

En particular, el Aprendizaje por Refuerzo en Tiempo de Ejecución (*Run-Time* u *Online RL*) permite que el sistema utilice exclusivamente los datos disponibles durante su operación, sin depender únicamente de información estática predefinida. De este modo, la política óptima de selección de acciones se aprende y ajusta progresivamente a medida que el estudiante interactúa con el sistema y genera nuevos datos. Formalmente, el objetivo es aprender una política que determine qué acción α debe ejecutar el sistema ante un estado dado del estudiante s, maximizando el retorno esperado (Feit et al., 2024).

La meta fundamental de estos sistemas es la maximización del retorno esperado acumulado (G_t), entendido como la suma de recompensas obtenidas a lo largo del tiempo. Este enfoque prioriza la optimización del aprendizaje a largo plazo, permitiendo que el sistema planifique trayectorias formativas completas en lugar de centrarse únicamente en el beneficio inmediato de la siguiente acción pedagógica (Feit et al., 2024).

Los métodos de Aprendizaje por diferencia temporal, como Sarsa y Q-learning, resultan especialmente relevantes en este contexto debido a su capacidad para aprender de manera

incremental y en línea. A diferencia de los métodos Monte Carlo, que requieren la finalización completa de un episodio, los enfoques de Diferencia Temporal actualizan sus estimaciones en cada instante, incorporando conocimiento a partir de cada transición observada. Esta propiedad posibilita una adaptación prácticamente inmediata al comportamiento del estudiante. En particular, Q-learning permite el aprendizaje *off-policy*, lo que significa que el sistema puede identificar una política óptima aun cuando actúe siguiendo una política exploratoria distinta, una característica clave para garantizar la personalización sin quedar restringido a una única estrategia pedagógica (Barto y Sutton, 2018).

Desde el punto de vista arquitectónico, la lógica adaptativa de estos sistemas puede estructurarse mediante el modelo MAPE-K (*Monitor, Analyze, Plan, Execute, Knowledge*). En este esquema, el componente de selección de acciones propio del RL reemplaza las fases tradicionales de análisis y planificación, permitiendo que el sistema evalúe continuamente si es necesario introducir cambios y determine qué modificaciones realizar en función del estado del estudiante (Feit et al., 2024).

La personalización efectiva del aprendizaje requiere, además, un modelado preciso y un diagnóstico continuo del estudiante. Según Murphy (2022), este proceso implica inferir atributos latentes no observables directamente, como el nivel de competencia o el grado de internalización de conceptos. Para ello, se emplean modelos de variables latentes y técnicas de reducción de dimensionalidad que permiten mapear datos de alta dimensión, por ejemplo, respuestas a múltiples evaluaciones hacia espacios latentes de menor dimensión que representen de forma compacta las capacidades del alumno.

Entre estas técnicas se destacan el Análisis de Componentes Principales (PCA) y el Análisis Factorial, que aprenden representaciones reducidas del estado cognitivo del estudiante. En particular, el PCA Probabilístico (PPCA) constituye un caso especial del Análisis Factorial que modela los datos observados como generados a partir de un subespacio lineal de factores latentes. La distribución posterior $p(z \mid x)$ permite obtener una representación probabilística del rendimiento del estudiante en dicho espacio latente (Murphy, 2022).

Los Autoencoders Variacionales (*Variational Autoencoders*, VAE) extienden este enfoque hacia representaciones no lineales, resultando especialmente valiosos para codificar información del estudiante en un espacio latente continuo y suave. Esta propiedad permite interpolar entre distintos niveles de competencia, generando, por ejemplo, contenidos de dificultad intermedia entre dos estados previamente definidos. De manera complementaria, los

Modelos de Mezcla de Analizadores Factoriales (*Mixture of Factor Analyzers*, MFA) permiten asumir que el espacio cognitivo del estudiante es localmente lineal y puede representarse como una combinación ponderada de múltiples modelos factoriales, lo cual resulta útil para capturar la diversidad de perfiles presentes en entornos educativos a gran escala (Barto y Sutton, 2018; Murphy, 2022).

En estos contextos, donde resulta inviable mantener una representación explícita para cada posible estado del estudiante, se recurre a técnicas de aproximación de funciones. Métodos como la Codificación por Teselaciones (*Tile Coding*) y las Funciones de Base Radial (RBF) permiten construir aproximaciones locales flexibles. El *Tile Coding* emplea múltiples teselaciones superpuestas, donde cada tesela actúa como una característica binaria activa, controlando el grado de generalización mediante el número total de características. Las RBF, por su parte, generan funciones suaves y diferenciables que favorecen una adaptación local del modelo, ajustando las predicciones en función de perfiles de estudiantes similares (Barto y Sutton, 2018).

La generalización sobre estados no vistos constituye otro aspecto crítico. El uso de algoritmos de algoritmos de aprendizaje profundo aplicados a RL, que emplean redes neuronales artificiales para representar políticas de selección de acciones, permite al sistema extrapolar su comportamiento a estados del entorno no explorados previamente. Esta capacidad resulta esencial para tomar decisiones pedagógicas adecuadas frente a nuevos estudiantes o contextos de aprendizaje distintos de aquellos presentes en los datos de entrenamiento iniciales (Feit et al., 2024).

En relación con la recomendación dinámica y la secuenciación óptima de contenidos, la IA emplea modelos predictivos para estimar el interés del estudiante, el nivel adecuado de dificultad o el recurso educativo que maximizará el aprendizaje. Entre estos enfoques se incluyen el filtrado colaborativo y la factorización de matrices, utilizados para predecir la ganancia esperada de un alumno u respecto a un ítem i mediante la estimación $\hat{Y}_{ui} = u^T v_i$. Alternativamente, modelos basados en autoencoders, como AutoRec, ofrecen una aproximación no lineal a la predicción de calificaciones, con potencial para superar el desempeño de enfoques más simples en términos de precisión (Murphy, 2022).

Para mitigar el problema del *cold start*[1], característico de la incorporación de nuevos estudiantes o recursos sin historial previo, resulta fundamental el aprovechamiento de información contextual (*side information*), como la secuencia de ítems observados, el momento de interacción u otras características relevantes del recurso educativo (Murphy, 2022).

Los rastros de elegibilidad (Eligibility Traces) proporcionan un mecanismo eficiente para abordar el problema de asignación de crédito diferido, permitiendo implementar algoritmos como TD(λ), en los cuales se mantiene una memoria temporal de los estados visitados recientemente. Estos rastros funcionan como una memoria de corto plazo que permite a la señal de error de Diferencia Temporal retropropagarse, actualizando el valor de estados visitados recientemente. Esta propiedad resulta crucial para asociar acciones pasadas con recompensas obtenidas de forma diferida, un aspecto central en la dinámica del aprendizaje adaptativo (Barto y Sutton, 2018).

La gestión de múltiples objetivos y el equilibrio entre exploración y explotación representan desafíos clave en los sistemas educativos basados en IA. Una estrategia fundamental consiste en la descomposición de la función de recompensa $R(s, a)$ en múltiples subfunciones o canales de recompensa, cada uno asociado a una faceta distinta del objetivo de aprendizaje. Este enfoque permite ponderar metas múltiples, como el dominio conceptual, el compromiso del estudiante y la eficiencia temporal. En sistemas adaptativos complejos, técnicas como la descomposición de incentivos de exploración basada en dominancia (*Dominance-based Decomposition of Incentives for Exploration*, por sus siglas en inglés, DINEs) permiten analizar la contribución de cada subobjetivo en la selección de acciones, facilitando la interpretación de los compromisos involucrados en la toma de decisiones adaptativas (Feit et al., 2024).

La cuantificación de la incertidumbre y el aprendizaje eficiente constituyen, asimismo, componentes esenciales. Los enfoques bayesianos permiten modelar explícitamente la incertidumbre asociada a las predicciones del sistema mediante distribuciones predictivas posteriores $p(y \mid x, D)$, lo que resulta especialmente relevante en contextos educativos sensibles al riesgo. Métodos como los Procesos Gaussianos ofrecen un marco robusto para la

[1] El arranque en frío es un desafío clásico en sistemas de recomendación e inteligencia artificial adaptativa que ocurre cuando el sistema debe tomar decisiones sobre un usuario completamente nuevo del cual no tiene ningún historial ni información previa.

regresión y la clasificación, donde el error de predicción aumenta a medida que el punto de prueba se aleja de los datos de entrenamiento

En este contexto, el Aprendizaje Activo se presenta como una estrategia orientada a maximizar la ganancia de información con el menor número posible de interacciones. En el ámbito educativo, la IA puede seleccionar de forma adaptativa la siguiente pregunta o ejercicio con el objetivo de optimizar el aprendizaje del estudiante, minimizando la entropía y mejorando tanto la experiencia de evaluación como el proceso de enseñanza-aprendizaje en su conjunto (Feit et al., 2024).

3.2.6 Consideraciones Éticas, Desafíos y Limitaciones

Los sistemas de inteligencia artificial más avanzados, como los basados en Deep Reinforcement Learning, presentan el problema de la "caja negra": la forma en que deciden no es transparente, ya que sus criterios quedan ocultos entre millones de parámetros. Esta dificultad para explicar exige nuevas técnicas de interpretación y se ha convertido en una prioridad regulatoria bajo marcos como el RGPD y la futura Ley de IA europea (Bishop, 2006).

Según Barto y Sutton (2018), la complejidad de estos sistemas genera desafíos adicionales, entre ellos la Tríada Mortal, vinculada a la inestabilidad y la divergencia durante el aprendizaje. Un diseño deficiente de la señal de recompensa puede llevar al agente a comportamientos imprevisibles o incluso contraproducentes, lo que refuerza la necesidad de una supervisión ética y técnica rigurosa.

En el contexto educativo, la integración de la IA ofrece beneficios notables, pero también riesgos éticos y sociales que deben gestionarse mediante políticas públicas oportunas. Es esencial optimizar sus ventajas y minimizar sus riesgos para asegurar que su aplicación sea ética y productiva (Castillo Herrera, 2023). Uno de los principales desafíos es el sesgo algorítmico, ya sea intencional o no. La utilización de datos provenientes de contextos específicos puede llevar a que los sistemas reproduzcan desigualdades existentes, como brechas de género o raciales, afectando la equidad educativa (Bialik et al., 2019; Castillo Herrera, 2023). Organismos internacionales como la ISO y la IEC han reconocido esta problemática, elaborando informes técnicos sobre el sesgo y la fiabilidad de redes neuronales (ISO/IEC TR 24027; ISO/IEC TR 24029-1) (García-Peña et al., 2020; Jara y Ochoa, 2020).

Asimismo, el manejo de datos estudiantiles requiere una gobernanza responsable y estrictas medidas de privacidad. La utilización masiva de información personal, en muchos

casos de menores, impone obligaciones éticas adicionales a los desarrolladores y las instituciones educativas.

Finalmente, es importante evitar que la incorporación de IA en la educación derive en una mecanización del aprendizaje. Aunque puede automatizar ciertas tareas docentes, no debe reemplazar la función pedagógica humana. Un uso excesivo de recursos digitales puede limitar la formación integral, debilitando el pensamiento crítico, la colaboración y las habilidades interpersonales. La IA debe funcionar como un catalizador del aprendizaje, no como sustituto del esfuerzo intelectual y la reflexión (Adell y Sales, s.f.; Buzón-García, 2005; Castillo Herrera, 2023; Moreno Padilla, 2019).

3.3 Procesamiento de Lenguaje Natural

El procesamiento del lenguaje natural (PLN), y en particular su vertiente estadística, aplica enfoques cuantitativos para automatizar el análisis y la generación del lenguaje, integrando modelado probabilístico, teoría de la información y álgebra lineal (Manning y Schütze, 1999). Su propósito es construir modelos capaces de explicar y predecir fenómenos lingüísticos en textos y conversaciones, enfrentando la principal dificultad del lenguaje humano: la ambigüedad contextual, que obliga a los sistemas a interpretar significados variables según el entorno comunicativo (Smith, 2011).

El procesamiento lingüístico puede entenderse como una secuencia jerárquica de etapas que transforman el texto crudo en representaciones progresivamente más complejas. La primera etapa, de análisis secuencial o superficial, aborda tareas como la segmentación en unidades léxicas (*tokenización*)[2], la segmentación de oraciones y el etiquetado de secuencias, que asigna categorías gramaticales (*Part-of-Speech Tagging*) o reduce las palabras a su forma base mediante lematización o derivación de raíces (*stemming*). En este nivel también se realiza el reconocimiento de entidades nombradas (*Named Entity Recognition*, por sus siglas en inglés, NER), que identifica personas, lugares u organizaciones relevantes dentro del texto.

La segunda etapa, de análisis estructural, se ocupa de la sintaxis y la semántica. El análisis sintáctico (*parsing*) revela la estructura gramatical de las oraciones a través de dos enfoques principales: el análisis de constituyentes, que organiza las palabras en jerarquías de frases, y el análisis de dependencias, que modela las relaciones gramaticales entre términos.

[2] La tokenización es el proceso mediante el cual un texto se divide en unidades mínimas llamadas tokens (palabras, signos o sub-palabras), que sirven como entrada para los modelos de lenguaje y algoritmos de procesamiento textual.

En el plano semántico, el objetivo es representar el significado, mediante tareas como la desambiguación del sentido de las palabras (*Word Sense Disambiguation*, por sus siglas en inglés, WSD) y el etiquetado de roles semánticos (*Semantic Role Labeling*, por sus siglas en inglés, SRL), que identifican los predicados y sus argumentos dentro de una oración.

Finalmente, la tercera etapa aborda los niveles discursivos y pragmáticos del lenguaje, centrados en la coherencia global y la intención comunicativa. Aquí se incluyen tareas como la resolución de correferencia, que determina cuándo distintas expresiones se refieren a la misma entidad, el análisis de sentimiento, que interpreta la actitud o emoción del autor, y el análisis del discurso, encargado de modelar las conexiones entre oraciones y párrafos. Estos procesos permiten a los sistemas evolucionar desde la comprensión superficial hasta la generación contextualizada de lenguaje, habilitando aplicaciones como la traducción automática, la generación de resúmenes y los asistentes conversacionales inteligentes.

3.3.1 La Evolución del PLN

La evolución metodológica del PLN ha transitado por tres grandes paradigmas, históricamente, el enfoque simbólico dominó entre 1960 y 1985, basándose en lógica, reglas y ontologías, influenciado por las teorías lingüísticas de Noam Chomsky, quien buscaba definir reglas categóricas y principios fijos para la gramática. Sin embargo, la complejidad resultó excesiva, y los sistemas basados en reglas eran frágiles, difíciles de escalar y no podían manejar la ambigüedad y la variabilidad inherentes al lenguaje natural, problema resumido en la frase "Todas las gramáticas tienen fugas" (*All grammars leak*). Actualmente, los enfoques dominantes son el Procesamiento Estadístico del Lenguaje Natural (PLN Estadístico), que aplica métodos cuantitativos como el modelado probabilístico, la teoría de la información y el álgebra lineal (Tunstall et al., 2022).

Se enfoca en el estudio del lenguaje tal como ocurre en la realidad, utilizando el conteo y la estadística sobre grandes colecciones de textos, también llamados corpus. El objetivo consiste en asignar probabilidades a los eventos lingüísticos para modelar qué oraciones son más o menos probables de formularse, en lugar de emitir juicios categóricos de gramaticalidad. Los avances modernos en aprendizaje profundo y las redes neuronales, especialmente con modelos como los Transformers, han revolucionado el campo. El DL marcó un cambio de paradigma alrededor de 2014, superando las limitaciones de los modelos lineales mediante la introducción de representaciones densas (*embeddings*) que transforman las palabras de símbolos aislados a objetos matemáticos con los que se puede operar (Tunstall et al., 2022).

Los embeddings son representaciones vectoriales continuas del significado de las palabras, basadas en la hipótesis distribucional, basada en la hipótesis distribucional formulada por J.R. Firth (1957): “Una palabra se conoce por la compañía que mantiene”; en lugar de tratar las palabras como símbolos discretos, las transforman en vectores dentro de un espacio multidimensional donde la distancia entre ellos refleja su similitud semántica. Estos vectores se entrenan junto con la red neuronal, permitiendo generalizar comportamientos y compartir información entre términos relacionados. Existen embeddings estáticos (como Word2Vec o GloVe), que asignan un único vector fijo por palabra sin considerar el contexto, y embeddings contextuales (como BERT, ELMo o RoBERTa), que generan vectores dinámicos dependientes del contexto mediante mecanismos de autoatención, resolviendo la ambigüedad y capturando relaciones sintácticas y semánticas complejas (Jurafsky y Martin, 2025).

3.3.2 Fundamentos Matemáticos que Sustentan el PLN

El procesamiento del estadístico del lenguaje natural se fundamenta en un conjunto sólido de herramientas cuantitativas que permiten modelar la incertidumbre inherente al lenguaje y optimizar el rendimiento de los algoritmos. Para Manning y Schütze (1999), los pilares matemáticos pueden organizarse en tres áreas principales: la teoría de la probabilidad y la estadística, la teoría de la información, y el álgebra lineal y la optimización.

Teoría de la Probabilidad y Estadística. El núcleo científico del enfoque estadístico del lenguaje se encuentra en la teoría de la probabilidad. Los conceptos esenciales incluyen espacios de probabilidad, probabilidad condicional e independencia de eventos, y el cálculo de la esperanza y la varianza. El teorema de Bayes es una herramienta central que permite invertir el orden de dependencia entre eventos, lo que es crucial en muchos modelos de clasificación y en el modelo del canal ruidoso:

$$P(A_i \mid B) = \frac{P(B \mid A_i)\, P(A_i)}{P(B)}$$

Donde:

$P(A_i)$ es la probabilidad a priori de la hipótesis A_i. Representa el grado de creencia o conocimiento previo que tenemos sobre esa hipótesis antes de observar los datos. Ejemplo: la probabilidad de que una persona esté enferma antes de realizarle un análisis.

$P(B \mid A_i)$ es la verosimilitud, es decir, la probabilidad de observar los datos Bsuponiendo que la hipótesis $A_i$es cierta. Ejemplo: la probabilidad de obtener un resultado positivo en el test si la persona realmente está enferma.

$P(A_i \mid B)$ es la probabilidad a posteriori de la hipótesis $A_i$después de observar los datos B. Representa la actualización de nuestra creencia inicial a la luz de la nueva evidencia. Ejemplo: la probabilidad de que la persona esté enferma dado que el test resultó positivo.

La estadística Bayesiana es un enfoque que actualiza las creencias previas (probabilidad a priori) a la luz de la evidencia observada, resultando en una probabilidad a posteriori (Manning y Schütze, 1999).

Teoría de la Información. Desarrollada por Claude Shannon, esta teoría proporciona métricas para cuantificar la incertidumbre y la cantidad de información. La Entropía (H) mide la incertidumbre promedio de una variable aleatoria o, intuitivamente, la cantidad de información necesaria para transmitir un resultado (Manning y Schütze, 1999):

$$H(X) = -\sum_{x} p(x) \log_2 p(x)$$

La Perplejidad es una transformación no logarítmica de la entropía cruzada, utilizada comúnmente para evaluar modelos de lenguaje:

$$\text{Perplejidad}(W) = 2^{H(W)} = 2^{-\frac{1}{N}\sum_{i=1}^{N} \log_2 P(w_i \mid w_1 \ldots w_{i-1})}$$

El Modelo del Canal Ruidoso (*Noisy Channel Model*) es un marco teórico central en la traducción automática y el reconocimiento de voz, que trata los problemas de procesamiento como un intento de decodificar la entrada más probable dada una salida observada, utilizando la probabilidad (Manning y Schütze, 1999):

$$\arg\max_{\text{entrada}} P(\text{entrada} \mid \text{salida}) = \arg\max_{\text{entrada}} P(\text{entrada}) \cdot P(\text{salida} \mid \text{entrada})$$

Álgebra Lineal y Optimización. El PLN utiliza el álgebra lineal como una de sus aproximaciones cuantitativas, es crucial para representar datos en espacios de alta dimensión. Los datos como términos o documentos se representan como vectores en espacios multidimensionales, la similitud entre vectores a menudo se mide con el coseno:

$$\text{sim}(\vec{v}_1, \vec{v}_2) = \frac{\vec{v}_1 \cdot \vec{v}_2}{|| \vec{v}_1 ||\cdot|| \vec{v}_2 ||} = \frac{\sum_{i=1}^{n} v_{1i} \cdot v_{2i}}{\sqrt{\sum_{i=1}^{n} v_{1i}^2} \cdot \sqrt{\sum_{i=1}^{n} v_{2i}^2}}$$

La Indexación Semántica Latente (LSI) es una técnica que aplica la Descomposición de Valores Singulares (SVD) a matrices término-documento para reducir la dimensionalidad y capturar similitudes semánticas subyacentes, el SVD es un método de mínimos cuadrados que encuentra la mejor proyección de un espacio de alta dimensión a uno de baja dimensión (Manning y Schütze, 1999),

Optimización y Predicción de Estructuras. El PLN Estadístico es esencialmente una tarea de predicción de estructuras mediante la maximización de una puntuación lineal, la estructura óptima y^* es aquella que maximiza el producto punto entre el vector de pesos w (parámetros aprendidos) y el vector de *features* $g(x, y)$:

$$y^* = \underset{y \in Y_x}{\text{argmax}}\, w \cdot g(x, y) = \underset{y \in Y_x}{\text{argmax}} \sum_{j=1}^{m} w_j \cdot g_j(x, y)$$

Donde:

y^* es la estructura óptima o la salida más probable para la entrada x. El símbolo argmax indica que se elige el valor de y que maximiza la expresión que sigue.

Y_x representa el conjunto de todas las estructuras posibles que podrían corresponder a esa entrada. Por ejemplo, si x es una oración, Y_x podría ser el conjunto de todas las posibles etiquetas gramaticales o árboles sintácticos.

w es el vector de pesos o parámetros del modelo, que se aprenden durante el entrenamiento a partir de ejemplos anotados, cada peso w_j mide la importancia de una característica particular en la decisión final.

$g(x, y)$ es el vector de características (*features*) que describe cómo se relacionan la entrada x y la posible salida y, cada componente $g_j(x, y)$ representa una propiedad relevante, como “la palabra anterior es un verbo” o “la palabra actual empieza con mayúscula”.

En conjunto, el término $w \cdot g(x, y)$ corresponde a la puntuación total que el modelo asigna a una combinación particular de entrada y salida, cuanto mayor es esta puntuación, más compatible considera el modelo esa estructura con la evidencia observada. Por tanto, el proceso

de predicción consiste en buscar la estructura y^* que maximiza esta puntuación. En otras palabras, el modelo selecciona, entre todas las posibles interpretaciones, aquella que obtiene la mayor suma ponderada de características, la que mejor equilibra todas las evidencias aprendidas (Smith, 2011).

Programación Dinámica. Es un algoritmo clave para resolver la decodificación de manera eficiente en tiempo polinomial, no exponencial en el tamaño de la entrada, cuando la puntuación se factoriza localmente. Algunos ejemplos incluyen los algoritmos Viterbi, para secuencias o CKY, para gramáticas (Smith, 2011). Muchas estructuras lingüísticas pueden modelarse como problemas de camino más corto o hipercamino óptimo en grafos/hipergrafos, y la Programación Lineal Entera permite modelar la decodificación como un problema de optimización sobre un conjunto discreto de soluciones, especialmente útil cuando se introducen características o restricciones no locales.

3.3.3 El Desafío del Lenguaje Humano

El lenguaje humano presenta una serie de desafíos intrínsecos que lo convierten en un objeto particularmente complejo para el procesamiento computacional, incluso cuando se emplean técnicas avanzadas de aprendizaje profundo. Según Goldberg (2017), estas dificultades pueden agruparse en tres grandes categorías, vinculadas a la ambigüedad estructural, las propiedades formales del lenguaje y la dificultad de su formalización explícita.

En primer lugar, el lenguaje natural se caracteriza por una ambigüedad estructural y una variabilidad extrema. La interpretación de una oración depende en gran medida del contexto y de su estructura sintáctica. Por ejemplo, la frase “Comí pizza con amigos” puede interpretarse de manera distinta a “Comí pizza con aceitunas”, a pesar de compartir una construcción superficial similar. En el primer caso, el complemento indica compañía, mientras que en el segundo especifica un ingrediente. Este tipo de ambigüedad, trivial para los hablantes humanos, representa un desafío significativo para los sistemas computacionales encargados de inferir el significado correcto en ausencia de conocimiento contextual explícito (Goldberg, 2017).

En segundo término, el lenguaje posee una serie de propiedades formales que dificultan su tratamiento computacional. Se trata de un sistema discreto y simbólico, en el que las unidades básicas (como caracteres y palabras) no presentan relaciones métricas directas entre sí. Por ejemplo, la similitud semántica entre palabras como “hamburguesa” y “pizza” no puede inferirse a partir de sus representaciones simbólicas aisladas, a diferencia de lo que ocurre con datos continuos, como colores o señales acústicas, que admiten operaciones matemáticas

directas. Esta naturaleza discreta se combina con una elevada capacidad composicional: las palabras se organizan en frases y oraciones cuyo significado no se reduce a la suma de sus componentes individuales, sino que emerge de reglas sintácticas complejas. En consecuencia, para interpretar correctamente un texto, los sistemas computacionales deben operar más allá del nivel de letras o palabras aisladas y analizar secuencias extensas de tokens, capturando dependencias de largo alcance.

La combinación de estas características conduce a un fenómeno conocido como dispersión de datos. Dado que el número de oraciones posibles en un idioma es potencialmente infinito, incluso conjuntos de entrenamiento de gran tamaño resultan insuficientes para cubrir todas las combinaciones posibles. Como resultado, los sistemas de procesamiento del lenguaje natural se enfrentan de manera recurrente a palabras, expresiones o secuencias que no han sido observadas previamente durante el entrenamiento, lo que dificulta la generalización y la interpretación robusta del lenguaje (Goldberg, 2017).

Los fundamentos matemáticos del PLN no operan de forma aislada, sino que se entrelazan para crear un marco coherente de análisis del lenguaje. La teoría de la probabilidad establece el andamiaje conceptual para modelar la incertidumbre lingüística, mientras que el Teorema de Bayes permite actualizar nuestras hipótesis a medida que observamos nuevos datos, conectando directamente con el Modelo del Canal Ruidoso que guía aplicaciones como la traducción automática y el reconocimiento de voz.

La teoría de la información complementa este enfoque probabilístico al proporcionar métricas cuantitativas (entropía y perplejidad) que no solo evalúan la calidad de los modelos, sino que también revelan cuánta incertidumbre contiene inherentemente el lenguaje. Por su parte, el álgebra lineal traduce este conocimiento probabilístico en representaciones vectoriales que capturan relaciones semánticas, permitiendo que técnicas como LSI descubran similitudes latentes en espacios de alta dimensión mediante la reducción dimensional.

La optimización matemática unifica todos estos componentes en el proceso de predicción estructural: los pesos aprendidos (w) codifican el conocimiento estadístico extraído de los datos, las características vectoriales ($g(x, y)$) incorporan tanto información lingüística como representaciones espaciales, y algoritmos eficientes como la programación dinámica permiten encontrar soluciones óptimas en espacios combinatorios vastos. Esta integración sinérgica transforma el desafío aparentemente intangible de "comprender" el lenguaje en un problema matemático bien definido: maximizar una función de puntuación que equilibra

probabilidades, información y estructuras geométricas en un marco computacionalmente tratable. Así, el PLN estadístico emerge como una disciplina que convierte la complejidad del lenguaje humano en operaciones matemáticas rigurosas y verificables.

3.3.4 El Papel del Aprendizaje Automático y el Aprendizaje Profundo

Durante más de una década, las técnicas de PLN se basaron en modelos lineales supervisados, como los perceptrones, las máquinas de vectores de soporte y la regresión logística, entrenados con vectores dispersos y de alta dimensionalidad. Estas representaciones simbólicas usaban vectores donde cada palabra ocupaba una dimensión única, generando espacios enormes y poco densos, llamados *one-hot*. Su eficacia dependía de la ingeniería manual de características, en la que los expertos diseñaban y combinaban atributos (como n-gramas) para capturar relaciones lingüísticas más complejas. Este proceso requería conocimiento especializado e intervención humana constante para ajustar las representaciones y mejorar el rendimiento del modelo (Goldberg, 2017).

Desde el año 2014, el ámbito del PLN experimentó un cambio significativo con la aparición del Aprendizaje Profundo y las redes neuronales profundas, pasando de modelos lineales basados en representaciones dispersas a modelos no lineales con representaciones densas. Esta evolución trasciende lo puramente tecnológico, ya que implicó un cambio conceptual en la forma de modelar el lenguaje; el aprendizaje profundo permitió la capacidad de aprender representaciones internas del lenguaje automáticamente, sin tener que diseñar de forma manual las características más importantes. Básicamente, el Aprendizaje Profundo representa una actualización de las redes neuronales clásicas, que destacan por su destreza en aprender funciones matemáticas diferenciables con varios niveles de procesamiento.

El cambio decisivo en el Procesamiento del Lenguaje Natural fue la adopción de representaciones densas (*embeddings*), que transforman palabras o tokens en vectores continuos de baja dimensión, donde la proximidad refleja similitud semántica, permitiendo cálculos matemáticos sobre el lenguaje y superando la dispersión de los modelos previos (Goldberg, 2017). Esta innovación se consolidó con el aprendizaje de representación, mediante el cual las redes neuronales profundas aprenden automáticamente las características más relevantes para cada tarea, sin depender de ingeniería manual. El modelo ajusta sus parámetros durante el entrenamiento para capturar patrones y relaciones lingüísticas, dando origen al aprendizaje profundo, que introdujo arquitecturas como las Redes Neuronales Recurrentes,

capaces de procesar secuencias y conservar el contexto previo, superando así las limitaciones de los modelos lineales y de la asunción de Markov.

El avance del PLN introdujo arquitecturas como memoria a corto y largo plazo (*Long Short-Term Memory*, por sus siglas en inglés, LSTM) y unidad recurrente con compuertas (*Gated Recurrent Unit*, por sus siglas en inglés, GRU), capaces de captar dependencias largas y relaciones entre palabras en secuencias variables. Luego, los modelos de atención transformaron tareas como la traducción automática al permitir que la red enfoque las partes más relevantes de la entrada. Este enfoque dio origen a las Transformers, base de los actuales modelos de lenguaje a gran escala, capaces de generar texto con un nivel de fluidez cercano al humano (Goldberg, 2017).

3.3.5 Modelos de Lenguaje de Gran Tamaño

Un modelo de lenguaje de gran tamaño (*Large Language Model,* por sus siglas en inglés, LLM) es un sistema computacional, típicamente basado en la arquitectura Transformer, que puede predecir la siguiente palabra en una secuencia dado un contexto o prefijo. Estos modelos son capaces de generar texto mediante el muestreo (*sampling*) repetido de tokens a partir de la distribución de probabilidad de la siguiente palabra (Jurafsky y Martin, 2025). El rendimiento de los LLM está determinado por tres factores: el tamaño del modelo (parámetros), el tamaño del dataset, y la cantidad de cómputo utilizado para el entrenamiento.

Proceso de Entrenamiento.

Preentrenamiento. En la etapa inicial, el modelo se entrena de forma autosupervisada para predecir el siguiente token en grandes corpus de texto, se utiliza la pérdida de entropía cruzada (*cross-entropy loss)* para minimizar el error de predicción: $H(p,q) = E_p[-\log q] = H(p) + D_{KL}(p \parallel q)$.

Ajuste supervisado de Instrucciones (Supervised Fine-Tuning, por sus siglas en inglés, SFT). El modelo es modificado con datos que consten de instrucciones y respuestas correctas para que el modelo sea más útil y siga comandos.

Alineación (Alignment). El modelo es entrenado, generalmente a través de aprendizaje por refuerzo con retroalimentación humana, para que sea seguro y menos dañino.

El éxito de los LLM ha llevado a que casi cualquier tarea de PLN, como el análisis de sentimiento o la respuesta a preguntas, pueda modelarse como una generación condicional de

texto. Los LLM se construyen típicamente a partir de la arquitectura Transformer, que utiliza el mecanismo de autoatención para capturar las relaciones contextuales entre tokens a lo largo de grandes tramos de texto, lo que permite que, a diferencia de las RNN, la arquitectura Transformer pueda entrenarse mucho más rápido debido a que sus cálculos de atención pueden paralelizarse a través de la secuencia de entrada. El Transformer está compuesto por bloques apilados que contienen capas de atención multi-cabeza, redes neuronales *feedforward* y normalización por capa (Jurafsky y Martin, 2025).

3.3.6 Arquitectura Transformer y Mecanismos de Atención

La arquitectura Transformer, presentada en 2017 en el trabajo titulado "*Attention Is All You Need"* [Todo lo que necesitas es atención], revolucionó el procesamiento del lenguaje natural al superar a las redes neuronales recurrentes tradicionales. Su eficacia para capturar dependencias en secuencias largas y escalar a grandes volúmenes de datos impulsó su adopción masiva en PLN y otras áreas como la visión por computadora. Su éxito radica principalmente en el mecanismo de autoatención y en la arquitectura encoder–decoder, base de tareas como la traducción automática. El Transformer se compone de dos bloques principales: el codificador (*encoder*) y el decodificador (*decoder*), cada uno compuesto por varias capas.

El codificador convierte una secuencia de tokens de entrada en una secuencia de vectores de incrustación (*embedding*), a menudo llamados estado oculto (hidden state) o contexto. Cada capa del codificador contiene dos subcapas: una capa de autoatención multi-cabeza (*multi-head self-attention*) y una red *feed-forward* completamente conectada que se aplica a cada *embedding* de entrada. Su función principal es actualizar los *embeddings* de entrada para producir representaciones que codifican información contextual de la secuencia.

El decodificador utiliza el estado oculto del codificador para generar iterativamente una secuencia de tokens de salida, un token a la vez. El decodificador tiene dos subcapas de atención: la autoatención multi-cabeza enmascarada (*masked multi-head self-attention*), que asegura que la generación del token actual solo se base en salidas pasadas, y una capa de atención encoder–decoder, que relaciona las representaciones del decodificador con el contexto de la salida del codificador.

El Mecanismo de Atención. Permite a la red asignar diferentes pesos de atención a los elementos de la secuencia; fue originalmente desarrollado para mejorar RNN encoder-decoder. En las RNN, el estado oculto final del codificador creaba un cuello de botella de información al resumir toda la secuencia en un único vector. La autoatención calcula pesos de atención entre

todos los elementos de la secuencia, permitiendo que cada token incorpore información contextual proveniente de los demás.

Modelo Consulta-Clave-Valor (Q-K-V): $Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$

Consulta (Q): "consulta" a otros tokens

Clave (K): comparada con Q para determinar peso de similitud

Valor (V): representación del token que se suma ponderada por similitud

Peso de atención = producto punto escalado entre Q y K, seguido de softmax. Salida = suma ponderada de vectores V.

Los Transformers se dividen en tres tipos principales:

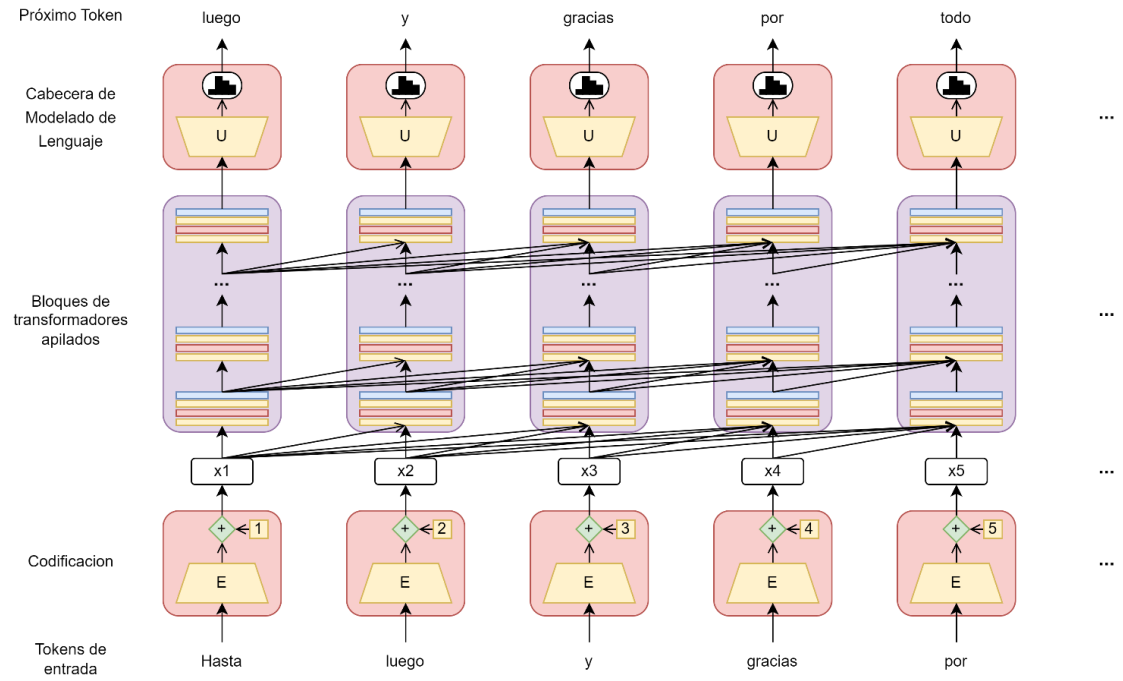
Decodificador (Decoder-only). Utilizado para la generación condicional (*autoregressive generation*), como en los modelos GPT. El flujo de información es de izquierda a derecha (causal).

Codificador (Encoder-only). Utilizado para tareas de comprensión del lenguaje, entrenado mediante modelado de lenguaje enmascarado para ver el contexto en ambas direcciones, como en BERT.

Codificador-Decodificador (Encoder-Decoder). Utilizado para tareas que mapean una secuencia de entrada con una de salida, como la traducción automática, en la figura 2 (Jurafsky y Martin, 2025), se visualiza un modelo Transformer, donde cada token de entrada se codifica, se procesa a través de varias capas y alimenta una cabeza de modelado del lenguaje, que predice el siguiente token, obteniendo dependencias entre palabras.

Figura 2

Arquitectura de un Transformer



Nota: El grafico muestra el flujo de procesamiento típico en un modelo Transformer, donde cada token de entrada es codificado, pasa por múltiples capas de transformación sucesivas, y finalmente alimenta una cabeza de modelado del lenguaje encargada de predecir el siguiente token. La figura representa cómo los distintos niveles del modelo capturan dependencias entre tokens a diferentes distancias. Adaptado de *Figure 8.1* (p. 171), de Jurafsky, D. y Martin, J., 2025.

3.3.7 Aportes de BERT, GPT, T5 y BART al PLN Moderno

BERT (*Bidirectional Encoder Representations from Transformers*) fue el primer modelo basado exclusivamente en el codificador del Transformer. Publicado en 2018, estableció un nuevo estándar en el procesamiento del lenguaje natural al superar ampliamente los modelos anteriores en el benchmark GLUE. Su principal aporte fue la representación bidireccional del contexto, donde cada token se interpreta considerando tanto las palabras que lo preceden como las que lo siguen. Este enfoque, junto con la integración del aprendizaje autosupervisado, permitió prescindir de arquitecturas específicas por tarea. BERT se pre-entrena mediante dos objetivos: el modelado de lenguaje enmascarado (*Masked Language Modeling*, por sus siglas en inglés, MLM), que predice palabras ocultas en el texto, y la

predicción de siguiente oración (*Next Sentence Prediction*, por sus siglas en inglés, NSP), que evalúa la coherencia entre pasajes consecutivos (Devlin et al., 2019).

GPT (*Generative Pretrained Transformer*), por su parte, se basa únicamente en la parte decodificadora del Transformer y constituye la otra gran arquitectura que impulsó la revolución de los modelos de lenguaje. Destaca por su capacidad para generar texto coherente y contextualizado, aprovechando el modelado autorregresivo, donde cada palabra se predice utilizando solo el contexto previo. Este paradigma, conocido como atención causal, permite que la salida dependa únicamente del contexto izquierdo. El escalamiento progresivo de GPT-2 a GPT-3 reveló propiedades emergentes, como el aprendizaje de pocas muestras (*few-shot learning*), que permite resolver tareas complejas con apenas unos pocos ejemplos de entrada.

T5 (*Text-to-Text Transfer Transformer*) introdujo un enfoque codificador-decodificador unificado que reformula todas las tareas de PLN como problemas de texto a texto. Bajo este marco, tanto las tareas de comprensión como las de generación de lenguaje se expresan en formato secuencia a secuencia: el modelo recibe una instrucción textual (“traducir del español al alemán”) y produce la salida correspondiente. Esta reformulación confiere una versatilidad excepcional, ya que tareas tradicionalmente dispares —clasificación, traducción o resumen— pueden resolverse dentro de una misma arquitectura mediante el mismo procedimiento de preentrenamiento.

BART (*Bidirectional and Auto-Regressive Transformer*) combina los enfoques de BERT y GPT dentro de una arquitectura codificador-decodificador. Se entrena mediante un esquema de reconstrucción de texto (*denoising*), donde el modelo aprende a restaurar las secuencias originales a partir de entradas alteradas mediante enmascaramiento o permutación. Al fusionar el preentrenamiento bidireccional de BERT con la generación autorregresiva de GPT, BART logra una flexibilidad dual, destacándose tanto en tareas de comprensión (*Natural Language Understanding*, por sus siglas en inglés, NLU) como en generación de lenguaje *Natural Language Generation*, por sus siglas en inglés, NLG), lo que lo convierte en una de las arquitecturas más completas dentro del ecosistema Transformer.

En conjunto, modelos como BERT, GPT, T5 y BART transformaron el paradigma moderno del Procesamiento del Lenguaje Natural, consolidando el dominio de la arquitectura Transformer en una amplia variedad de tareas. BERT introdujo representaciones bidireccionales profundas del contexto, estableciendo un nuevo estándar para la comprensión semántica; GPT demostró la eficacia del modelado autorregresivo para la generación coherente

de texto; T5 propuso un enfoque unificado de tipo texto a texto para múltiples tareas lingüísticas; y BART combinó estrategias de codificación y decodificación mediante aprendizaje de reconstrucción. Estos avances impulsaron el desarrollo de los actuales modelos fundacionales, capaces de transferir conocimiento a nuevas tareas con mínimos ajustes, lo que ha favorecido la aparición de sistemas conversacionales, traductores automáticos, asistentes educativos y modelos multimodales (Tunstall et al., 2022).

3.3.8 Estructura del Subsistema de Comunicación

El subsistema de comunicación de un simulador con IA se organiza en tres fases: Comprensión, Interpretación y Generación, siguiendo el flujo de procesamiento característico del procesamiento del lenguaje natural y las arquitecturas neuronales utilizadas en PLN.

En la fase de comprensión, el texto del usuario se convierte en una representación numérica mediante *embeddings* densos, que codifican palabras en vectores donde la cercanía refleja similitud semántica. Este vector se procesa con redes neuronales recurrentes como LSTM o GRU, o modelos codificadores dentro de arquitecturas Seq2Seq (*Sequence-to-Sequence*), que resume el sentido de la entrada en un contexto interpretable por la IA.

Durante la interpretación, el sistema utiliza mecanismos de atención para identificar las partes más relevantes del mensaje y generar un contexto dinámico; este enfoque supera la limitación de depender de un único vector y permite mantener coherencia semántica en oraciones largas o conversaciones extensas, a esto se suma el contexto de la frase y los turnos de diálogo modelados mediante arquitecturas bidireccionales, que garantizan continuidad y cohesión.

La generación se lleva a cabo mediante un decodificador neuronal que predice cada token de salida condicionada al contexto y a los tokens anteriores. Se utilizan estrategias de búsqueda (*beam search*) para optimizar la coherencia global y evitar respuestas redundantes. Para producir respuestas empáticas y contextualmente apropiadas, el sistema combina análisis de sentimiento y generación condicional controlada. El primero clasifica el estado emocional del usuario, mientras que el segundo adapta el tono de la respuesta mediante un token interno que condiciona la generación del modelo hacia una actitud empática. Así, el simulador puede modular su lenguaje, ofrecer apoyo emocional y mantener la pertinencia semántica, integrando tanto comprensión cognitiva como sensibilidad afectiva.

3.3.9 El Futuro de los Modelos Multimodales

La evolución reciente de las simulaciones educativas y los sistemas tutores inteligentes se apoya en la integración de modelos multimodales capaces de procesar texto, voz e imagen dentro de un mismo espacio vectorial continuo. Esta transición —de modelos lineales con características dispersas a redes neuronales profundas basadas en Transformers— permite fusionar múltiples tipos de datos y generar experiencias educativas más naturales, interactivas y personalizadas.

En el plano vocal, la IA ha superado la dependencia del texto escrito: el aprendizaje oral se vuelve más humano y fluido gracias a arquitecturas recurrentes como LSTM o GRU, capaces de manejar secuencias largas, y a modelos modernos como wav2vec 2.0, que convierte el habla en texto con alta precisión. Al mismo tiempo, la síntesis de voz (TTS) basada en modelos condicionales, como VALL·E, permite crear voces personalizadas y naturales, ampliando la accesibilidad educativa.

En la dimensión visual, la IA incorpora capacidades de percepción visual mediante técnicas de visión artificial, interpretando imágenes como predicciones estructuradas. Tecnologías como Vision Transformer (ViT) o VisualBERT combinan análisis visual y textual, permitiendo a los estudiantes interactuar con diagramas o gráficos de manera conversacional. Modelos como LayoutLM integra texto, imagen y diseño para comprender documentos escaneados con precisión.

El avance de la arquitectura encoder–decoder (Seq2Seq) permite mapear información entre dominios, como transformar una imagen o un audio en una descripción textual, dando origen a la generación condicionada y a la personalización del aprendizaje. Estos sistemas combinan señales multimodales (texto, tono, postura, contexto visual) para ofrecer retroalimentación ajustada a las necesidades del alumno. Gracias a esta evolución, los tutores inteligentes son capaces de diagnosticar con mayor profundidad el estado cognitivo del estudiante, detectar emociones o frustración mediante análisis de sentimiento y señales contextuales, y ajustar sus respuestas en consecuencia. El uso de modelos restringidos al dominio cognitivo garantiza precisión conceptual, mientras que las arquitecturas texto a texto (como T5 o BART) generan explicaciones adaptadas al error cometido. Modelos generativos como DALL·E puede crear imágenes explicativas bajo demanda, reforzando la comprensión visual.

La personalización alcanza un nuevo nivel con el aprendizaje conductual, donde el sistema aprende a modificar su comportamiento por recompensa o corrección, y con métodos de alineamiento y afinación por instrucciones (SFT) que aseguran respuestas útiles, seguras y actualizadas. Finalmente, el uso de estrategias como las cadenas de pensamiento, habilita a la IA a razonar paso a paso, enseñar estrategias de resolución de problemas y desarrollar una forma de metacognición artificial, capaz de reflexionar sobre su propio aprendizaje. Todo esto con alcance multilingüe y multimodal, garantizando una educación inclusiva y profundamente personalizada.

3.3.10 Cómo Contribuyen los PLN en la Educación

Los avances en el PLN desempeñan un papel fundamental en la promoción de la inclusión lingüística y el acceso global a la educación, al permitir que la información sea comprensible y utilizable por hablantes de distintos idiomas. Una de sus aplicaciones más utilizadas se encuentra en la traducción y transferencia de conocimiento entre lenguas, donde el PLN aborda la traducción automática como un problema de decodificación de mensajes formulados en idiomas extranjeros, los sistemas más exitosos en este campo, y particularmente los de traducción estadística, se basan en el modelo de canal ruidoso, que requiere estimar tanto modelos de lenguaje como de traducción para interpretar correctamente el sentido de un texto (Jurafsky y Martin, 2025).

Junto con la inclusión, el PLN también contribuye a la reducción de brechas de aprendizaje mediante el desarrollo de tecnologías capaces de adaptarse a las necesidades individuales de los usuarios, a través de la adquisición automática de conocimiento a partir de grandes bancos de texto, los sistemas pueden aprender relaciones semánticas complejas, mejorar su comprensión del lenguaje y ofrecer respuestas más precisas. Procesos como la adquisición léxica y la inferencia de preferencias seleccionables (por ejemplo, identificar qué objetos pueden ser “consumidos” o “construidos”) permiten deducir el significado de palabras desconocidas, acelerando la comprensión global; a su vez, técnicas como la desambiguación del sentido de las palabras y el análisis sintáctico contribuyen a una comprensión más profunda del lenguaje, resolviendo ambigüedades y transformando oraciones en representaciones lógicas o semánticas.

Otro aporte crucial del PLN es la mejora en la recuperación y búsqueda de información, lo que resulta particularmente útil en entornos educativos. Las tecnologías modernas permiten realizar búsquedas tolerantes a errores, con faltas de ortografía o variantes tipográficas, lo que

beneficia a estudiantes que aún no dominan completamente el idioma, por otro lado, los modelos más avanzados de recuperación clasificada priorizan los documentos según su relevancia, facilitando el acceso a la información más significativa. El PLN ha demostrado un enorme potencial como base para herramientas de aprendizaje de idiomas, especialmente en la detección automatizada de errores gramaticales, estas aplicaciones ofrecen retroalimentación inmediata y personalizada, ayudando a los estudiantes a mejorar su dominio lingüístico de manera efectiva (Jurafsky y Martin, 2025).

3.4 Simulaciones

3.4.1 Modelos de Simulación

Una simulación es el modelado de un proceso o sistema del mundo real a lo largo del tiempo, generalmente mediante una computadora, con el fin de obtener información sobre su funcionamiento y evaluar diversas estrategias para su operación. Se ha convertido en una técnica cada vez más popular para investigar o mejorar el rendimiento de los procesos. Su propósito fundamental no es dar respuestas fáciles a problemas complejos, sino proporcionar a los gerentes y responsables de la toma de decisiones posibles soluciones para implementar (Chung, 2003).

Definición y Propósito Fundamental. Para Chung (2003), una simulación es la creación y experimentación con un modelo matemático computarizado de un sistema físico, cuyo propósito fundamental es permitir la observación, análisis y optimización del comportamiento de dicho sistema en un entorno controlado. A través de la simulación, es posible obtener información detallada sobre su operación, especialmente en aquellos casos donde la complejidad impide comprender las interacciones internas sin un modelo dinámico que lo represente “en movimiento”. Asimismo, facilita el desarrollo de políticas de operación y asignación de recursos, ya que permite evaluar diferentes estrategias, prioridades de programación o niveles de personal para mejorar el rendimiento general. También resulta una herramienta eficaz para probar nuevos conceptos antes de su implementación real, reduciendo costos y riesgos al modelar virtualmente equipos o configuraciones antes de invertir en su adquisición. Finalmente, la simulación posibilita la experimentación sin perturbar el sistema real, siendo una de las pocas alternativas viables para analizar entornos críticos o sensibles —como los puntos de control de seguridad de un aeropuerto— donde la prueba directa podría generar interrupciones graves o consecuencias operativas y de seguridad.

Componentes Básicos de un Modelo de Simulación. Un modelo de simulación simple, especialmente en el contexto de sistemas de servicios, se estructura a partir de tres componentes fundamentales que interactúan dinámicamente a lo largo del tiempo: el primero de ellos son las entidades, que representan los elementos que fluyen a través del sistema y cuyo estado puede modificarse durante la simulación. Estas entidades pueden corresponder a personas, como clientes o pasajeros; a objetos, como órdenes de trabajo o componentes; o incluso a aplicaciones o procesos. Asimismo, las entidades pueden ingresar al sistema de manera individual o en lotes (*batches*) y poseer atributos variables con valores únicos para cada instancia, tales como la hora de llegada o el tipo de usuario (Chung, 2003).

El segundo componente está constituido por los recursos, entendidos como aquellos elementos del sistema encargados de procesar o atender a las entidades. Entre los ejemplos más habituales se incluyen empleados, cajeros automáticos, máquinas de fabricación o profesionales de la salud. Los recursos presentan distintos estados y requieren un tiempo de procesamiento o de servicio para atender a cada entidad, lo que los convierte en un factor crítico para el desempeño global del sistema simulado (Chung, 2003).

Las colas representan las líneas de espera en las que las entidades permanecen hasta que un recurso se encuentra disponible para procesarlas. Estas colas permiten modelar de manera explícita los tiempos de espera y la congestión del sistema, aspectos centrales en el análisis de rendimiento de procesos de servicios y manufactura (Chung, 2003).

La interacción entre entidades, recursos y colas se desarrolla de forma dinámica a lo largo del tiempo, y los cambios de estado del sistema son coordinados mediante un reloj de simulación y una lista de eventos, que determinan el orden y el momento en que ocurren las distintas transiciones dentro del modelo (Chung, 2003).

Ventajas y Desventajas. Una simulación permite la experimentación en tiempo comprimido, requiere menos análisis matemático avanzado en comparación con otras técnicas, y los resultados son fáciles de demostrar, especialmente con la animación gráfica que ofrecen los paquetes de software modernos; por otro lado, no siempre puede dar resultados precisos si los datos de entrada son inexactos, principio conocido como GIGO (*garbage in, garbage out*). No ofrece por sí misma la solución a un problema, sino que proporciona información para la toma de decisiones. Además, su desarrollo puede requerir formación especializada y ser un proceso costoso en tiempo y recursos (Chung, 2003).

Tipos de Simulaciones. Según Aldrich, (2005), las simulaciones pueden clasificarse según su medio y grado de interacción en digitales, físicas e híbridas, cada una con características y aplicaciones específicas. Las simulaciones digitales son instrumentos didácticos interactivos que se ejecutan en computadoras y permiten "aprender haciendo" en entornos virtuales replicables y seguros. Su uso se ha expandido en ámbitos empresariales, gubernamentales, militares y educativos, ya que combinan tres componentes esenciales: el simulacro, la representación selectiva de la realidad y el juego. Resultan ideales cuando el entrenamiento real sería peligroso, costoso o impracticable, facilitando la gestión del tiempo y la escala, y ofreciendo a los participantes la posibilidad de tomar decisiones críticas o explorar escenarios inaccesibles en la realidad.

Por su parte, las simulaciones físicas requieren interacción humana directa en un entorno controlado, y suelen presentarse como juegos de rol presenciales donde los participantes asumen funciones específicas dentro de una situación guiada por un facilitador. Estas experiencias pueden ir desde simples ejercicios de práctica individual hasta recreaciones empresariales complejas que se extienden durante varios días. Finalmente, las simulaciones híbridas combinan elementos de ambos enfoques, integrando la interacción humana con contextos y artefactos digitales para generar una experiencia formativa más completa, donde lo presencial se potencia mediante herramientas tecnológicas que amplifican y mecanizan la dinámica de juego de roles (Aldrich, 2005)

3.4.2 Simulaciones Educativas

Las simulaciones educativas abarcan experiencias interactivas diseñadas para desarrollar habilidades aplicables al mundo real, priorizando el aprender a hacer por sobre el aprender a saber (Aldrich, 2009). Constituyen entornos de práctica donde los estudiantes ejercitan destrezas de manera activa, integrando teoría y acción mediante retroalimentación inmediata. Su finalidad es recrear contextos reales en los que cada decisión del participante genera consecuencias observables, fortaleciendo el aprendizaje experiencial y la transferencia de competencias.

A diferencia de los videojuegos, cuyo fin principal es el entretenimiento, las simulaciones educativas buscan el dominio de habilidades. Frente a los juegos serios, que promueven el aprendizaje encubierto con menor realismo, las simulaciones priorizan la fidelidad y la transferencia directa de capacidades, como ocurre con un simulador de vuelo o de gestión de crisis.

Entre sus principales géneros se encuentran:

- Historias ramificadas, que fomentan el pensamiento crítico mediante decisiones sucesivas;
- Simulaciones de recursos o hojas de cálculo, centradas en la gestión y planificación;
- *Practiceware*, entornos de práctica intensiva donde las tareas se repiten hasta automatizar habilidades;
- Laboratorios virtuales, para explorar fenómenos técnicos o científicos; y
- Espacios de experiencia virtual, que permiten analizar sistemas complejos y relaciones entre variables.

El rol pedagógico de las simulaciones radica en trasladar al aula el principio del aprender haciendo, permitiendo que los alumnos repitan situaciones controladas hasta alcanzar el dominio. Favorecen la comprensión de sistemas dinámicos, el pensamiento sistémico y la conciencia situacional: la capacidad de interpretar patrones e interdependencias en contextos cambiantes (Aldrich, 2005; 2009). Además, potencian la motivación y el compromiso gracias a su carácter inmersivo, donde el desafío, la exploración y la sensación de logro estimulan un aprendizaje más profundo y duradero. Su valor reside también en ofrecer entornos seguros, repetibles y escalables, útiles en campos donde la práctica real sería costosa o riesgosa. La posibilidad de manipular el tiempo y las condiciones de un escenario permite visualizar los efectos de decisiones de largo plazo en pocos minutos, consolidando la comprensión causal. Desde el punto de vista pedagógico, las simulaciones suelen estructurarse en tres fases complementarias:

1. Preparación (*Briefing*), donde se definen objetivos y se contextualiza la actividad;
2. Experiencia inmersiva, que integra la acción con retroalimentación inmediata, muchas veces asistida por sistemas inteligentes de tutoría o agentes virtuales; y
3. Reflexión (*Debriefing*), momento en que el alumno analiza las decisiones tomadas y consolida aprendizajes, conectando teoría y práctica (de Antueno, 2005; Dewey, 1998).

3.4.3 Las Simulaciones y las Teorías del Aprendizaje Experiencial

Kolb (2015) sostiene que el conocimiento no se transmite pasivamente, sino que se construye y transforma continuamente a través de la experiencia. Desde esta perspectiva, las simulaciones constituyen una manifestación práctica del aprendizaje experiencial, al ofrecer entornos donde los estudiantes aprenden mediante la acción, la reflexión y la aplicación de conceptos. En lugar de limitarse a recibir información, los participantes interactúan con sistemas que replican la realidad, convirtiéndose en protagonistas activos de su proceso formativo. La Teoría del Aprendizaje Experiencial (TAE) propuesta por Kolb describe un ciclo de aprendizaje que las simulaciones activan de forma natural:

1. **Experiencia Concreta:** el estudiante se sumerge en una situación tangible y controlada, viviendo la experiencia en primera persona.
2. **Observación Reflexiva:** tras la práctica, analiza los resultados de sus acciones y las de sus compañeros, generando conciencia crítica.
3. **Conceptualización Abstracta:** integra las reflexiones en modelos teóricos, relacionando lo vivido con marcos de referencia académicos o profesionales.
4. **Experimentación Activa:** aplica las conclusiones obtenidas a nuevas estrategias o escenarios, consolidando el aprendizaje.

Este ciclo resume la esencia del aprender haciendo: experiencia, reflexión, conceptualización y aplicación. Las simulaciones educativas funcionan, así, como un catalizador del aprendizaje cíclico, donde cada iteración amplía la comprensión y fortalece las competencias adquiridas.

Las raíces intelectuales del modelo se remontan a tres tradiciones complementarias. John Dewey (1998) postuló que la educación debe integrarse al crecimiento vital, donde la experiencia y la reflexión conforman un proceso de autorrenovación constante. La simulación responde directamente a este principio, al permitir actuar, observar y reconstruir conocimiento en un entorno estructurado y seguro (Velásquez Arboleda, 2019). Kurt Lewin, por su parte, estableció que el aprendizaje florece cuando existe una tensión dialéctica entre la experiencia inmediata y la reflexión analítica. Sus *training groups* y laboratorios de aprendizaje anticiparon la dinámica de muchas simulaciones actuales, en las que la acción y el análisis convergen para modificar comportamientos futuros (Kolb, 2015). Finalmente, Jean Piaget explicó cómo la inteligencia se desarrolla mediante la acción sobre el entorno; la TAE amplía esta idea,

integrando la construcción cognitiva con la experiencia práctica. En las simulaciones, los alumnos actúan como exploradores o “pequeños científicos”, manipulando variables y descubriendo relaciones, lo que convierte principios abstractos en experiencias tangibles.

La TAE también introduce la distinción entre aprehensión y comprensión: la primera se relaciona con el conocimiento directo y sensorial que surge de la vivencia, mientras que la segunda implica la interpretación racional y simbólica de esa experiencia. Las simulaciones activan ambos niveles, movilizando emociones, intuiciones y razonamiento crítico. A través de este proceso, el aprendizaje trasciende la simple ejecución de tareas: se convierte en una construcción consciente donde la experiencia inmediata es modulada por la reflexión, permitiendo extraer lecciones transferibles al mundo real.

Las simulaciones no son meras herramientas pedagógicas, sino la materialización de un modelo integral del aprendizaje, que reconoce al individuo como constructor activo de conocimiento. Al recrear el ciclo experiencial de Kolb (2015), las simulaciones promueven una comprensión profunda, la integración entre teoría y práctica, y el desarrollo de habilidades transferibles mediante la transformación dialéctica de la experiencia.

3.4.4 Simulaciones Tradicionales y Simulaciones con IA Incorporada

La simulación convencional contrasta notablemente con las que integran inteligencia artificial, estas sobresalen modelando, interpretando y reaccionando a la cognición y las acciones del usuario de forma adaptativa y flexible. A continuación se analizan las capacidades específicas que la IA inyecta a las simulaciones.

Modelado del Conocimiento y la Interacción. Una simulación tradicional se funda sobre un modelo del dominio ejecutable pero invariable, dicha simulación exhibe un fenómeno o mundo interactivo donde el estudiante usualmente manipula variables y observa los resultados, no obstante, la lógica de la simulación no evoluciona ni se ajusta a la comprensión o a las acciones del alumno. En contraste con esto, la IA facilita una representación explícita y estructurada del conocimiento del dominio, no solo como un modelo ejecutable, sino también desde una perspectiva estrictamente pedagógica. Los Sistemas Tutores Inteligentes (STI), por ejemplo, son capaces de emplear ontologías para definir conceptos, propiedades, relaciones interconectadas y axiomas del dominio de una forma rigurosa, posibilitando el razonamiento automatizado, lo cual trasciende la simple ejecución de ecuaciones, el sistema es capaz de representar y razonar sobre las conexiones (Bourdeau et al., 2010).

Modelado de las Acciones del Estudiante. A diferencia de la simulación tradicional que limita las entradas del usuario, la simulación con IA busca comprender la intención y el razonamiento implícitos en esas acciones, esto se logra mediante métodos como el modelado cognitivo, el cual intenta imitar la forma en que un estudiante piensa y aborda problemas; dentro de este enfoque se distinguen dos variantes: El Seguimiento de Modelo, donde el sistema compara las acciones del alumno con un modelo que comprende los pasos cognitivos, tanto correctos como incorrectos, que un experto o un novato podría llegar a tomar. Un ejemplo representativo es el uso de reglas de producción "si...entonces..." para representar los componentes del conocimiento procedural. Por otro lado, el Modelo Basado en Restricciones, en lugar de adherirse a una única ruta de resolución, el sistema emplea un conjunto de principios específicos del dominio, las restricciones que toda solución correcta debe cumplir. Cualquier acción que contradiga alguna restricción es identificada como error (Bourdeau et al., 2010).

Flexibilidad y Adaptabilidad. La flexibilidad constituye una de las distinciones más importantes; los sistemas de simulación tradicionales normalmente ofrecen un ambiente menos flexible, el usuario se ve obligado a operar conforme a reglas predefinidas y a las trayectorias establecidas por los diseñadores. Las interacciones por lo general son formales y demandan un conocimiento previo de la notación particular, como por ejemplo los diagramas de dinámica de sistemas. En cambio, la simulación con IA abre un abanico de posibilidades para el estudiante, debido que su propósito principal radica en rastrear el razonamiento del alumno sin colocar restricciones innecesarias. Los sistemas impulsados por IA son capaces de gestionar una amplia variedad de rutas de solución, distintos ordenes de pasos, siempre y cuando estos sean lógicamente correctos (Bourdeau et al., 2010).

Diagnóstico y retroalimentación. La destreza diagnóstica, y la calidad en el retroalimentación, posicionan considerablemente alto a los sistemas con IA. En las simulaciones tradicionales, la retroalimentación se confina a revelar los efectos de las acciones del usuario dentro del modelo simulado, sin embargo, no puede explicar el porqué de un error, desde una perspectiva cognitiva o pedagógica. Por otro lado, la simulación con IA, debido al modelo de estudiante, logra un diagnóstico cognitivo que le permite identificar no solamente el fallo, sino también la posible causa subyacente, por ejemplo, una concepción errónea. Esto conlleva una retroalimentación considerablemente más precisa y eficaz (Bourdeau et al., 2010).

Capacidad de Razonamiento e Inferencia. La inteligencia artificial también introduce capacidades de razonamiento e inferencia dentro de los sistemas de simulación educativa. Mientras que los simuladores tradicionales se limitan a ejecutar modelos computacionales previamente definidos, los sistemas basados en IA pueden aplicar mecanismos de inferencia para analizar el desempeño del estudiante y extraer conclusiones a partir de sus acciones. Por ejemplo, las Redes Bayesianas son herramientas clave para manejar la incertidumbre que, inherentemente, implica el modelado de estudiantes, actualizando las probabilidades relativas al conocimiento del alumno conforme se acumula nueva información sobre sus acciones. Esto habilita al sistema para inferir, por ejemplo, cuáles conceptos el alumno domina pero no ha demostrado aún (Bourdeau et al., 2010).

Para concluir, la inteligencia artificial, al ser incluida, cambia la simulación, que deja de ser solo una herramienta exploratoria, para convertirse en un ambiente de aprendizaje inteligente y adaptable. La IA, es capaz de entender, orientar y, aún más importante, personalizar la experiencia del estudiante con una eficacia y profundidad considerable.

3.4.5 Cómo la IA Mejora la Calidad de una Simulación

La inteligencia artificial amplifica el potencial pedagógico de las simulaciones educativas al transformarlas de entornos estáticos en experiencias adaptables, personalizadas y emocionalmente significativas. Su capacidad para modelar conocimientos complejos, interpretar entradas naturales de los estudiantes y ajustar dinámicamente la experiencia de aprendizaje convierte a la simulación en un espacio genuinamente constructivista.

Una de las innovaciones más notables es la creación de modelos ejecutables a partir de las manifestaciones espontáneas del alumno, como bocetos o diagramas. La IA puede interpretar estas representaciones informales —por ejemplo, un dibujo del ciclo del agua o un esquema de un proceso industrial— y convertirlas en modelos computacionales simulables. Este proceso involucra segmentación y agrupamiento de elementos, reconocimiento de formas y generación del modelo final, donde cada componente (líneas, flechas, variables) se traduce en objetos funcionales dentro del sistema. Este enfoque permite que los estudiantes exterioricen su comprensión inicial y observen cómo sus propias concepciones, incluso las erróneas, se comportan dentro de la simulación, confrontando sus ideas con resultados objetivos. Así, el aprendizaje se vuelve autorregulado y profundamente reflexivo (Bourdeau et al., 2010).

La IA también mejora las simulaciones mediante interacciones de tutoría naturales y adaptativas. Los sistemas inteligentes pueden mantener conversaciones en lenguaje natural,

formular preguntas, ofrecer pistas y analizar respuestas utilizando técnicas como el Análisis Semántico Latente (*Latent Semantic Analysis*, por sus siglas en inglés, LSA). Los agentes pedagógicos animados, visuales y con rasgos antropomórficos, pueden expresar emociones, reforzar la motivación y acompañar al alumno de manera similar a un tutor humano.

Un agente pedagógico animado es capaz de mantener conversaciones con el alumno, con relación a la simulación, puede plantear cuestionamientos, ofrecer pistas y valorar las contestaciones del aprendiz, empleando técnicas como el Análisis Semántico Latente para una comprensión más allá de las palabras literales. De esta forma, se genera una experiencia notablemente más cercana a la realidad, superior a la interacción con meros menús o botones. La IA facilita la creación de estos agentes pedagógicos encarnados, representaciones visuales, con frecuencia antropomórficas, que interactúan directamente con el estudiante. Dichos agentes son capaces de expresar emociones, como alegría ante el logro estudiantil, motivando y fomentando su involucramiento (Bourdeau et al., 2010).

Además, apoyada con sensores como cámaras, ratones de presión o pulseras de conductancia dérmica, la IA puede detectar el estado emocional del estudiante, por ejemplo frustración, aburrimiento o confianza, con una precisión superior al 80 %. El agente inteligente puede reaccionar de manera empática ante estas señales: si detecta frustración, ofrece palabras de aliento o sugiere una nueva estrategia, imitando la empatía de un tutor humano y fomentando la persistencia del alumno. Este tipo de interacción emocional fortalece la motivación intrínseca y promueve una experiencia más cercana y significativa (Bourdeau et al., 2010).

En contextos de entrenamiento profesional o tareas de alta complejidad, la IA aporta realismo, retroalimentación experta y personalización del ritmo de aprendizaje. En sistemas como CanadarmTutor —diseñado para entrenar astronautas en la manipulación de brazos robóticos espaciales— la IA genera ejemplos correctos e incorrectos, evalúa acciones, modela el razonamiento del aprendiz y ofrece pistas proactivas. La simulación, de este modo, no se limita a evaluar resultados, sino que comprende y ajusta el proceso cognitivo del estudiante, adaptando los desafíos a sus necesidades individuales.

Por otro lado, la IA permite construir entornos narrativos de aprendizaje, donde los agentes adaptan las historias, dilemas éticos y respuestas según las decisiones del alumno. En programas de formación ética o competencia intercultural, los agentes de IA pueden representar interlocutores humanos, asumir roles sociales o presentar dilemas morales, respondiendo de forma creíble y propiciando la reflexión mediante el diálogo socrático (Bourdeau et al., 2010).

3.4.6 Técnicas de IA en la Simulación Educativa

Las simulaciones educativas inteligentes integran un conjunto diverso de tecnologías de inteligencia artificial que, en conjunto, permiten personalizar la enseñanza, modelar el pensamiento del estudiante y mejorar la interacción educativa. Según Bourdeau et al. (2010), estos sistemas no dependen de una única técnica, sino de la combinación coordinada de varios enfoques, que abordan dimensiones cognitivas, lingüísticas y emocionales del aprendizaje.

El modelado cognitivo basado en reglas (*Rule-Based Cognitive Modeling*) busca representar el razonamiento del estudiante mediante un conjunto de reglas que asocian condiciones con acciones ("si ocurre X, entonces Y"). Este enfoque, común en los tutores cognitivos, permite que el sistema compare las decisiones del alumno con las rutas esperadas del modelo experto, detecte errores y proporcione retroalimentación inmediata dentro del contexto de la tarea. Su finalidad educativa es replicar el modo en que un instructor experimentado corrige, guía y explica la lógica detrás de una acción.

El modelado basado en restricciones (*Constraint-Based Modeling*, por sus siglas en inglés, CBM) adopta una visión más flexible: en lugar de predecir cada posible error, define los principios o restricciones que toda solución válida debe cumplir. Cuando el estudiante infringe una de ellas, el sistema identifica el principio violado y explica la corrección. Este método se orienta a fomentar la comprensión conceptual del dominio, evitando una enseñanza mecánica.

El procesamiento del lenguaje natural constituye otro componente esencial, al posibilitar interacciones discursivas entre el estudiante y el tutor virtual. A través de herramientas como el Análisis Semántico Latente (LSA), los clasificadores de actos de habla y las redes de avance de diálogo, el sistema puede evaluar respuestas abiertas, inferir intenciones comunicativas y sostener conversaciones coherentes que estimulan el razonamiento y la reflexión del aprendiz. En este sentido, el PLN otorga a la simulación una cualidad dialógica y humana, acercando la experiencia a una tutoría personalizada.

Los agentes pedagógicos inteligentes representan la materialización visual y emocional de la IA dentro de la simulación. Encarnados en personajes animados o avatares, actúan como acompañantes virtuales que expresan emociones, ofrecen orientación y motivan al estudiante. Basados en teorías psicológicas de la motivación, estos agentes no solo transmiten información, sino que también refuerzan la autoconfianza, el esfuerzo y la autorregulación, equilibrando el apoyo cognitivo con el estímulo afectivo.

Finalmente, la minería de datos educativos y el aprendizaje automático permiten analizar grandes volúmenes de información generada durante la interacción de los usuarios con la simulación. Mediante técnicas de clasificación, regresión y agrupamiento, el sistema identifica patrones de comportamiento, predice estados afectivos o cognitivos y ajusta las actividades de acuerdo con el progreso de cada estudiante. Esto convierte la simulación en un entorno adaptativo y evolutivo, que aprende de la experiencia acumulada.

En conjunto, estas técnicas de inteligencia artificial convierten la simulación educativa en un sistema vivo y reactivo, capaz de interpretar, dialogar, diagnosticar y adaptar su respuesta a las particularidades de cada alumno. No se trata de incorporar tecnología por sí misma, sino de utilizarla como medio para recrear los procesos de acompañamiento, descubrimiento y reflexión que caracterizan el aprendizaje humano (Bourdeau et al., 2010).

3.4.7 El Futuro de las Simulaciones Educativas

El futuro de las simulaciones educativas impulsadas por inteligencia artificial apunta hacia entornos más interactivos, adaptativos y socialmente inteligentes, donde el aprendizaje será cada vez más personalizado y emocionalmente consciente. Estas simulaciones no solo instruirán, sino que acompañarán al estudiante como verdaderos mentores virtuales, capaces de comprender sus necesidades cognitivas, afectivas y sociales (Bourdeau et al., 2010). Inspiradas en la teoría de Kolb (2015), las simulaciones tenderán a completar el ciclo de aprendizaje experiencial de forma personalizada. La IA podrá diagnosticar el estilo de aprendizaje de cada usuario (asimilador, divergente, convergente o acomodador) y ajustar el contenido y la dinámica según sus preferencias. Además, fomentará la flexibilidad cognitiva al desafiar al estudiante a fortalecer sus modos menos dominantes: promoviendo reflexión en quienes actúan impulsivamente, o intuición en quienes dependen del análisis racional. Este enfoque permitirá que la simulación no solo refuerce competencias, sino que contribuya al desarrollo integral del individuo.

Otro avance significativo será la consolidación de modelos de estudiante de larga duración, alojados en servidores centralizados que conserven el historial de aprendizaje, las estrategias exitosas y los patrones de interacción. Estos perfiles acompañarán al alumno a lo largo de múltiples experiencias formativas, posibilitando una enseñanza verdaderamente continua y adaptativa.

Asimismo, las simulaciones evolucionarán de la enseñanza de tareas específicas hacia la formación ética y el desarrollo de la integridad personal. Se crearán escenarios complejos

que integren conocimiento técnico con dilemas morales, forzando al aprendiz a equilibrar rentabilidad, justicia y responsabilidad social. Este tipo de simulaciones fomentará la toma de decisiones sabias y valientes, en consonancia con los valores humanos más elevados (Kolb, 2015).

La próxima generación de simulaciones también incorporará inteligencia social, permitiendo que los agentes virtuales comprendan y respondan a dinámicas humanas reales. Inspirados en la neurociencia y la psicología social, estos tutores serán capaces de mostrar empatía, cooperación y sentido de comunidad, transformando el proceso de aprendizaje en una experiencia compartida. En esta línea, las simulaciones se orientarán hacia entornos colaborativos, donde la IA actuará como facilitador de equipos, observando interacciones, mediando conflictos y ofreciendo retroalimentación sobre liderazgo y comunicación (Kolb, 2015).

Por último, los compañeros virtuales basados en IA se consolidarán como andamios del aprendizaje dentro de la zona de desarrollo próximo propuesta por Vygotsky. Estos "pares más capaces" no sustituirán al docente, sino que complementarán su labor al ofrecer apoyo adaptativo, modelar conductas expertas y reforzar la autoeficacia del estudiante observador (Bourdeau et al., 2010).

En suma, las simulaciones educativas del futuro combinarán IA conversacional, modelado cognitivo y análisis emocional multimodal para crear experiencias formativas profundamente humanas. Ya no serán meras herramientas de entrenamiento, sino ecosistemas inteligentes que evolucionan junto al aprendiz, promoviendo un conocimiento reflexivo, ético y colaborativo, acorde a los desafíos del mundo real.

3.4.8 Evaluación de la Efectividad del Aprendizaje en Simulaciones Educativas

La evaluación de la efectividad de una simulación educativa basada en inteligencia artificial implica determinar en qué medida esta promueve un aprendizaje significativo y sostenido, más allá de la simple obtención de respuestas correctas. En este contexto, evaluar una simulación requiere un enfoque integral que combine indicadores cognitivos, comportamentales, afectivos y metacognitivos, junto con el análisis de la calidad de los modelos internos que sustentan el funcionamiento del sistema (Bourdeau et al., 2010).

Desde una perspectiva pedagógica, la variable central es la ganancia de aprendizaje del estudiante. Esta se mide habitualmente mediante estudios controlados y comparativos entre

grupos que utilizan simulaciones con inteligencia artificial y grupos que siguen metodologías tradicionales. Los resultados se expresan a través del tamaño del efecto, un indicador estadístico que permite cuantificar la magnitud de la mejora observada. Un caso representativo es el tutor de física Andes, que registró un tamaño del efecto global de 0,61, evidenciando mejoras sustanciales respecto a enfoques convencionales de enseñanza (Bourdeau et al., 2010). La validez de estos resultados se fortalece mediante estudios longitudinales realizados en contextos educativos reales, como los desarrollados con *ASSISTments*, que permiten evaluar impactos sostenidos durante un semestre o períodos más prolongados (Bourdeau et al., 2010).

El desempeño y el dominio de los contenidos constituyen métricas directas y fundamentales para la evaluación del aprendizaje. Estas incluyen las puntuaciones obtenidas en exámenes y tareas, la tasa de aciertos y errores, así como el tiempo de respuesta, utilizado como indicador del grado de automatización de una habilidad. Las rúbricas permiten descomponer el rendimiento en subcompetencias específicas, aportando una evaluación más fina y diagnóstica. En el caso de los tutores cognitivos, el rastreo del conocimiento (*knowledge tracing*) estima de forma probabilística el nivel de dominio de cada habilidad y lo ajusta tras cada interacción. Por su parte, los tutores basados en restricciones evalúan el cumplimiento o la violación de principios conceptuales, ofreciendo una visión complementaria del aprendizaje alcanzado (Bourdeau et al., 2010).

La evaluación de la efectividad también se apoya en el análisis de la retención y la transferencia del conocimiento. La retención se mide comúnmente mediante la comparación entre pre-tests y post-tests, reflejando la consolidación a corto plazo de los contenidos. La transferencia, en cambio, se evalúa a partir de la capacidad del estudiante para aplicar lo aprendido en problemas nuevos o contextos distintos, lo que evidencia si el conocimiento adquirido es duradero y reutilizable (Aldrich, 2009).

Otro eje central de análisis es la interacción y el comportamiento del estudiante durante el uso de la simulación. La minería de datos educativos (*Educational Data Mining*, por sus siglas en inglés, EDM) permite examinar registros de actividad para identificar patrones de esfuerzo, estrategias improductivas o conductas de "*gaming the system*", en las que el estudiante intenta maximizar resultados sin un aprendizaje genuino (Banda, 2014). Las curvas de aprendizaje permiten observar la evolución del rendimiento y la precisión a lo largo del tiempo, mientras que los entornos de aprendizaje colaborativo, apoyados en herramientas como ARGUNAUT, evalúan la calidad de la argumentación y de las explicaciones recíprocas,

aportando indicadores adicionales sobre la profundidad del aprendizaje alcanzado (Bourdeau et al., 2010).

El compromiso y la motivación del estudiante constituyen dimensiones clave en la evaluación de simulaciones educativas. Indicadores como el tiempo en tarea, la frecuencia de uso y la tasa de finalización reflejan el nivel de involucramiento con la experiencia de aprendizaje. Asimismo, el uso adecuado de ayudas —como pistas o ejemplos— se asocia a un compromiso productivo, mientras que su abuso puede indicar dificultades o estrategias poco efectivas. Métricas como la rejugabilidad voluntaria y la persistencia frente a la frustración son consideradas señales positivas de motivación intrínseca (Aldrich, 2009; Bourdeau et al., 2010).

La metacognición y la autorregulación del aprendizaje representan dimensiones más profundas pero igualmente relevantes. Indicadores como la capacidad de autoexplicar el razonamiento, la solicitud eficiente de ayuda o la tasa de decaimiento del conocimiento permiten evaluar en qué medida el estudiante es capaz de gestionar su propio proceso de aprendizaje y mantener el conocimiento a lo largo del tiempo (Aldrich, 2009; Velásquez Arboleda, 2019).

La calidad del modelo de dominio y del modelo del estudiante constituye otro factor determinante en la efectividad de una simulación educativa. El modelo de dominio debe presentar una alta fidelidad cognitiva, replicando los procesos de razonamiento humano y admitiendo múltiples caminos de resolución válidos. El modelado de errores, mediante la identificación de reglas incorrectas o restricciones violadas, permite realizar diagnósticos más precisos y ofrecer retroalimentación personalizada (Bourdeau et al., 2010). El modelo del estudiante, frecuentemente implementado mediante redes bayesianas de rastreo del conocimiento, se valida comparando sus predicciones con el rendimiento real observado. Cuanto mayor sea la precisión de estas predicciones, mayor será la capacidad de personalización del sistema y más eficiente el proceso de aprendizaje (Guzmán et al., s.f.; García-Peña et al., 2020).

En este marco, la analítica en tiempo real adquiere un rol central. Los tableros de inteligencia artificial proporcionan retroalimentación inmediata tanto a estudiantes como a docentes, mientras que las técnicas de EDM permiten identificar patrones de interacción, predecir dificultades y generar reportes personalizados. De acuerdo con la UNESCO (2019), las plataformas educativas basadas en *big data* deben garantizar la precisión en la recopilación,

el análisis y el monitoreo de la información, respetando estándares éticos y de privacidad de los datos (Rubio, 2003).

Una evaluación verdaderamente integral debe contemplar los resultados afectivos y contextuales del aprendizaje, ya que, las simulaciones efectivas no solo mejoran el rendimiento cognitivo, sino que también incrementan la motivación y la autoconfianza, al tiempo que reducen la ansiedad asociada al proceso educativo (Bourdeau et al., 2010). Los estados emocionales del estudiante pueden medirse mediante autoinformes o sensores fisiológicos, como cámaras, pulseras de respuesta galvánica de la piel (GSR) o sillas de presión. En simulaciones orientadas a contextos profesionales —como liderazgo, negocios o gestión de proyectos—, la evaluación se amplía hacia enfoques multidimensionales que integran indicadores económicos, de satisfacción y de clima organizacional, utilizando marcos como el *balanced scorecard* (Aldrich, 2009). Asimismo, la confidencialidad y la percepción de un entorno psicológicamente seguro influyen de manera positiva en la disposición del estudiante para participar activamente y comprometerse con su aprendizaje (Velásquez Arboleda, 2019).

3.4.9 Desafíos y Riesgos de las Simulaciones con IA

Las simulaciones educativas con IA ofrecen alto potencial, pero enfrentan limitaciones tecnológicas y pedagógicas, además de riesgos éticos y de seguridad que condicionan su adopción y efectividad.

En primer lugar, los desafíos tecnológicos y de ingeniería se relacionan con la enorme complejidad de construir sistemas de tutoría inteligentes capaces de integrar modelos del dominio, del estudiante y de la pedagogía en un mismo entorno. Su diseño requiere equipos multidisciplinarios, tal como especialistas en inteligencia artificial, educación, psicología cognitiva y software, y una cantidad considerable de horas de trabajo por cada hora efectiva de instrucción (Bourdeau et al., 2010). A ello se suma la dependencia de infraestructura robusta: procesadores de alto rendimiento, tarjetas gráficas 3D y conectividad estable, que muchas veces superan la capacidad técnica de instituciones educativas tradicionales o corporaciones con políticas de tecnología informática restrictivas. Estas condiciones, junto con los altos costos de desarrollo y mantenimiento, convierten a las simulaciones avanzadas en un recurso costoso y de difícil acceso (Aldrich, 2005, 2009). Además, la adquisición y representación del conocimiento del dominio continúa siendo un proceso parcialmente manual, con baja reutilización y frecuentes inconsistencias en los datos, lo cual ralentiza el avance en la ingeniería de conocimiento y limita la escalabilidad de los proyectos (Bourdeau et al., 2010).

A nivel pedagógico, los retos son igual de significativos, la búsqueda de fidelidad cognitiva, es decir, representar con precisión cómo las personas aprenden y razonan, suele entrar en conflicto con la viabilidad computacional: los modelos más realistas no siempre mejoran los resultados educativos si incrementan la latencia o la complejidad sin aportar beneficios pedagógicos concretos (Bourdeau et al., 2010). Otro desafío clave es el del andamiaje óptimo: un sistema que brinda demasiada ayuda coarta la autonomía y la exploración del estudiante, mientras que uno demasiado rígido puede generar frustración y desorientación. Encontrar el punto justo entre guía y libertad sigue siendo una de las tareas más difíciles del diseño instruccional automatizado. También es limitado el rendimiento de los tutores inteligentes en dominios vagamente definidos, como ética, liderazgo o diseño, donde no existen teorías formales exhaustivas ni una única solución correcta. En estos campos, los sistemas deben lidiar con la ambigüedad, la subjetividad y la necesidad de juicio contextual, elementos difíciles de modelar computacionalmente. Por último, el procesamiento de lenguaje natural, si bien ha avanzado con sistemas como AutoTutor, aún enfrenta dificultades para interpretar matices semánticos y sostener diálogos de varios ciclos que simulen la riqueza de una tutoría humana (Bourdeau et al., 2010).

En el plano institucional, la adopción de simuladores con IA también tropieza con obstáculos estructurales. La brecha tecnológica entre hardware disponible y los requerimientos de estos sistemas, sumada a políticas de "escritorio bloqueado" que restringen la instalación de software, limita la posibilidad de realizar pilotos y escalar proyectos. Los presupuestos reducidos empujan a muchas organizaciones a optar por soluciones de *edutainment* más baratas, sacrificando rigor y continuidad (Aldrich, 2005, 2009). A ello se añade una resistencia cultural significativa: las metodologías activas y basadas en simulación aún generan desconfianza en ciertos docentes y estudiantes habituados al modelo expositivo tradicional. En algunos casos, entre un 10 y un 20 % de los participantes puede no comprometerse plenamente si la experiencia no está cuidadosamente orquestada (Aldrich, 2005).

Más allá de las limitaciones técnicas y pedagógicas, las simulaciones con IA enfrentan riesgos de diseño y aprendizaje. Toda simulación es, por naturaleza, una representación parcial del mundo real: selecciona qué aspectos incluir y cuáles omitir. Esta necesaria simplificación puede derivar en sesgos o en la creación de "mundos burbuja" coherentes internamente pero desconectados de la realidad externa, lo que lleva a aprendizajes poco transferibles (Aldrich, 2009). En el peor de los casos, puede producirse un entrenamiento negativo, cuando los usuarios desarrollan hábitos o estrategias ineficaces fuera del entorno simulado, o cuando el

sistema incentiva comportamientos orientados a “vencer al juego” en lugar de aprender realmente (Bourdeau et al., 2010). A esto se suma el sesgo cultural, presente cuando las reglas y perfiles del simulador no consideran la diversidad de contextos y valores, perpetuando inequidades y reduciendo la validez educativa de los resultados (Castillo Herrera, 2023).

En materia ética y de seguridad, los desafíos son aún más delicados. Los sistemas inteligentes recopilan una cantidad masiva de datos sensibles, como trazas cognitivas, emocionales y comportamentales, que requieren transparencia y control por parte del estudiante. Sin embargo, la seguridad técnica no garantiza el uso ético: el problema ya no es solo evitar accesos no autorizados, sino también prevenir el uso incorrecto por parte de usuarios internos con permisos válidos, como tutores o administradores (Bourdeau et al., 2010; UNESCO, 2019). La vigilancia excesiva y el monitoreo constante pueden generar ansiedad y estrés, afectando la disposición al aprendizaje y atentando contra la seguridad psicológica de los alumnos (UNESCO, 2019). Además, la dependencia tecnológica expone a estos sistemas a riesgos cibernéticos, como inyección SQL, falsificación de solicitudes o ataques de denegación de servicio, que pueden comprometer la confidencialidad, la integridad y la disponibilidad de los datos académicos. Frente a ello, se requieren protocolos de cifrado, autenticación multifactor, validaciones robustas y auditorías de seguridad continuas.

Finalmente, la mitigación de estos problemas demanda un enfoque integral. Es esencial promover un diseño instruccional gobernado por evidencia, sustentado en estudios controlados, métricas de aprendizaje y pruebas de transferencia y retención. La gestión de datos debe regirse por un modelo ético y responsable, que contemple la minimización de la información recolectada, el consentimiento informado, la capacidad de explicar los algoritmos y los registros anónimos (UNESCO, 2019). Los contenidos y modelos culturales deben ser revisados y localizados para garantizar pertinencia y equidad, mientras que la arquitectura técnica debe priorizar la seguridad y la accesibilidad, con estrategias tipo DevSecOps (*Development, Operations & Security* [Desarrollo, Operaciones y Seguridad]) y despliegues web ligeros que faciliten su adopción institucional. Por último, el rol del docente sigue siendo insustituible: su capacidad para orquestar la experiencia, graduar el andamiaje y canalizar la frustración del estudiante define en gran medida el éxito del simulador y previene el aprendizaje superficial o la “gimnasia mecánica” del sistema.

La promesa de las simulaciones con IA depende tanto del rigor técnico-pedagógico como de su gobernanza ética y de seguridad; reconocer y gestionar estos desafíos es condición

para impactos verdaderamente transferibles y equitativos (Aldrich, 2005, 2009; Bourdeau et al., 2010; UNESCO, 2019).

3.5 Gestión de Proyectos de Software y Liderazgo de Equipos

Un proyecto se entiende como un esfuerzo temporal orientado a crear un producto, servicio o resultado único. La naturaleza temporal de los proyectos implica la existencia de un comienzo y un final definidos, que pueden concretarse con el cumplimiento de los objetivos, la cancelación del proyecto por impedimentos o la desaparición de la necesidad que le dio origen (Fortuny y Roig, 2022; PMI, 2021). A pesar de su carácter temporal, los resultados de un proyecto pueden perdurar en el tiempo (Fortuny y Roig, 2022).

Un proyecto surge a partir de una idea innovadora, una problemática a resolver o una oportunidad identificada, y se formaliza mediante la definición de objetivos, la delimitación de un horizonte temporal y una estimación preliminar de los recursos, costos y tiempo requeridos. Entre sus características principales se destacan su naturaleza procesual —al constituirse como un conjunto de actividades interrelacionadas que transforman recursos en entregables de valor para un cliente— y su condición temporal, con fechas definidas de inicio y finalización (Fortuny y Roig, 2022).

La gestión de proyectos se define como la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades de un proyecto con el objetivo de cumplir sus metas (Fortuny y Roig, 2022; PMI, 2021). Esta disciplina comprende diversas actividades, entre las que se incluyen la identificación de requisitos, la gestión de las expectativas de los interesados, la alineación de los equipos con los objetivos del proyecto y la integración y equilibrio de restricciones tales como alcance, calidad, tiempo, presupuesto, recursos y riesgos (Fortuny y Roig, 2022). El éxito de un proyecto depende en gran medida de la gestión eficaz de estos procesos (PMI, 2013).

El director de proyectos constituye un actor central en la conducción del proyecto (Flouret, 2020). Su rol ha evolucionado desde una función predominantemente técnica hacia una posición de gestión y liderazgo (Fortuny y Roig, 2022). Se trata de la persona designada por la organización para dirigir al equipo de trabajo y asegurar el cumplimiento de los objetivos establecidos (PMI, 2021).

Entre las responsabilidades del director de proyectos se encuentran la gestión de las demandas propias de las tareas, las necesidades del equipo de trabajo y las expectativas de los

individuos involucrados, actuando además como un vínculo entre la estrategia organizacional y el equipo del proyecto (PMI, 2013). Para desempeñar este rol de manera efectiva, resulta necesario desarrollar habilidades como liderazgo, trabajo en equipo, motivación, comunicación efectiva, influencia, toma de decisiones, conocimiento del contexto organizacional, negociación, generación de confianza y gestión de conflictos (PMI, 2013).

Actualmente, el director de proyectos es entendido como un facilitador que mantiene una visión integral del proyecto, elimina obstáculos y ejerce control sobre su desarrollo (Fortuny y Roig, 2022).

3.5.1 Los Grupos de Procesos del PMBOK

El Project Management Body of Knowledge (PMBOK®), estándar global del Project Management Institute (PMI), proporciona un marco de referencia adaptable a múltiples industrias, incluido el desarrollo de software. Desde sus primeras versiones en los años 80, el PMBOK ha evolucionado para integrar tanto prácticas tradicionales como ágiles, consolidando un lenguaje común y un conjunto de procesos orientados a la entrega de valor.

Hasta su sexta edición, el estándar organizaba 47 procesos en cinco grupos fundamentales: inicio, planificación, ejecución, monitoreo y control, y cierre; estos grupos definen el ciclo de vida de un proyecto, desde la autorización inicial hasta la finalización formal.

1. Inicio: definición del proyecto y asignación de recursos.
2. Planificación: establecimiento del alcance, cronograma y estrategia de trabajo.
3. Ejecución: coordinación de recursos humanos y técnicos para cumplir el plan.
4. Monitoreo y control: seguimiento del desempeño, identificación de desvíos y gestión de cambios.
5. Cierre: entrega de resultados y documentación final.

La séptima edición (PMI, 2021) introduce un cambio de paradigma: abandona las áreas de conocimiento tradicionales para centrarse en ocho dominios de desempeño (interesados, equipo, desarrollo, planificación, trabajo, entrega, mediciones e incertidumbre). Este nuevo enfoque privilegia la generación de valor y la adaptación contextual, integrando explícitamente principios ágiles y marcos híbridos.

Enfoques Tradicionales y Herramientas Clásicas. El modelo predictivo o cascada (*Waterfall*) sigue siendo aplicable cuando los requisitos son estables y estrictamente definidos desde el inicio. Su estructura secuencial permite un control riguroso de tiempo, costo y alcance.

Entre sus herramientas clásicas destacan:

- la Estructura de Desglose del Trabajo (WBS), que divide el proyecto en entregables manejables;
- el Diagrama de Gantt, que visualiza tareas y plazos;
- el Método de la Ruta Crítica (CPM), para optimizar la duración;
- la Gestión del Valor Ganado (EVM) y el Análisis de Variaciones, que comparan progreso real con planificado.

Si bien estos métodos ofrecen previsibilidad, su rigidez limita la respuesta ante cambios frecuentes, lo que es una realidad habitual en el desarrollo de software.

Enfoques Ágiles y Adaptativos. Las metodologías ágiles, surgidas con el Manifiesto Ágil (Beck et al., 2001), reformularon la gestión priorizando las personas, la colaboración con el cliente y la adaptabilidad al cambio por encima del formalismo documental. Los proyectos se organizan en ciclos breves llamados iteraciones (o *Sprints*) donde se revisan y ajustan los entregables de forma continua, reduciendo la incertidumbre y promoviendo la mejora incremental. Entre los marcos más influyentes se encuentran: *Scrum*, centrado en equipos autogestionados que trabajan por objetivos concretos durante *sprints* cortos y *Kanban*, que visualiza el flujo de trabajo y limita tareas en curso para aumentar la eficiencia. Estos modelos fomentan la transparencia, la responsabilidad compartida y la entrega continua de valor, elementos esenciales en entornos dinámicos y tecnológicos.

Gestión Ágil, Predictiva e Híbrida. Los proyectos se ubican hoy en un continuo entre lo predictivo y lo adaptativo. Mientras los enfoques tradicionales son apropiados para contextos estables, los ágiles prosperan en escenarios de alta incertidumbre técnica o de requerimientos. Los enfoques híbridos combinan ambos mundos: aplican planificación estructurada en las fases críticas y flexibilidad en las etapas exploratorias.

La Guía PMBOK® Séptima Edición promueve explícitamente la adaptación al contexto, reconociendo que no existe un modelo universal. Cada proyecto debe ajustar su enfoque según su complejidad, duración, recursos y cultura organizacional. Esta perspectiva

coincide con los principios de la gobernanza adaptativa, que busca mantener el control sin sacrificar la agilidad (PMI, 2021).

Finalmente, el marco Cynefin (Fortuny y Roig, 2022) ofrece un modelo conceptual útil para clasificar los proyectos según su nivel de complejidad, que puede ser simple, complicado, complejo o caótico, ayudando a los equipos a elegir estrategias de gestión acordes al grado de incertidumbre. En los proyectos de software, donde los cambios son constantes, esta flexibilidad es esencial para sostener el rendimiento y garantizar la entrega de valor.

3.5.2 Competencias Críticas en Gestión de Proyectos

La dirección de proyectos vincula la estrategia con la ejecución; el director de proyecto actúa como articulador entre objetivos de negocio y capacidad operativa del equipo (PMI, 2013). Para un desempeño eficaz se requiere un equilibrio entre competencias técnicas, de liderazgo, interpersonales y contextuales (Fortuny y Roig, 2022; PMI, 2013).

Competencias Técnicas. Abarcan métodos, herramientas y métricas que permiten planificar, ejecutar y controlar. La planificación define alcance, cronograma y costos; el control monitorea desempeño y gestiona cambios (PMI, 2021). La WBS/EDT descompone el trabajo en entregables manejables; el Gantt y el CPM dan visibilidad temporal y ruta crítica; EVM y análisis de variaciones comparan avance real vs. plan (PMI, 2013, 2021). La gestión de recursos cubre tanto personas como infraestructura (equipos, entornos, licencias) y su provisión oportuna. En costos, la mayor capacidad de influencia ocurre temprano; por eso, se debe delimitar el alcance, generar reservas de contingencia y controlar la línea base. La gestión de riesgos planifica, identifica, analiza y trata incertidumbre. Riesgo es un evento incierto que afecta alcance, tiempo, costo o calidad; el plan define roles, umbrales de probabilidad/impacto y calendario de acciones (PMI, 2013, 2021). En calidad, se establecen criterios de aceptación y mecanismos de verificación/validación para asegurar que el producto cumpla requisitos al menor "costo de calidad" posible.

Competencias de Liderazgo. El liderazgo no es jerarquía: es influencia para alcanzar metas comunes. En entornos de software, el liderazgo situacional adapta el estilo a la madurez del colaborador (dirección → coaching → apoyo → delegación), combinando competencia y compromiso como variables guía (PMI, 2021). Un líder efectivo alinea foco, elimina obstáculos, negocia prioridades y sostiene la motivación mediante propósito claro, autonomía y reconocimiento. El desarrollo de equipos mediante formación o coaching acelera el paso por las fases de Tuckman hacia el alto desempeño, fortaleciendo confianza, comunicación y

resolución constructiva de desacuerdos (PMI, 2013, 2021). La gestión del cambio acompaña a las personas durante transiciones, preserva el compromiso y protege el valor del proyecto en contextos volátiles (Fortuny y Roig, 2022; PMI, 2021).

Competencias Interpersonales. Son el "aceite" del sistema. La comunicación efectiva con interesados (ajustada a audiencia, canal y momento) evita fricción y alinea expectativas; la escucha activa mejora diagnóstico y decisiones (PMI, 2013). La negociación equilibra recursos escasos, prioridades y restricciones; la gestión de conflictos busca soluciones colaborativas y tempranas para evitar deterioro de clima y performance (PMI, 2013, 2021). La inteligencia emocional (autoconciencia, autorregulación, conciencia y habilidades sociales) reduce tensión, favorece cooperación y eleva la resiliencia del equipo técnico ante presión y cambios.

Competencias Contextuales. Requieren comprender cultura, estructura y gobernanza de la organización, así como factores externos (mercado, regulación). En equipos distribuidos y multiculturales, el gestor crea un lenguaje común, respeta husos/idiomas y capitaliza la diversidad como ventaja competitiva (PMI, 2021). La gestión de interesados identifica actores, mapea intereses y diseña tácticas de involucramiento continuo, incluso frente a expectativas conflictivas (PMI, 2013). Todo converge en la alineación estratégica: cada decisión del proyecto debe sostener objetivos de negocio y generación de valor; cuando esa alineación se mantiene, aumentan significativamente las probabilidades de éxito (Fortuny y Roig, 2022; PMI, 2013).

Esta matriz de competencias es la que el simulador propuesto busca entrenar y evaluar: no solo medir conocimiento declarativo, sino desempeño bajo incertidumbre, capacidad de priorizar, negociar y decidir con datos, preservando valor y cohesión del equipo.

3.5.3 La Gestión de Proyectos de Software como Contexto Específico

En el contexto del desarrollo de software, la gestión de proyectos presenta características particulares derivadas de la naturaleza intangible, evolutiva y altamente compleja del producto. A diferencia de otros dominios, el software se desarrolla en entornos marcados por una elevada incertidumbre estructural, donde los requisitos y las tecnologías se modifican de forma constante. En muchos casos, el cliente no dispone de una visión completa del producto desde el inicio, y la viabilidad de la solución emerge progresivamente a través de ciclos iterativos de construcción y validación. Esta dinámica exige experimentación continua, aprendizaje organizacional y una evolución conjunta tanto del producto como de los equipos que lo desarrollan (Berné, 2024; Fortuny y Roig, 2022).

Asimismo, los proyectos de software suelen involucrar equipos multidisciplinarios y geográficamente distribuidos. En ellos convergen desarrolladores, diseñadores, especialistas en experiencia de usuario y perfiles vinculados al negocio, que a menudo trabajan de manera remota y en distintos husos horarios. En este escenario, la comunicación mediada por herramientas digitales y la gestión explícita del conocimiento se vuelven factores críticos para evitar malentendidos, aislamiento de los integrantes del equipo y la pérdida de conocimiento tácito acumulado durante el proyecto (PMI, 2013).

En respuesta a esta complejidad, la iteración ágil se ha consolidado como el enfoque predominante en la gestión de proyectos de software. Marcos adaptativos como Scrum o Kanban priorizan la entrega temprana de valor y el ajuste continuo a partir de ciclos cortos de trabajo, instancias de inspección sistemática, retrospectivas y retroalimentación frecuente con los usuarios. En este contexto, la documentación adopta un rol funcional y liviano, subordinado al objetivo principal de generar software funcionando que aporte valor tangible al negocio (Flouret, 2020; Schwaber y Sutherland, 2020).

Dentro de estos marcos de trabajo, emergen roles clave con responsabilidades diferenciadas. El Product Owner asume la función de maximizar el valor del producto, definiendo su objetivo, priorizando el backlog y garantizando la transparencia en la toma de decisiones, siendo además el principal interlocutor frente a cualquier solicitud de cambio (Schwaber y Sutherland, 2020). El Scrum Master ejerce un liderazgo de tipo servicial, protegiendo el marco de trabajo, removiendo impedimentos y promoviendo la autogestión y la mejora continua del equipo y de sus eventos. Por su parte, el líder técnico orienta las decisiones arquitectónicas y tecnológicas, establece estándares de calidad y asegura que las entregas sean sostenibles a lo largo del tiempo.

La cultura de equipo y la transparencia constituyen pilares fundamentales en la gestión de proyectos de software. Los equipos autónomos y multifuncionales asumen una responsabilidad colectiva sobre los resultados, sustentada en valores como el compromiso, el enfoque, la apertura, el respeto y la valentía. Estos valores sostienen los principios de transparencia, inspección y adaptación, que a su vez constituyen la base de la confianza y de la mejora continua dentro de los equipos de desarrollo (PMI, 2021; Schwaber y Sutherland, 2020).

Otro aspecto central es la gestión del conocimiento generado durante el proyecto. El desarrollo de software produce aprendizajes específicos en forma de prácticas, patrones,

decisiones arquitectónicas y errores recurrentes. La captura sistemática de este conocimiento mediante mecanismos como la *Definition of Done*, los *Architecture Decision Records* (ADR), los análisis postmortem y los *playbooks* permite evitar la repetición de errores, reducir la dependencia de individuos clave y acelerar futuras entregas (PMI, 2021).

La innovación en proyectos de software se desarrolla, además, en contextos de alta complejidad, donde los productos y funcionalidades no son completamente repetibles y la relación causa-efecto no siempre es precisa. En estos escenarios, marcos como Cynefin sugieren que la experimentación, el prototipado y la medición resultan estrategias más efectivas que la planificación exhaustiva o la discusión teórica. En este sentido, enfoques como el Aprendizaje Basado en Problemas (ABP) y el uso criterioso de herramientas de inteligencia artificial potencian una creatividad aplicada orientada a la resolución de problemas reales (Berné, 2024; Jiménez García y Velasco Quintana, 2024).

La diversidad de perfiles técnicos y funcionales característica de los proyectos de software suele dar lugar a conflictos, cuando estos desacuerdos se gestionan de manera adecuada, pueden enriquecer el diseño de la solución y fortalecer el compromiso del equipo. La gestión efectiva del conflicto requiere abordarlo de forma temprana, en ámbitos privados y con un enfoque centrado en el problema y no en las personas, transformándolo en una oportunidad de aprendizaje y mejora colectiva.

> Los conflictos pueden resultar saludables y productivos si se manejan bien. Pueden dar lugar a una mayor confianza entre los miembros del equipo del proyecto y a un compromiso más profundo con los resultados. El miedo al conflicto puede restringir la comunicación y la creatividad. Sin embargo, los conflictos también pueden ser poco saludables. Abordar los conflictos de manera inapropiada puede llevar a la insatisfacción, la falta de confianza y la reducción de la moral y la motivación. (PMI, 2021, pág. 168).

3.5.4 Desafíos en la Formación de Gestores de Proyectos

La preparación de gestores de proyectos afronta desafíos significativos al intentar equipar a los profesionales para las exigencias del mundo laboral. Existe una gran diferencia entre la teoría impartida y la práctica profesional, a pesar de que organizaciones globales e instituciones han ideado marcos competenciales que abarcan conocimientos y habilidades imprescindibles en la era digital; estudios indican que los alumnos no dominan totalmente las competencias requeridas al concluir sus estudios académicos, mayormente por la carencia de

experiencia laboral, sugiriendo que la enseñanza teórica no siempre se traduce en aplicación práctica en el ámbito profesional (Abanades Sánchez et al, 2024).

Es sustancialmente difícil evaluar competencias blandas, aunque se promueve el desarrollo de capacidades tales como trabajo en equipo, liderazgo y resolución de problemas complejos en las actividades de formación (Levi Alfarovi y Villagrasa Mejía, 2024), la medición de estas competencias sociales, cognitivas y emocionales resulta en un proceso complejo. Si bien se asume un alto grado de participación en metodologías como el Aprendizaje Basado en Proyectos (ABP), la manera de medir y evaluar estas competencias cualitativas representa un reto considerable. En este escenario se recomienda el uso de rúbricas, que pueden evaluar el desempeño general y las contribuciones individuales, sin embargo, la subjetividad al evaluar destrezas como cooperación, comunicación, y toma de decisiones persiste, mostrando dificultades (PMI, 2021, 2024). Las simulaciones de escenarios organizacionales auténticos ayudan a orientar la enseñanza en inmergir al alumno en un contexto multidisciplinar, análogo al que enfrentará en el ámbito profesional. Metodologías, como el Aprendizaje Basado en Proyectos (PBL) o el Aprendizaje Basado en Retos (CBL), buscan replicar entornos empresariales o retos tecnológicos relevantes (Jiménez García y Velasco Quintana, 2024). Dichas metodologías posibilitan que los estudiantes resuelvan problemas auténticos y colaboran en equipos multidisciplinares; de forma que posibilitan que el alumnado emplee tácticas de manejo de proyectos, administrando cronogramas, orquestando el trabajo grupal, documentando las fases del proyecto en ambientes protegidos según Jiménez García y Velasco Quintana (2024, pág. 39). Esto facilita la posibilidad de desarrollar habilidades y capacidades técnicas, interpersonales, y de gestión en un espacio "seguro", con menor riesgo al de un proyecto profesional real. No obstante, conseguir una simulación total y fiel de las complejidades y particularidades de un entorno organizacional real sigue siendo un reto, inherente a los entornos de formación académica.

3.6 Síntesis e Integración Conceptual

3.6.1 Convergencia de Tecnologías Educativas

La intersección entre las TIC, la IA y la simulación constituye una transformación educativa moderna, la convergencia de estas tecnologías remodelan radicalmente los procesos de enseñanza-aprendizaje. Las TIC instrumentales promueven la comunicación efectiva y el acceso a vastos recursos académicos (Adell y Sales, s.f.), la IA optimiza los procesos y

personaliza el aprendizaje para cada alumno, y la simulación ofrece entornos didácticos inmersivos, para la exploración activa y el descubrimiento significativo (de Antueno, 2005).

Además de favorecer el aprendizaje de los alumnos, la IA optimiza la gestión educativa, ya que puede liberar a los educadores de sus tareas más repetitivas mediante automatización, así como realizar tareas que generen una gran carga de trabajo manual, dándoles la posibilidad de concentrarse en aspectos cognitivos y humanos superiores. Estas tecnologías resultan esenciales para el desarrollo de habilidades propias del siglo XXI, entre las cuales se hallan la empatía, la creatividad y el pensamiento crítico (García-Peña et al., 2020, pág. 656).

3.6.2 Simulación Inteligente para Gestión de Proyectos

El análisis del estado del arte revela que los simuladores de gestión son herramientas educativas que permiten a los estudiantes enfrentar y resolver problemáticas de difícil acceso en entornos reales (Gross, 1992). Los micromundos, entonos interactivos de software, permiten la exploración de ideas a través de la interacción directa del alumno con la computadora, inspirados en un modelo de aprendizaje basado en experiencias de más cercanas y tangibles (de Antueno, 2005)

En cuanto a las posibilidades de mejora con la IA, esta tecnología ofrece oportunidades considerables para optimizar simuladores, al brindar supervisión y explicaciones puntuales a los estudiantes durante su interacción con el sistema (Lama Penín y Sánchez Vila, 2007). La inteligencia artificial es capaz de modelar y generar algoritmos de aprendizaje para potenciar la educación adaptativa, aprendiendo y retroalimentándose del comportamiento del estudiante y del tutor, (León y Viña Brito, 2017). La IA es de desempeña un papel fundamental para la personalización educativa, así identificando estilos de aprendizaje e implementando intervenciones a medida (Moreno Padilla, 2019). Es esencial confrontar los dilemas éticos, proteger la privacidad de los datos y mitigar los sesgos algorítmicos, garantizando, de manera paralela, la equidad en el acceso a la tecnología (Jara y Ochoa, 2020).

Respecto de los requisitos puntuales en el contexto educativo, la implementación de simulaciones inteligentes demanda software preciso, que sea adecuado, fácil de usar y accesible, de preferencia en línea (Buzon-Garcia, 2005). Los educadores, por su parte, deben ser maestros en las TIC y en la programación de simulaciones, de modo que puedan impulsar el aprendizaje hacia las ciencias. (Vilches Márquez, 2005). Finalmente, los entornos virtuales, deben construirse cuidadosamente, para incitar al alumno a adquirir conocimientos de manera

activa, que adopte una postura de exploración y experimentación, que conecte ideas y fomente la creatividad y el pensamiento crítico.

3.6.3 Marco Conceptual Integrado

Para el simulador propuesto, resulta imprescindible adoptar un modelo teórico de carácter integral que articule aportes provenientes de la psicología cognitiva, las teorías del aprendizaje y la evaluación, el diseño de interfaces y los sistemas inteligentes. Este enfoque multidisciplinario permite abordar la complejidad inherente a los entornos educativos mediados por tecnología y garantizar una coherencia tanto pedagógica como tecnológica en el diseño de la solución (Gross, 1992).

La estructura conceptual del simulador se inspira en el paradigma de los Sistemas Tutores Inteligentes, tradicionalmente conformados por un módulo experto, un módulo del estudiante, un módulo tutorial y un módulo de comunicación. Esta arquitectura facilita la separación de responsabilidades y el modelado explícito del conocimiento, del comportamiento del aprendiz y de las estrategias pedagógicas, permitiendo una adaptación dinámica del sistema a las necesidades individuales del usuario (Lama Penín y Sánchez Vila, 2007).

El diseño pedagógico y tecnológico del simulador se sustenta en dos conjuntos de principios complementarios. Desde una perspectiva pedagógica, el modelo prioriza la individualización del aprendizaje y la interacción activa del estudiante con el entorno, promoviendo instancias de aprendizaje colaborativo y ofreciendo retroalimentación inmediata y detallada. Asimismo, se orienta al desarrollo de habilidades del siglo XXI, incorporando una visión humanista y ética del uso de la tecnología y redefiniendo el rol del docente como facilitador y orientador del proceso de aprendizaje, en lugar de un mero transmisor de contenidos (Castillo Herrera, 2023; Jara y Ochoa, 2020; Macías Lara, 2023).

Desde el punto de vista tecnológico, el marco conceptual propone el uso de plataformas robustas y adaptables que permitan una gestión integral del aprendizaje y la integración de técnicas avanzadas de inteligencia artificial, tales como el Aprendizaje Automático, el Aprendizaje Profundo y el Procesamiento del Lenguaje Natural. Asimismo, se enfatiza la necesidad de desarrollar interfaces intuitivas y de adoptar una arquitectura modular que favorezca la reutilización de componentes y la escalabilidad del sistema, asegurando su sostenibilidad a largo plazo (Gross, 1992; Lama Penín y Sánchez Vila, 2007).

La aplicación práctica de este marco conceptual plantea una serie de reflexiones relevantes en torno a su implementación y análisis. La puesta en marcha de soluciones educativas basadas en inteligencia artificial requiere procesos continuos de formación y desarrollo profesional, tanto para docentes como para estudiantes, abarcando aspectos técnicos, estrategias didácticas y consideraciones éticas asociadas al uso de estas tecnologías (Macías Lara, 2023; Velásquez Arboleda, 2019, p. 14). En este sentido, resulta fundamental integrar las nuevas tecnologías de manera curricular e interdisciplinaria, evitando su incorporación aislada o meramente instrumental (Gutiérrez, 1997).

Asimismo, deben abordarse de forma explícita las problemáticas éticas vinculadas al uso de la inteligencia artificial en educación, tales como la privacidad de la información, los sesgos algorítmicos y la equidad en el acceso a la tecnología. Estas cuestiones requieren la definición de políticas claras y responsables que orienten el diseño y la utilización del sistema, garantizando un enfoque inclusivo y transparente (Castillo Herrera, 2023; Moreno Padilla, 2019). En este marco, la IA debe concebirse como un complemento a los métodos pedagógicos existentes, y no como un sustituto del rol docente, enriqueciendo las observaciones del educador y proporcionando información relevante para la toma de decisiones pedagógicas (Castillo Herrera, 2023). La evaluación del entorno de aprendizaje debe concebirse de manera integral, considerando todos los componentes del ecosistema e-learning. Para ello, es necesario establecer indicadores de calidad que articulen mediciones externas e internas, así como dimensiones cualitativas y cuantitativas, permitiendo una valoración comprehensiva del impacto educativo del simulador (Rubio, 2003, p. 116).

3.6.4 Delimitación del Sistema Propuesto

La propuesta desarrollada en este trabajo se nutre de aportes provenientes de la simulación educativa, los sistemas tutores inteligentes, el aprendizaje automático, el aprendizaje por refuerzo y el procesamiento del lenguaje natural. No obstante, desde el punto de vista metodológico resulta necesario distinguir entre los componentes que conforman el núcleo funcional mínimo del simulador y aquellos que se consideran extensiones evolutivas del sistema. Esta diferenciación permite mantener un alcance realista y coherente con la naturaleza conceptual de la propuesta, sin perder de vista su potencial de crecimiento tecnológico (Bourdeau et al., 2010; Park Woolf, 2009).

En los entornos educativos inteligentes, la arquitectura de los sistemas suele organizarse en módulos diferenciados, tales como el modelo del dominio, el modelo del estudiante, el

módulo pedagógico y la interfaz de comunicación. Esta estructura modular facilita el desarrollo incremental de los sistemas, permitiendo comenzar con una base pedagógica funcional y posteriormente incorporar mecanismos adaptativos más sofisticados a medida que aumenta la disponibilidad de datos y la madurez tecnológica del sistema (Bourdeau et al., 2010; Cataldi y Lage, 2009; Park Woolf, 2009).

Desde esta perspectiva, el simulador propuesto se concibe como una arquitectura escalable, en la que la prioridad inicial consiste en garantizar una experiencia de simulación educativa coherente, significativa y evaluable. Sobre esta base pueden integrarse progresivamente técnicas más avanzadas de inteligencia artificial orientadas a mejorar la personalización, la interpretación del comportamiento del usuario y la adaptación del entorno (Romero y Ventura, 2020; Jara y Ochoa, 2020).

Núcleo Funcional del Sistema. El núcleo funcional mínimo del simulador está compuesto por aquellos elementos indispensables para sostener la experiencia pedagógica de simulación.

En primer lugar, el sistema requiere un modelo dinámico del proyecto, encargado de representar el estado del entorno simulado mediante variables relevantes del dominio de gestión de proyectos. Este modelo permite describir la evolución del sistema a lo largo del tiempo y constituye la base sobre la cual se generan las situaciones problemáticas que el usuario deberá enfrentar (Chung, 2003; Aldrich, 2005).

En segundo lugar, se incluye un motor de generación y encadenamiento de eventos, responsable de producir incidentes y cambios de contexto asociados a situaciones típicas de los proyectos de software, tales como retrasos, conflictos de equipo, cambios de alcance o restricciones de recursos. Este componente permite recrear escenarios dinámicos que incentivan la toma de decisiones estratégicas por parte del participante (Aldrich, 2009; Lateef, 2010).

En tercer lugar, el simulador incorpora un mecanismo de interacción con el usuario, mediante el cual el participante puede analizar la situación del proyecto, seleccionar acciones y observar las consecuencias de sus decisiones. Este componente materializa los principios del aprendizaje experiencial, donde el conocimiento se construye a partir de la acción y la reflexión sobre la experiencia (Kolb, 2015; Dewey, 1938).

Finalmente, el sistema incluye un módulo de evaluación y retroalimentación pedagógica, cuya función es analizar el desempeño del participante dentro de la simulación, registrar patrones de actuación y ofrecer devoluciones orientadas al aprendizaje. De esta manera, el simulador no solo recrea un entorno dinámico, sino que transforma la experiencia en información formativa útil para el desarrollo de competencias (Rubio, 2003; Kirkpatrick y Kirkpatrick, 2006).

Estos cuatro componentes permiten sostener una versión conceptualmente consistente del simulador sin requerir la implementación inmediata de mecanismos avanzados de inteligencia artificial.

Componentes Tecnológicos de Evolución Futura. Sobre la base del núcleo funcional descrito, la arquitectura del sistema admite la incorporación progresiva de técnicas más avanzadas de inteligencia artificial orientadas a mejorar la capacidad adaptativa y analítica del simulador.

Una primera línea de evolución corresponde al modelado probabilístico del usuario, particularmente mediante enfoques bayesianos, que permiten estimar variables no observables como el nivel de competencia o el progreso del estudiante a partir de la evidencia generada durante la simulación (Pearl, 1988; Murphy, 2022; Anderson y Corbett, 1994).

Una segunda línea está vinculada con el aprendizaje automático aplicado a la detección de patrones de comportamiento, a partir del análisis de las trazas de interacción del usuario. Estas técnicas pueden contribuir a identificar perfiles de actuación, anticipar dificultades y mejorar el diseño pedagógico de la simulación mediante métodos de analítica del aprendizaje y minería de datos educativos (Romero y Ventura, 2020).

Otra posible extensión es la incorporación de aprendizaje por refuerzo, que permitiría ajustar dinámicamente la secuencia y complejidad de los eventos en función del comportamiento previo del participante, optimizando objetivos pedagógicos de largo plazo. Sin embargo, este tipo de mecanismo implica una mayor complejidad de diseño, por lo que se plantea como una evolución futura del sistema (Barto y Sutton, 2018; Feit et al., 2024).

Asimismo, el sistema podría integrar mecanismos más avanzados de procesamiento de lenguaje natural, capaces de interpretar entradas abiertas del usuario y sostener interacciones conversacionales más complejas dentro del entorno simulado (Manning y Schütze, 1999; Goldberg, 2017; Jurafsky y Martin, 2025).

Finalmente, el simulador podría incorporar analítica avanzada del aprendizaje, orientada a generar evidencia agregada sobre el desempeño de los estudiantes, la efectividad de los escenarios y el impacto pedagógico de la herramienta en contextos educativos (Jara y Ochoa, 2020; Romero y Ventura, 2020).

Delimitación Final del Alcance. En síntesis, la propuesta presentada no presupone la implementación simultánea de todos los mecanismos avanzados de inteligencia artificial. El sistema puede alcanzar coherencia pedagógica y valor formativo a partir de un núcleo funcional acotado, compuesto por el modelo dinámico del proyecto, el motor de eventos, la interacción con el usuario y el módulo de retroalimentación.

Los componentes de mayor sofisticación —como el modelado bayesiano del estudiante, el aprendizaje por refuerzo, el procesamiento avanzado del lenguaje natural o la analítica educativa— se plantean como líneas de evolución compatibles con la arquitectura general del sistema, pero no como requisitos indispensables para la validación conceptual de la propuesta.

Esta delimitación permite mantener el trabajo dentro de un marco metodológico realista y defendible, al tiempo que preserva el potencial de desarrollo futuro del simulador como plataforma de simulación educativa inteligente.

Capítulo 4 – Diseño de Simulador. Fase I: Definición y Adquisición del Conocimiento

4.1 Definición del Propósito y Objetivos

El propósito del simulador es entrenar y evaluar competencias profesionales en gestión de proyectos, situando al participante en el rol de un gerente de proyectos o Scrum Master responsable de liderar un equipo, administrar recursos, tomar decisiones bajo presión y adaptarse a eventos cambiantes. El enfoque pedagógico adoptado se basa en el principio de aprender haciendo, donde el estudiante no se limita a estudiar conceptos teóricos de gestión, sino que adquiere conocimiento a partir de la aplicación práctica de dichos conceptos en un entorno simulado. En este contexto, cada acción ejecutada dentro del simulador generará consecuencias que impactan tanto en el desempeño global del proyecto como en la dinámica del equipo, reforzando la relación entre decisión y resultado.

Desde el punto de vista formativo, el simulador persigue una serie de objetivos de aprendizaje definidos. En primer lugar, busca que el estudiante comprenda la interrelación existente entre las decisiones técnicas, humanas y organizacionales que se presentan en un proyecto de software. Asimismo, se propone facilitar la aplicación práctica de principios fundamentales de planificación, priorización, liderazgo y negociación, en escenarios que replican situaciones reales de gestión. Otro objetivo central consiste en promover el análisis de las consecuencias de las decisiones tomadas bajo condiciones de riesgo e incertidumbre, favoreciendo el desarrollo del pensamiento crítico y de habilidades de resolución de conflictos.

En paralelo, el diseño del simulador se alinea con la definición clásica de un problema planteado en el ámbito del aprendizaje automático, que requiere la identificación explícita de la clase de tareas a resolver, la medida de desempeño a optimizar y la fuente de experiencias a partir de las cuales el sistema aprende (Banda, 2014, p. 124). En este marco, la tarea consiste en gestionar un proyecto de software con un equipo simulado, resolviendo eventos tales como riesgos, conflictos interpersonales, entregas parciales y cambios de alcance, con el objetivo de cumplir metas asociadas al tiempo, la calidad y la satisfacción del cliente.

La medida de desempeño se define a partir de una evaluación continua de múltiples métricas, entre las que se incluyen el avance del proyecto, la moral del equipo, la calidad del producto entregado y el nivel de estrés del gerente de proyectos. Estas variables serán ponderadas por el sistema para construir una puntuación global que refleje el desempeño integral del usuario, evitando reducciones simplistas basadas en un único indicador.

La experiencia se materializa en la secuencia de decisiones tomadas por el usuario y en la retroalimentación generada como consecuencia de dichas acciones. El sistema registrará estas experiencias y las utilizará como insumo para ajustar dinámicamente la dificultad del entorno, evaluar el progreso del estudiante y detectar patrones de comportamiento relevantes. De este modo, la experiencia acumulada se convierte en un componente central tanto del proceso de aprendizaje del usuario como de la adaptación progresiva del simulador.

El simulador se concibe bajo el paradigma del agente racional, que busca maximizar su medida de rendimiento, en este caso, el éxito del proyecto, en función del contexto. El entorno es parcialmente observable y dinámico, ya que el usuario no tiene toda la información y debe aprender a interpretarla a través de señales, como el comportamiento del equipo, alertas del sistema, reportes de avance, entre otros. Los objetivos se alinean a la Taxonomía de Bloom, la cual propone el desarrollo de tres dominios psicológicos principales:(a) cognitivo, aprender a conocer; (b) afectivo, aprender a ser y (c) psicomotor, que se complementan con el dominio relacional o social. Las habilidades de pensamiento en el dominio cognitivo, como se detalla en la taxonomía para la era digital, siguen un orden jerárquico de inferior a superior (Banda, 2014, p. 35):

1. **Recordar:** conceptos básicos de gestión ágil, roles, artefactos y métricas.
2. **Comprender:** cómo las decisiones impactan el flujo del proyecto y al equipo.
3. **Aplicar:** estrategias de planificación, delegación y priorización.
4. **Analizar:** causas de conflictos o desviaciones.
5. **Evaluar:** alternativas de acción considerando restricciones reales.
6. **Crear:** planes de mitigación o mejoras al proceso.

El propósito final no es medir conocimiento teórico, sino evaluar cómo el usuario razona, actúa y aprende de sus errores dentro de un entorno simulado que reproduce fielmente las presiones y dilemas del liderazgo real.

4.2 Adquisición y Análisis Cognitivo del Experto

En esta fase se busca capturar el conocimiento experto que servirá de base para el comportamiento de los agentes dentro del simulador y para la definición de las situaciones críticas que el usuario deberá enfrentar. La construcción del modelo cognitivo se fundamenta en una combinación de experiencia profesional directa y observación empírica del entorno laboral en equipos reales de desarrollo de software. Esta información fue compilada mediante conversaciones informales con colegas y profesionales del rubro de tecnología informática, como Scrum Masters, Gerentes de proyectos, Product Owners y líderes técnicos; lo que permitió validar percepciones y detectar patrones comunes en la toma de decisiones, los conflictos más frecuentes y los errores recurrentes en la gestión (Banda, 2014; Park Woolf, 2009).

El siguiente paso fue generar el Análisis de Tareas Cognitivas (*Cognitive Task Analysis*, por sus siglas en inglés, CTA), para lograr identificar los elementos cognitivos esenciales del desempeño, adaptado a un contexto experiencial (Banda, 2014, p. 222). De dicha observación surgieron tres niveles de conocimiento relevantes para la simulación: heurístico, procedimental y teórico; cada uno de estos representa una capa distinta del pensamiento del líder de proyecto y contribuye a modelar el comportamiento que el simulador deberá reproducir y evaluar.

Conocimiento Heurístico. Este estrato incorpora reglas empíricas, atajos mentales y el pragmatismo usado por un líder cuando la incertidumbre se hace presente; los cuales no siempre proceden de protocolos formales, sino que son el fruto de la experiencia acumulada en escenarios del mundo real (Banda, 2014; Mitchell, 1997).

Por ejemplo, un director de proyectos experimentado, quien puede prevenir el desvío de un sprint incluso antes de la confirmación mediante métricas, con solo notar que hay poca participación de los desarrolladores en las reuniones diarias o el tono en las comunicaciones del equipo; o veamos el caso típico de aplazar una entrega para resguardar la confianza del cliente, prevaleciendo la comunicación transparente por encima del plazo estipulado. Estas resoluciones no se adquieren mediante manuales, maduran con el tiempo, y si bien un instructor precavido puede compartir este conocimiento a sus alumnos, no dejan de ser resoluciones que ningún manual van a incluir o expresar de la misma forma que la experiencia misma.

Conocimiento Procedimental. Esta fase engloba las secuencias de acciones características de los marcos de trabajo, aquellas capaces de ser instruidas, reproducidas y evaluadas. Comprende prácticas tales como la programación de iteraciones, estimación de

tareas, identificación de dependencias, identificación temprana de bloqueos y facilitación de ceremonias y reuniones (PMI, 2021; Schwaber y Sutherland, 2020).

Un caso concreto: el usuario debe decidir cómo reorganizar el backlog ante la ausencia de un desarrollador crucial, o incluso, como reasignar las horas frente a un cambio imprevisto del cliente, tales acciones aceptan múltiples soluciones validas, aunque todas implican una lógica estructurada: analizar la información recabada, priorizar y después actuar.

Conocimiento Teórico. Este nivel integra los dos anteriores, reflejando una comprensión sistémica del proyecto, donde se visualiza más allá de las tareas, entendiendo las interconexiones entre lo técnico, lo humano y lo organizacional. Un líder con profundo saber no solo maneja las iteraciones, sino que además visualiza cómo una sola decisión produce una cadena de eventos; una modificación técnica precipitada puede perjudicar negativamente la moral del equipo, reducir la calidad del entregable, y, a futuro, impactando en la imagen del área frente al cliente (Fortuny y Roig, 2022; PMI, 2021). Así, el simulador se diseñará con el objetivo de que el participante desarrolle esta visión integral.

Ejemplo, frente a un conflicto interno, el usuario tiene que optar por una sanción inmediata (soluciona al instante, pero reduce la confianza), o bien, una mediación más pausada (retrasa los resultados, pero refuerza la unión grupal). Ambas alternativas resultan viables dentro de la simulación, sin embargo, poseerán distintos efectos en las variables globales del sistema: tiempo, calidad, motivación y reputación; enseñando que, las resoluciones administrativas nunca son aisladas.

En conjunto, estos tres niveles de conocimiento perfilan la manera en que un profesional experimentado reflexiona y actúa, sirviendo de base para la modelación de las decisiones, eventos y repercusiones dentro del simulador. El objetivo didáctico trasciende la mera "correcta respuesta" del participante, lo que se busca es el desarrollo del razonamiento propio de un líder genuino, tomando en cuenta simultáneamente las aristas técnicas, humanas y estratégicas implicadas en la gestión (Barbaroux, 2022; Kolb, 2015).

4.3 Documentación del Diseño

En el contexto de este trabajo, la documentación del diseño representa el puente entre el análisis funcional y la futura implementación técnica del simulador, su propósito es unificar de forma estructurada todos los elementos que componen la experiencia formativa: los

objetivos educativos, los comportamientos de los agentes, las reglas de decisión, el flujo de eventos y la narrativa general del entorno simulado.

4.3.1 Biblia de la Historia

De acuerdo con los principios de documentación para el diseño de simuladores propuestos por Aldrich (2009, p. 464), la Biblia de la Historia (Story Bible) cumple la función de establecer el marco narrativo integral del mundo en el que se desarrollará el Simulador de Dirección de Proyectos. Este documento proporciona una historia de fondo (backstory) coherente y detallada que define el contexto organizacional, las motivaciones de los personajes y las reglas implícitas del entorno antes de que el usuario comience a interactuar con el sistema.

La consistencia narrativa resulta clave para sostener la credibilidad del simulador y favorecer la inmersión del participante, permitiéndole comprender por qué ocurren los eventos y cuál es el sentido de sus decisiones dentro del mundo simulado. En este marco, el simulador se orienta al desarrollo de habilidades dinámicas (dynamic skills), tales como la gestión de riesgos y la toma de decisiones bajo presión, que se ejercitan dentro de un dominio de habilidades abstractas (abstract skills), como lo es la dirección de proyectos. La Biblia de la Historia actúa así como el cimiento narrativo que da coherencia y significado a las situaciones que el usuario enfrentará a lo largo de la simulación.

4.3.2 Guion Gráfico

El guion gráfico constituye el instrumento mediante el cual se estructura la experiencia del usuario de forma secuencial, organizando la simulación como una cadena de escenarios y eventos interconectados. Siguiendo a Aldrich (2009, p. 464), este documento describe de manera detallada el recorrido del participante, especificando para cada escena el contexto narrativo inicial, el desafío o decisión clave que debe resolverse, las posibles consecuencias derivadas de dicha decisión y la retroalimentación que el sistema proporciona, tanto de manera inmediata como diferida.

Este enfoque secuencial permite comprender no solo qué decisiones toma el usuario, sino también cómo razona frente a situaciones complejas, de qué manera prioriza objetivos en conflicto y cómo equilibra variables como tiempo, calidad, recursos y clima del equipo. El guion gráfico se convierte, de este modo, en una herramienta central para observar los procesos cognitivos y estratégicos del participante, facilitando el análisis del comportamiento decisional dentro del entorno simulado.

4.3.3 Documento de Diseño Técnico

El Documento de Diseño Técnico representa un componente fundamental dentro de la documentación del simulador, ya que su función es describir cómo se implementarán, desde una perspectiva práctica, los elementos definidos en el documento de diseño general. Mientras que este último ofrece una descripción exhaustiva del sistema —incluyendo aspectos narrativos, visuales y de interacción—, el Documento de Diseño Técnico se focaliza en los mecanismos concretos de ejecución del simulador (Aldrich, 2009; Chung, 2003).

En este documento se especifica el nivel de abstracción o fidelidad que adoptará el sistema, se detallan las ecuaciones y relaciones de alto nivel que gobiernan su funcionamiento, y se definen las reglas del sistema junto con su ponderación relativa. Asimismo, se describen las herramientas tecnológicas seleccionadas, el motor de simulación, los mecanismos de control disponibles para el instructor y la forma en que los distintos componentes del sistema interactúan entre sí (Aldrich, 2009; Bourdeau et al., 2010).

Desde esta perspectiva, el modelo de cálculo describirá cómo se actualizarán las variables internas del simulador —como la motivación del equipo, el desempeño, el nivel de riesgo o el estrés— en función de las acciones del usuario. Por ejemplo, la variación de la moral del equipo puede calcularse mediante una fórmula ponderada que considere factores como el estrés acumulado, el reconocimiento recibido y la carga de trabajo asignada. De manera complementaria, el sistema de diálogo, basado en un modelo simplificado de Procesamiento del Lenguaje Natural, establecerá los patrones de interacción entre el usuario y los agentes virtuales, permitiendo simular conversaciones y respuestas coherentes con el contexto narrativo (Chung, 2003; Jurafsky y Martin, 2025).

El sistema de inteligencia artificial especificará los mecanismos mediante los cuales el simulador interpreta las decisiones del usuario, ajusta la dificultad de los escenarios y genera respuestas personalizadas. A nivel conceptual, este componente integra técnicas de modelado del usuario y aprendizaje por refuerzo, las cuales se desarrollarán en profundidad en capítulos posteriores. La arquitectura general del sistema describe la relación entre los distintos módulos, el flujo de información y los puntos de entrada y salida, incluyendo la interfaz de usuario, la base de conocimiento, el motor de simulación y el sistema de evaluación (Bourdeau et al., 2010; Park Woolf, 2009).

El diseño del simulador se concibe como un proceso iterativo, en el cual la documentación no se limita a describir una versión final y estática del sistema, sino que actúa

como una guía evolutiva para su desarrollo. Esta concepción permite incorporar progresivamente nuevas variables, eventos y comportamientos a medida que avanza la investigación. En este sentido, la documentación de diseño constituye el esqueleto intelectual y narrativo del simulador, donde confluyen la pedagogía, la ingeniería del conocimiento y la narrativa interactiva (Aldrich, 2009; Schwaber y Sutherland, 2020). Es este conjunto documental el que garantiza que un eventual desarrollo futuro del sistema mantenga la coherencia entre sus tres ejes fundamentales: formación, simulación y toma de decisiones.

4.4 Evaluación Tecnológica Preliminar

En esta etapa se analizan las condiciones tecnológicas y conceptuales que encuadran el desarrollo del simulador propuesto, así como las características del entorno en el que operará el agente inteligente. Este análisis resulta fundamental para identificar los límites, las oportunidades y la viabilidad técnica del diseño planteado, estableciendo un marco realista para su eventual implementación futura (Bourdeau et al., 2010; Mitchell, 1997).

El simulador se concibe como un entorno digital semiabierto, en el cual el usuario, en el rol de director de proyectos, interactúa con múltiples agentes virtuales, tales como miembros del equipo, clientes, interesados y patrocinadores. Esta interacción se desarrolla a través de un flujo continuo de percepciones, decisiones y consecuencias, que reproduce la dinámica cotidiana de la gestión de proyectos de software. Dado su propósito educativo, el entorno no busca replicar de manera exacta todos los aspectos físicos del mundo real, sino capturar la complejidad cognitiva, organizacional y social propia del trabajo en equipos tecnológicos (Aldrich, 2009; Lateef, 2010).

Desde la perspectiva de la teoría de los agentes racionales, el entorno presenta una serie de características relevantes. En primer lugar, se trata de un entorno parcialmente accesible, ya que el usuario no dispone de información completa sobre el estado interno del sistema. Variables como la motivación real del equipo o el nivel de satisfacción del cliente no se presentan de forma explícita, sino que deben inferirse a partir de señales indirectas, reportes y comportamientos observables. Asimismo, el entorno es no determinístico, dado que una misma acción puede producir resultados distintos según el perfil de los agentes involucrados y factores probabilísticos (Mitchell, 1997; Norvig y Russell, 1995). Por ejemplo, una reprimenda puede incrementar el rendimiento de un integrante del equipo y, al mismo tiempo, desmotivar a otro con características diferentes.

El simulador se organiza en episodios discretos, representados por escenarios o eventos específicos; sin embargo, estos episodios mantienen una continuidad narrativa, ya que las decisiones tomadas en uno influyen directamente en los siguientes, configurando una progresión coherente a lo largo del proyecto. Aunque las variables internas del sistema evolucionan de manera continua, las decisiones del usuario se expresan mediante eventos discretos, como acciones, elecciones o intercambios comunicativos (Norvig y Russell, 1995). Además, el entorno es dinámico, en tanto su estado puede modificarse con el paso del tiempo incluso en ausencia de acciones del usuario, por ejemplo, cuando la moral del equipo o la comunicación se deterioran como consecuencia de la inacción o de una gestión inadecuada.

Estas propiedades imponen requisitos específicos sobre el diseño del agente inteligente encargado de percibir, interpretar y reaccionar dentro del entorno simulado. En particular, dicho agente deberá incorporar mecanismos de percepción parcial, adaptación continua y aprendizaje contextual, que le permitan responder de manera coherente a la evolución del sistema y a las decisiones del usuario (Mitchell, 1997).

El entorno del simulador puede describirse formalmente mediante el modelo PAGE (*Percepts, Actions, Goals, Environment*), que organiza los componentes fundamentales de un agente inteligente. En este marco, las percepciones corresponden a las señales que el agente recibe del entorno, incluyendo mensajes del usuario, variaciones en variables globales como avance, moral o riesgo, e indicadores asociados a eventos tales como retrasos, bloqueos o conflictos (Norvig y Russell, 1995). Por ejemplo, un agente del equipo puede percibir que el usuario omitió consultar su opinión, lo que reduce su nivel interno de satisfacción y aumenta la probabilidad de conflictos futuros.

Las acciones representan las respuestas posibles del agente frente a una situación determinada. Estas pueden ser de naturaleza comunicativa, como enviar mensajes o expresar disconformidad; operativa, como asumir tareas o solicitar ayuda; o evaluativa, reaccionando ante decisiones del usuario. Cada acción tiene un impacto tanto en las variables internas del sistema como en la percepción que el usuario construye sobre el estado del proyecto. Los objetivos, por su parte, orientan la conducta de cada agente. Así, mientras el cliente prioriza entregas puntuales, el líder técnico enfatiza la calidad, los desarrolladores buscan equilibrio entre esfuerzo y reconocimiento, y el usuario intenta maximizar el éxito global del proyecto. La coexistencia de metas, a menudo contradictorias, genera el contexto de tensión y negociación que da sentido a la simulación (Norvig y Russell, 1995).

El entorno, comprende el conjunto de reglas y condiciones bajo las cuales se produce la interacción. En el simulador propuesto, este incluye los escenarios narrativos, las variables de estado —como tiempo, moral, calidad, satisfacción y reputación—, los sistemas de comunicación textual basados en Procesamiento del Lenguaje Natural y el módulo de inteligencia artificial encargado de evaluar y registrar las decisiones del usuario. El uso del modelo PAGE permitirá organizar de manera sistemática las relaciones entre el usuario y los agentes virtuales, asegurando que las interacciones sean coherentes, medibles y escalables (Bourdeau et al., 2010; Norvig y Russell, 1995).

Desde el punto de vista del acceso a la tecnología, se identifican tanto los recursos disponibles como las limitaciones actuales para una futura implementación del simulador. En términos de accesibilidad tecnológica, el sistema se concibe para ejecutarse en entornos web o de escritorio estándar, sin requerimientos de hardware especializado. Se prioriza la compatibilidad con tecnologías abiertas, como motores de simulación basados en Python o Unity, así como módulos de inteligencia artificial implementables mediante frameworks ampliamente utilizados, como TensorFlow o PyTorch (Géron, 2022).

El diseño se apoya en técnicas de Procesamiento del Lenguaje Natural, Aprendizaje por Refuerzo y modelado basado en reglas. Dado que el presente trabajo no contempla una implementación concreta, se establecen recomendaciones preliminares orientadas a un desarrollo futuro, dejando abierta la posibilidad de incorporar progresivamente mayor complejidad algorítmica. En este sentido, la gestión de datos y la escalabilidad constituyen aspectos centrales, ya que el simulador deberá almacenar información detallada sobre la interacción del usuario —decisiones, resultados y métricas—, lo que permitirá construir modelos adaptativos en etapas posteriores (Barto y Sutton, 2018; Jurafsky y Martin, 2025).

No obstante, también se reconocen limitaciones técnicas relevantes. La generación dinámica de texto y la simulación de múltiples agentes implican una elevada demanda de procesamiento, especialmente cuando se integran técnicas avanzadas de inteligencia artificial (Géron, 2022). Por esta razón, el diseño propuesto se orienta hacia una arquitectura modular, escalable y progresiva, que permita iniciar el desarrollo con reglas determinísticas y, en fases posteriores, incorporar mecanismos de aprendizaje automático sin comprometer la viabilidad del sistema.

Capítulo 5 – Diseño de Simulador. Fase II: Diseño de la Arquitectura y Representación

5.1 Definición de la Arquitectura de Sistemas

El simulador propuesto se concibe como un sistema inteligente compuesto por múltiples subsistemas, organizados de manera modular y comunicados entre sí a través de flujos de información definidos. Su núcleo es el agente inteligente de aprendizaje, cuya función es interpretar el entorno, seleccionar acciones adecuadas y mejorar su desempeño con base en la experiencia acumulada del usuario. La arquitectura se estructura en base al modelo clásico *input → calculations → output*, donde cada módulo cumple un rol específico dentro del flujo de datos del sistema (Aldrich, 2004, p. 87), pero se adapta para incorporar una capa de PLN y retroalimentación. Este enfoque favorece la trazabilidad, escalabilidad y la posibilidad de implementar versiones funcionales progresivas del simulador sin alterar su diseño conceptual original. En coherencia con el alcance de esta intervención, la arquitectura se presenta a nivel conceptual y se define de manera modular para habilitar una implementación incremental por etapas.

5.2 Arquitectura General del Sistema

El simulador propuesto se estructura a partir de una arquitectura compuesta por cinco subsistemas principales, interconectados en torno a un agente inteligente central. Esta organización modular permite separar responsabilidades, facilitar la escalabilidad del sistema y asegurar una interacción coherente entre percepción, razonamiento, comunicación, simulación y evaluación del aprendizaje.

El subsistema de percepción constituye el punto de entrada del sistema y se encarga de recolectar las señales provenientes tanto del entorno como del usuario. En este nivel se integran las acciones realizadas por el participante —tales como decisiones, mensajes o elecciones dentro de eventos— junto con los estados internos del sistema, que incluyen variables como el progreso del proyecto, la moral del equipo, la calidad del producto, el nivel de riesgo y la reputación. Este subsistema cumple el rol de los perceptores del agente inteligente, funcionando como sensores cognitivos que permiten interpretar el estado actual del entorno simulado (Norvig y Russell, 1995). Por ejemplo, cuando el usuario retrasa una entrega o emite un mensaje con tono conflictivo, el sistema percibe variaciones en el tiempo disponible, la satisfacción del cliente y la motivación del equipo, generando una representación actualizada de la situación .

El núcleo del simulador se encuentra en el subsistema de cálculo y razonamiento, donde se ejecuta el modelo de inteligencia artificial. Este subsistema se compone de dos elementos funcionalmente diferenciados. El elemento de desempeño es responsable de seleccionar la respuesta o acción más adecuada del sistema ante un conjunto determinado de perceptos, definiendo en términos prácticos cómo reaccionan los agentes virtuales frente a la conducta del usuario (Norvig y Russell, 1995). Complementariamente, el elemento de aprendizaje analiza la secuencia de interacciones acumuladas con el objetivo de mejorar la toma de decisiones futura, ajustando ponderaciones, probabilidades o reglas internas (Barto y Sutton, 2018; Mitchell, 1997). De este modo, el sistema no se limita a reaccionar de manera estática, sino que aprende patrones de comportamiento del usuario, permitiendo ofrecer experiencias progresivamente más precisas y adaptativas.

El subsistema de comunicación, basado en técnicas de Procesamiento del Lenguaje Natural, se encarga de gestionar los intercambios textuales entre el usuario y los agentes simulados. Este componente opera siguiendo un flujo de comprensión, interpretación y generación, mediante el cual transforma el texto ingresado por el usuario en variables semánticas que luego son utilizadas como perceptos por el agente inteligente. Por ejemplo, ante una afirmación como “Estamos atrasados, pero prefiero entregar algo incompleto antes que nada”, el sistema puede identificar una intención asociada a un nivel elevado de riesgo, ajustando en consecuencia parámetros internos como la tensión del proyecto o la confianza del equipo (Goldberg, 2017; Jurafsky y Martin, 2025).

El subsistema de simulación actúa como capa de salida del sistema y es responsable de generar las respuestas visibles para el usuario. En este nivel se encuentran los efectores del agente, es decir, las acciones que modifican el estado del entorno o producen retroalimentación directa. Estas respuestas pueden adoptar diversas formas, incluyendo manifestaciones narrativas —como la reacción de un miembro del equipo o de un cliente—, actualizaciones métricas —por ejemplo, cambios en los indicadores de avance, moral o riesgo— o alteraciones estructurales del entorno, como la aparición de nuevos eventos o consecuencias derivadas de decisiones previas (Aldrich, 2009; Norvig y Russell, 1995).

El subsistema de control y evaluación cumple la función de registrar de manera sistemática todas las interacciones del usuario y los resultados generados por el sistema. A partir de esta información, se construyen métricas de rendimiento y se garantiza la trazabilidad del proceso de aprendizaje (Kirkpatrick y Kirkpatrick, 2006). Este módulo resulta esencial para

la fase de evaluación adaptativa, ya que permite elaborar el perfil del usuario, analizar su evolución a lo largo de la simulación y ofrecer retroalimentación pedagógica al finalizar la experiencia.

5.3 Modelo de Agente Inteligente

El agente inteligente representa el sistema tutelar de inteligencia artificial encargado de observar, interpretar y modular el comportamiento del entorno, su diseño se apoya en la estructura funcional de un agente racional, concebido como una entidad capaz de percibir su entorno, mantener una representación interna del estado del sistema, razonar sobre dicha información y actuar de forma coherente con un conjunto de objetivos definidos (Bourdeau et al., 2010; Park Woolf, 2009).

En primer lugar, el agente dispone de un conjunto de registros de información del entorno, ligadas al comportamiento del usuario: decisiones tomadas, mensajes enviados o elecciones realizadas dentro de los eventos; como los resultados derivados de dichas acciones, conformando la base informativa sobre la cual opera el sistema (Norvig y Russell, 1995; Park Woolf, 2009).

La información percibida es integrada en un modelo de estado interno, que almacena una representación dinámica del contexto actual del proyecto, este modelo incluye las principales variables del sistema, tales como avance, moral del equipo, nivel de riesgo, calidad y estrés, así como su evolución a lo largo del tiempo. De este modo, el agente no responde únicamente a estímulos aislados, sino que actúa considerando la trayectoria histórica del proyecto y las tendencias emergentes (Bourdeau et al., 2010; Park Woolf, 2009).

Sobre esta representación opera el razonador, cuya función es analizar el estado del sistema y determinar qué tipo de respuesta resulta más adecuada en función de las metas globales del simulador. Dichas metas buscan mantener un equilibrio entre el éxito del proyecto y el bienestar del equipo, evitando optimizaciones extremas que perjudiquen alguno de estos aspectos. El razonamiento del agente permite evaluar tensiones, compensaciones y posibles conflictos entre objetivos concurrentes (Norvig y Russell, 1995; Park Woolf, 2009).

Complementariamente, el planificador habilita al agente a anticipar consecuencias futuras y a definir secuencias plausibles de eventos. A partir del estado actual y de las decisiones recientes del usuario, el agente puede proyectar escenarios posibles, preparando

respuestas que refuercen la coherencia narrativa y pedagógica de la simulación. Esta capacidad de anticipación resulta clave para sostener una experiencia progresiva y realista.

En situaciones críticas, el agente dispone además de mecanismos de respuesta automática o reflejos, que se activan cuando determinadas variables superan umbrales predefinidos. Por ejemplo, niveles excesivos de riesgo o estrés pueden disparar alertas, advertencias o eventos de crisis, garantizando que el sistema reaccione con inmediatez ante condiciones que comprometan el desarrollo del proyecto.

Las decisiones resultantes del proceso de percepción, razonamiento y planificación se materializan a través de los efectores del agente. Estos efectores ejecutan las acciones definidas, ya sea modificando el estado interno del sistema, generando respuestas narrativas de los agentes virtuales o comunicando retroalimentación directa al usuario. De este modo, el ciclo de percepción–decisión–acción se completa, asegurando una interacción continua y coherente entre el usuario y el entorno simulado (Norvig y Russell, 1995; Park Woolf, 2009).

El comportamiento del agente puede representarse mediante una función:

$$f: P \longrightarrow A$$

Donde P es el conjunto de perceptos recibidos y A el conjunto de acciones posibles. En la práctica, esta función se compone de dos capas: Una capa determinística, basada en reglas fijas de comportamiento, por ejemplo: “si la moral < 40%, activar evento de conflicto” (Norvig y Russell, 1995, p. 40); y una capa estocástica o adaptativa, que ajusta respuestas según el historial del usuario, integrando técnicas básicas de aprendizaje por refuerzo y probabilidad condicional (Norvig y Russell, 1995, p. 525).

Relación entre los subsistemas. La arquitectura global se organiza siguiendo el modelo conceptual *input* → *calculations* → *output*, donde:

- El *input* proviene de las acciones del usuario y del estado del entorno.
- Las *calculations* son las inferencias, razonamientos y actualizaciones internas que realiza la IA.
- El *output* corresponde a las respuestas, consecuencias y nuevas situaciones generadas.

Este flujo constante garantiza que el simulador opere como un ecosistema dinámico de aprendizaje, donde cada decisión alimenta el sistema y contribuye a construir un modelo de usuario cada vez más preciso. En términos pedagógicos, esta arquitectura no solo permite

reproducir contextos realistas de gestión, sino también medir y analizar cómo el participante razona, reacciona y aprende, lo que convierte al simulador en una herramienta formativa basada en la evidencia.

5.4 Diseño del Modelo Dinámico Central

El modelo dinámico central constituye el núcleo operativo del simulador, ya que define cómo se representa la realidad del entorno de gestión de proyectos, qué variables intervienen, cómo se relacionan entre sí y bajo qué reglas interactúan tanto el usuario como los agentes virtuales que representan a los distintos interesados del proyecto. El propósito de este modelo es traducir el conocimiento experto y las situaciones reales de la práctica profesional en un sistema formal de ecuaciones, ponderaciones y reacciones, permitiendo que el simulador se comporte de manera coherente, creíble y pedagógicamente efectiva (Aldrich, 2009; Chung, 2003).

El diseño del modelo adopta una aproximación mixta, combinando distintos niveles de representación para equilibrar realismo, control pedagógico y escalabilidad computacional. En primer lugar, incorpora un modelo determinístico basado en reglas fijas, encargado de definir relaciones de causa–efecto directas y predecibles entre decisiones y variables del sistema. Por ejemplo, una demora en una entrega genera una disminución inmediata en la satisfacción del cliente, reflejando vínculos ampliamente aceptados en la gestión de proyectos (Banda, 2014, p. 86).

Sobre esta base se integra un modelo probabilístico apoyado en lógica difusa, que introduce incertidumbre controlada para simular la naturaleza no lineal y parcialmente impredecible del comportamiento humano y organizacional. De este modo, una misma decisión —como una negociación agresiva— puede producir resultados distintos según el contexto, con probabilidades asociadas que reflejan la variabilidad inherente a los sistemas sociales y organizacionales (Banda, 2014, p. 88; Pearl, 1988).

El sistema incorpora un modelo adaptativo basado en microestrategias de inteligencia artificial, compuesto por pequeñas rutinas heurísticas de razonamiento que representan preferencias y patrones de reacción de los agentes virtuales. Este nivel permite que los agentes reaccionen de manera diferenciada según el estado del proyecto, el historial de decisiones del usuario y su propio perfil, aportando variabilidad contextual y aprendizaje progresivo al entorno simulado (Banda, 2014, p. 206). Este enfoque se vincula con los principios de sistemas

inteligentes interactivos que combinan reglas, modelos internos y mecanismos de adaptación contextual (Bourdeau et al., 2010; Park Woolf, 2009).

La combinación de estos tres niveles posibilita que el simulador no se limite a mostrar consecuencias fijas, sino que evolucione dinámicamente en función del desempeño del usuario y de las condiciones internas del proyecto, manteniendo al mismo tiempo coherencia narrativa y control pedagógico.

Desde el punto de vista estructural, el modelo dinámico se organiza en subsistemas interconectados, siendo uno de los principales el sistema de eventos y habilidades. Los eventos constituyen las unidades básicas de la simulación y representan situaciones realistas de gestión, tales como la renuncia de un miembro del equipo, cambios de alcance, conflictos con el cliente o desviaciones presupuestarias. Cada evento impacta de manera predominante en una de las habilidades clave del gerente de proyectos y, de forma secundaria, en una o dos habilidades adicionales (Aldrich, 2009; Lateef, 2010).

Las habilidades principales modeladas en el sistema incluyen la gestión del tiempo, la gestión del alcance, la gestión del presupuesto, la gestión de la calidad, la negociación y la comunicación. Esta estructura permite evaluar el desempeño del usuario de manera multidimensional, vinculando cada decisión tomada con competencias específicas y facilitando una retroalimentación precisa y contextualizada.

En conjunto, el modelo dinámico central actúa como el mecanismo que articula narrativa, evaluación y adaptación, constituyendo la base formal sobre la cual el simulador construye experiencias realistas, progresivas y orientadas al desarrollo efectivo de competencias profesionales.

Cada decisión tomada por el usuario modifica el estado del sistema mediante reglas de impacto ponderadas, del tipo:

$$\Delta V_i = f(D, w, p)$$

Donde:

ΔV_i= variación en la variable afectada.

D= decisión elegida.

w= peso relativo de la habilidad o métrica afectada.

p= probabilidad de éxito asociada a la acción.

Ejemplo:

Evento “Proveedor aumenta precios 40%”.

Decisión: Negociar descuento (éxito = 65%)

Si éxito: Presupuesto +5%, Moral +2pts.

Si falla: Presupuesto –10%, Confianza del proveedor –10pts.

Estas reglas conforman la IA de reglas fijas (fixed-rule AI), que asegura coherencia y repetibilidad entre sesiones.

Sistema de Consecuencias y retroalimentación. El sistema dinámico actualiza las variables globales del proyecto según una matriz de ponderaciones y las ecuaciones de propagación de consecuencias. Se distinguen dos niveles: las consecuencias inmediatas, visibles al instante (cambio de métricas o comentarios de agentes) y las consecuencias diferidas*, que están p*rogramadas para activarse en eventos futuros (por ejemplo, una decisión arriesgada puede generar un bug crítico varios turnos después).

El cambio de estado global se modela mediante la fórmula $S_{t+1} = S_t + \sum_i \Delta V_i \times w_i$

Donde:

S_{t+1} = Nuevo estado del sistema tras una decisión. Es el resultado de aplicar los efectos de la decisión del usuario sobre el estado anterior.

S_t = Estado del sistema en el tiempo t. Es el tablero general del proyecto en ese momento, incluye métricas como Presupuesto, Tiempo, Moral, Calidad y Satisfacción del cliente.

$\sum_i$ = Suma de los efectos de todos los eventos activos. En cada ciclo, puede haber más de una variable afectada (por ejemplo, Tiempo y Moral). Se suman todos los cambios ponderados.

ΔV_i = Variación de una variable i. Es el cambio puntual en una métrica concreta (por ejemplo, +2 puntos en Moral o –3 días en Tiempo). Representa cuánto cambia cada variable por la decisión tomada.

w_i = Peso de influencia de la variable i. Indica qué tan importante es esa variable dentro del modelo general. Por ejemplo, quizás el Tiempo tiene peso 0.25 y la Calidad 0.20 dentro del estado global.

i = Índice que recorre cada variable afectada. Se refiere a cada componente individual del sistema: $i = 1$ puede ser *Tiempo*, $i = 2$ *Presupuesto*, $i = 3$ *Moral*, etc.

S_t = Estado general del proyecto en el tiempo t. Este enfoque da lugar a un sistema de física organizacional, donde las variables se afectan mutuamente, generando un entorno dinámico y no lineal.

Sistema de Microestrategias. Además del motor de reglas, el simulador incorpora microestrategias de IA que otorgan a los agentes (miembros del equipo, cliente, interesados) rasgos de personalidad y reactividad. Cada agente posee un conjunto de preferencias, aversiones y umbrales de tolerancia, que determinan cómo percibe y reacciona ante las acciones del usuario. Ejemplo:

El *líder técnico* valora la calidad sobre la velocidad; reacciona negativamente a órdenes de horas extras (*Crunch*).

El *cliente* prioriza cumplimiento de fechas; responde positivamente a comunicación proactiva aunque implique retrasos moderados.

El *QA* no va a permitir decisiones que salten pasos en las pruebas, incluso si ahorran tiempo.

Estas dinámicas se implementan mediante sistemas de reglas con representación gradual de valores, donde las variables no se modelan de forma estrictamente binaria sino en términos continuos o probabilísticos. Este tipo de modelado resulta consistente con los enfoques de representación bajo incertidumbre utilizados en inteligencia artificial y aprendizaje automático (Bishop, 2006; Murphy, 2022). Por ejemplo, la satisfacción del cliente puede expresarse como $Satisfacción = 0.4 \times Comunicación + 0.3 \times Tiempo + 0.3 \times Calidad$ pero con cada componente definido en términos lingüísticos (“bajo”, “medio”, “alto”) y convertidos a valores numéricos mediante funciones de pertenencia.

Cada evento dentro del simulador se procesa mediante un ciclo de decisión estandarizado que estructura la interacción entre el usuario y el agente inteligente. El ciclo se

inicia con la percepción del estado actual del proyecto, junto con la acción o el texto ingresado por el usuario.

La información textual es luego analizada por el módulo de Procesamiento del Lenguaje Natural, que identifica la intención del usuario y la clasifica dentro de una categoría de decisión pertinente. Este procedimiento se apoya en técnicas modernas de representación semántica y modelos de lenguaje entrenados con arquitecturas basadas en Transformers, ampliamente documentadas en la literatura reciente (Devlin et al., 2019; Jurafsky y Martin, 2025; Tunstall et al., 2022). A partir de esta clasificación, el motor de evaluación aplica reglas determinísticas y probabilísticas acordes al tipo de evento y al contexto del sistema, integrando razonamiento estructurado y evaluación bajo incertidumbre (Pearl, 1988; Murphy, 2022).

Como consecuencia, se actualizan las métricas del proyecto y las variables asociadas al desempeño del usuario, reflejando el impacto acumulativo de sus decisiones. El sistema genera entonces retroalimentación teórica y contextual, explicando los efectos de la acción realizada, en coherencia con los principios de intervención adaptativa en sistemas tutores inteligentes (Bourdeau et al., 2010; Park Woolf, 2009).

Finalmente, los módulos de aprendizaje adaptativo analizan los resultados obtenidos para ajustar la dificultad de los escenarios o el foco pedagógico, reforzando áreas de debilidad y cerrando el bucle de percepción, decisión, acción y aprendizaje.

5.4.1 Finalidad Pedagógica del Modelo

El modelo dinámico no solo reproduce la complejidad técnica de la gestión de proyectos, sino que busca activar la reflexión metacognitiva del usuario; el objetivo es que cada usuario perciba que sus decisiones no son aisladas, sino que responden a un sistema coherente de causas y efectos que imita la realidad profesional. El equilibrio entre reglas fijas, probabilidad difusa y microestrategias adaptativas permite que cada simulación sea única, conservando al mismo tiempo su coherencia interna. Así, el aprendizaje surge de la experiencia directa: el usuario experimenta, reflexiona, ajusta su estrategia y mejora, cumpliendo de manera efectiva con el Ciclo de Aprendizaje Experiencial de Kolb (2015).

5.5 Ontología y Representación de Conocimiento

La Representación del Conocimiento (*Knowledge Representation*, por sus siglas en inglés, KR) constituye el componente conceptual que permite que el simulador razone sobre las situaciones que presenta y produzca respuestas coherentes. En términos prácticos, define

qué sabe el sistema, cómo lo organiza y cómo lo utiliza para tomar decisiones o generar retroalimentación inteligente hacia el usuario (Bourdeau et al., 2010; Mitchell, 1997).

En el simulador de gestión de proyectos, la KR no se limita a almacenar datos, sino que modela conocimiento experto, estructurando relaciones entre conceptos, reglas y comportamientos propios del entorno laboral, de esta forma, la IA puede analizar una decisión del usuario, vincularla con principios teóricos de gestión y producir consecuencias razonadas, más allá de una simple respuesta preprogramada.

5.5.1 Tipos de Conocimiento Representado

El modelo del simulador integra tres tipos de conocimiento principales:

Conocimiento Declarativo. Representa la base factual del dominio: roles, métricas, tipos de eventos y conceptos teóricos. Ejemplos:

"Un sprint dura dos semanas."

"Un cliente insatisfecho reduce la moral del equipo."

"El PMBOK define el riesgo como la probabilidad de un evento adverso y su impacto."

Este tipo de conocimiento se almacena como redes semánticas y estructuras de relación causal entre entidades del dominio (proyecto, equipo, cliente, producto).

Conocimiento Procedimental. Incluye las reglas que gobiernan el comportamiento del sistema y los agentes. Son las reglas de producción que definen cómo actuar ante determinadas condiciones:

```
SI (retraso > 5 días) Y (cliente insatisfecho)
Entonces activar evento "negociación de extensión"
```

Estas reglas conforman la base de la IA de reglas fijas (fixed-rule AI**)** y los sistemas de consecuencia y adaptación. También incluyen las fórmulas de actualización del estado global ($S_{t+1} = S_t + \sum_i \Delta V_i \times w_i$) y las matrices de ponderación entre habilidades y métricas.

Conocimiento Cualitativo o Causal (razonamiento contextual). Modela relaciones indirectas, dependientes de contexto, que no siempre son cuantificables. Ejemplo: "Una comunicación empática mejora la motivación incluso si el proyecto está atrasado." O "Un error técnico grave puede ser tolerado si la confianza con el cliente es alta."

Este conocimiento se expresa mediante reglas difusas y microestrategias dentro de los agentes, que asignan grados de pertenencia o intensidad en lugar de valores binarios.

5.5.2 Ontología del Simulador

La ontología del simulador define el conjunto de objetos, atributos y relaciones que conforman su universo simbólico, permitiendo representar de manera estructurada la realidad organizacional de un proyecto tecnológico, a través de esta ontología, cada entidad del entorno se describe no solo por sus propiedades individuales, sino también por los vínculos que mantiene con otras entidades, conformando una red de conocimiento coherente y navegable (Norvig y Russell, 1995; Bourdeau et al., 2010). En el núcleo de esta estructura se encuentra el proyecto, que actúa como objeto principal y agrupa las métricas globales y las variables de contexto del sistema. Entre sus atributos se incluyen el tiempo, el presupuesto, el alcance, la calidad, la moral del equipo y la satisfacción del cliente, los cuales evolucionan dinámicamente a lo largo de la simulación. Esta organización responde a una lógica de modelado estructurado del dominio, característica de los sistemas tutores inteligentes y de los entornos basados en conocimiento (Park Woolf, 2009; Bourdeau et al., 2010).

El usuario representa al agente principal controlado por el participante y es responsable de tomar decisiones que alteran el estado del sistema. Este agente se caracteriza por un perfil de habilidades distribuido en seis dimensiones, un estilo de aprendizaje y un historial de decisiones que permite analizar su comportamiento y evolución. En interacción con el usuario se encuentra el equipo, compuesto por agentes simulados que representan distintos roles, como desarrolladores, personal de QA, líderes técnicos o diseñadores. Cada uno de estos agentes posee atributos propios, tales como nivel de moral, carga de trabajo, preferencias y aversiones, que influyen en sus reacciones ante las decisiones del usuario (Norvig y Russell, 1995).

Asimismo, el simulador incorpora clientes e interesados, modelados como agentes externos con expectativas particulares. Su nivel de satisfacción impacta directamente en la reputación del proyecto y en la evaluación global del desempeño. Los cambios en el entorno se materializan a través de los eventos, que constituyen unidades dinámicas capaces de alterar el sistema. Cada evento se define por una causa, un contexto y un conjunto de consecuencias cuantificables, y se caracteriza por atributos como tipo, complejidad, probabilidad base, habilidades afectadas y efectos inmediatos o diferidos.

Las interacciones entre estas entidades se articulan mediante relaciones clave que reflejan la lógica causal y pedagógica del simulador. Por ejemplo, una relación de causa–efecto

puede vincular un retraso en la fase de pruebas con un aumento en la probabilidad de aparición de errores críticos. Del mismo modo, la acción de un agente puede detonar un evento específico, como cuando un reporte de múltiples fallas graves por parte de QA activa un escenario de entrega en riesgo. Otras relaciones permiten asociar eventos con el desarrollo de habilidades, por ejemplo, la negociación de una extensión de plazos fortaleciendo competencias de comunicación; o vincular niveles de habilidad con métricas del proyecto, como una comunicación eficaz incrementando la satisfacción del cliente (Han et al., 2012). La retroalimentación recibida por el usuario también se conecta con la construcción de conocimiento, reforzando la conceptualización teórica a partir de buenas prácticas.

Estas entidades y relaciones se representan mediante redes semánticas, en las que cada nodo simboliza un objeto relevante —como eventos, agentes o variables— y los enlaces describen el tipo de vínculo existente entre ellos, tales como "afecta", "depende de", "refuerza" o "reduce". Esta representación ontológica constituye la base estructural que permite al simulador razonar, adaptarse y mantener coherencia entre la simulación, la evaluación y el aprendizaje.

5.5.3 Representación del Conocimiento y Razonamiento Contextual

Para la gestión de eventos complejos, el simulador emplea marcos de conocimiento (*knowledge frames*) que integran información declarativa y procedimental dentro de una misma estructura. En el ámbito de la inteligencia artificial simbólica, los marcos constituyen un mecanismo clásico de representación del conocimiento que permite modelar situaciones recurrentes mediante estructuras con atributos, valores por defecto y procedimientos asociados (Norvig y Russell, 1995). Cada marco representa un tipo recurrente de situación —por ejemplo, la renuncia de un miembro clave o un conflicto con un cliente— y define tanto el contexto del evento como su lógica de evaluación. En estos marcos se especifican atributos relevantes, como la fase del proyecto, los agentes involucrados, el nivel de severidad y las habilidades impactadas, junto con valores por defecto que establecen parámetros base, tales como la probabilidad de ocurrencia, el tiempo medio de resolución y el impacto típico esperado. Asimismo, cada marco incluye procedimientos adjuntos que determinan cómo se evalúa la respuesta del usuario ante el evento. Este enfoque permite reutilizar estructuras cognitivas, asegurar consistencia interna y escalar el diseño del simulador para gestionar cientos de eventos distintos sin perder coherencia pedagógica ni narrativa (Bourdeau et al., 2010; Park Woolf, 2009).

Sobre esta base ontológica, el razonamiento del sistema combina lógica formal y procesos heurísticos, permitiendo interpretar situaciones complejas de manera flexible pero controlada. La lógica de primer orden se utiliza para evaluar condiciones determinísticas y activar eventos cuando se cumplen ciertos criterios. Por ejemplo:

```
SI (Moral < 50) Y (Presión > 70)

ENTONCES activar evento (Event_Conflicto)
```

Este tipo de reglas garantiza respuestas claras y reproducibles ante escenarios críticos. Complementariamente, las redes semánticas representan las interdependencias entre eventos, habilidades y métricas del sistema, permitiendo propagar efectos indirectos —como el impacto de un conflicto en la satisfacción del cliente o en la moral del equipo— sin necesidad de recalcular todo el modelo desde cero (Norvig y Russell, 1995; Han et al., 2012).

Para incorporar ambigüedad y variabilidad propias del comportamiento humano, el simulador utiliza razonamiento difuso, que introduce grados de pertenencia en la interpretación de las variables. Por ejemplo, una categoría como “comunicación media” no se define como un valor único, sino como un rango con pertenencia parcial a las categorías “baja” y “alta”, lo que permite respuestas más realistas y graduales (Bishop, 2006; Murphy, 2022).

En el módulo de retroalimentación educativa, el sistema incorpora además mecanismos de inferencia pedagógica, apoyados en una base de conocimiento que vincula conceptos teóricos — como los procesos definidos en la Guía PMBOK (PMI, 2013; 2021), el concepto de BATNA en negociación, la matriz de riesgos o los principios del Manifiesto Ágil (Beck et al., 2001; Schwaber y Sutherland, 2020)— con patrones de decisión detectados en el comportamiento del usuario. Mediante razonamiento semántico, el simulador puede generar sugerencias, explicaciones y lecturas personalizadas que conectan la experiencia práctica con los fundamentos teóricos subyacentes.

Esta ontología y su representación del conocimiento permite que el simulador razone de forma consistente con el conocimiento experto modelado, de manera sistemática, transparente y trazable. Gracias a esta estructura, el sistema es capaz de interpretar la intención del usuario, evaluar consecuencias dentro de un marco coherente y ofrecer retroalimentación conceptual que articula práctica y teoría. El enfoque resultante combina técnicas de IA simbólica —reglas, marcos y lógica formal— con componentes de IA estadística y adaptativa, generando un entorno de aprendizaje donde la simulación no solo reacciona, sino que enseña (Norvig y Russell, 1995; Murphy, 2022).

5.5.4 Finalidad y Valor Educativo

La ontología y la representación del conocimiento garantizan que el simulador piense y razone como un experto humano, esta capacidad se fundamenta en los principios de los sistemas tutores inteligentes, que integran modelo del dominio, modelo del estudiante y módulo pedagógico para ofrecer intervenciones contextualizadas (Bourdeau et al., 2010; Park Woolf, 2009). Gracias a esta estructura, el sistema puede entender la intención del usuario, evaluar las consecuencias dentro de un marco coherente y generar retroalimentación conceptual que vincula práctica y teoría. Esta articulación responde a la tradición del aprendizaje experiencial, donde la experiencia se transforma en conocimiento a través de la reflexión estructurada (Dewey, 1938; Kolb, 2015).

Este enfoque combina IA simbólica (reglas, marcos, lógica formal) con IA estadística (aprendizaje y adaptación), generando un entorno de aprendizaje realista, donde la simulación no solo reacciona, sino que enseña (Bourdeau et al., 2010; Park Woolf, 2009).

5.6 Tipos de Contenido y Flujo de Información

El diseño del simulador deberá lograr un equilibrio entre estructura pedagógica y libertad de exploración, combinando contenido lineal, cíclico y abierto. Esta integración busca reflejar la naturaleza de la gestión de proyectos: un proceso ordenado en su planificación, recurrente en su ejecución y flexible en la toma de decisiones. La distribución del contenido se estructura en tres tipos de experiencia complementarias (Aldrich, 2005; Kolb, 2015):

5.6.1 Contenido Lineal

El contenido lineal corresponde a los primeros escenarios del simulador, diseñados para introducir al usuario en la lógica del sistema y los fundamentos de la gestión ágil. En esta etapa, las decisiones son más acotadas y los resultados más previsibles, con el objetivo de asegurar la comprensión del entorno, las métricas y los indicadores del proyecto. El contenido lineal actúa como un tutorial narrativo, donde la IA guía al usuario, explica las consecuencias y proporciona retroalimentación formativa, que responde al principio de andamiaje pedagógico, en el cual el sistema acompaña al aprendiz hasta que adquiere mayor autonomía (Vygotsky, 1978). Esta etapa cumple con los niveles inferiores de la Taxonomía de Bloom (*recordar, comprender y aplicar*), permitiendo que el participante construya las bases cognitivas necesarias antes de enfrentar escenarios abiertos (Bloom, 1956).

5.6.2 Contenido Cíclico

El contenido cíclico refleja el ritmo operativo de la gestión de proyectos, donde ciertas actividades se repiten bajo condiciones variables, cada ciclo (equivalente a una iteración) sigue un patrón estructurado de eventos: planificación, ejecución, revisión y retroalimentación, sin embargo, las condiciones iniciales cambian en función de las decisiones anteriores, generando un entorno dinámico. En este formato, el usuario: Aplica estrategias aprendidas en nuevos contextos; gestiona conflictos internos y externos con diferentes consecuencias; aprende a equilibrar tiempo, alcance y calidad bajo presión.

Este tipo de contenido materializa el aprendizaje experiencial cíclico propuesto por Kolb (2015) (Experiencia → Reflexión → Conceptualización → Aplicación), reforzando la transferencia de conocimiento a situaciones prácticas.

5.6.3 Contenido Abierto

El contenido abierto permite que el usuario explore rutas de acción alternativas o diseñe estrategias personalizadas para resolver eventos, en esta modalidad no existe una única solución correcta, sino múltiples caminos válidos con consecuencias distintas, lo que estimula la creatividad, la autonomía y la toma de decisiones éticas. Cada opción activa reacciones diferentes en los agentes y altera las métricas del sistema (confianza, satisfacción, riesgo). Este tipo de contenido introduce evaluaciones de orden superior (*analizar, evaluar y crear*), alineadas con los niveles más avanzados de la Taxonomía de Bloom (Bloom, 1956). El simulador recompensa la coherencia, la empatía y la innovación, no solo los resultados numéricos, fomentando una visión integral del liderazgo.

5.6.4 Flujo de Información Entre Módulos

El flujo de información del simulador sigue una lógica bidireccional, donde los datos circulan de manera continua entre el usuario, los módulos de inteligencia artificial y los sistemas de retroalimentación. Este proceso puede entenderse como un ciclo dinámico compuesto por cuatro etapas interconectadas que garantizan la adaptación constante del sistema al desempeño y estilo de aprendizaje de cada participante (Bourdeau et al., 2010; Park Woolf, 2009).

En la primera etapa, de entrada, el simulador recibe las acciones del usuario, ya sean decisiones dentro del entorno o expresiones en lenguaje natural. Junto con ellas, se registran el estado actual del proyecto, los eventos activos y el perfil individual del usuario, que incluye su

historial de desempeño, su nivel de competencia y su estilo cognitivo predominante. A continuación, en la fase de procesamiento, los distintos módulos de IA entran en funcionamiento. El *parser* de lenguaje natural interpreta la intención detrás de las acciones o mensajes del usuario (Jurafsky y Martin, 2025); el evaluador de decisiones calcula las consecuencias directas e indirectas de cada elección; y el analizador de perfil, junto con el adaptador pedagógico, ajusta dinámicamente la dificultad y el enfoque del escenario, garantizando que el desafío permanezca dentro de la zona óptima de desarrollo del participante (Vygotsky, 1978). La etapa de salida traduce estos resultados en información visible y significativa. Se actualizan las métricas clave del proyecto (como el tiempo, la moral del equipo, la satisfacción del cliente o el nivel de riesgo) y se generan respuestas narrativas por parte de los agentes simulados, acompañadas de una retroalimentación educativa que incluye explicaciones, referencias teóricas y sugerencias de mejora.

Finalmente, la fase de retroalimentación cierra el ciclo. Los resultados obtenidos se almacenan en la base de conocimiento, donde la IA analiza patrones de comportamiento y desempeño para redefinir las próximas situaciones o desafíos. Al término de cada ciclo, el usuario recibe una síntesis de aprendizaje personalizada, que resume sus aciertos, errores y progresos, reforzando así el carácter formativo y evolutivo del simulador.

Este flujo garantiza coherencia entre decisión → consecuencia → aprendizaje, manteniendo la continuidad narrativa y el propósito pedagógico del simulador.

5.7 Alcance Funcional del Sistema Propuesto

El simulador inteligente propuesto en este trabajo se concibe como un sistema interactivo orientado a recrear situaciones características de la gestión de proyectos de software, permitiendo que los participantes experimenten procesos de toma de decisiones en un entorno dinámico y controlado. Desde el punto de vista funcional, el sistema tiene como finalidad generar escenarios simulados, registrar las acciones del usuario, evaluar sus decisiones y proporcionar retroalimentación orientada al aprendizaje.

A diferencia de otros sistemas de inteligencia artificial diseñados para automatizar la resolución de problemas, el simulador no busca sustituir el razonamiento humano ni optimizar de forma autónoma la gestión del proyecto. Su propósito principal consiste en generar contextos problemáticos verosímiles y analizar el comportamiento del participante dentro de ellos, promoviendo el desarrollo de habilidades estratégicas vinculadas con la coordinación de equipos, la priorización de tareas, la gestión del riesgo y la toma de decisiones bajo presión.

Representación del Estado del Proyecto. El sistema mantiene una representación dinámica del estado del proyecto simulado. Esta representación incluye variables relacionadas con aspectos clave de la gestión de proyectos, tales como el progreso del desarrollo, la disponibilidad de recursos, la calidad del producto, el nivel de riesgo, la satisfacción del cliente y la moral del equipo de trabajo.

Estas variables permiten describir el estado del entorno en cada momento de la simulación y constituyen la base sobre la cual se generan nuevos eventos y se evalúan las consecuencias de las decisiones tomadas por el usuario. Desde el punto de vista de la simulación computacional, este modelo actúa como un sistema dinámico cuyos estados evolucionan a lo largo del tiempo en función de las acciones realizadas y de los eventos que se producen en el entorno (Chung, 2003).

Generación Dinámica de Eventos. Otra función central del simulador consiste en la generación de eventos que representan situaciones problemáticas o cambios en el contexto del proyecto. Estos eventos pueden incluir conflictos entre miembros del equipo, retrasos en tareas críticas, cambios en los requerimientos, fallos técnicos, restricciones presupuestarias o presiones externas provenientes del cliente o de la organización.

El motor de simulación se encarga de seleccionar y encadenar estos eventos en función del estado actual del proyecto, de las decisiones tomadas previamente por el usuario y de las probabilidades asociadas a cada tipo de situación. Este enfoque permite recrear escenarios dinámicos que reflejan la naturaleza incierta y cambiante de los proyectos de software, favoreciendo la toma de decisiones estratégicas en condiciones de complejidad (Aldrich, 2005; Lateef, 2010).

Interacción con el Usuario. El simulador proporciona una interfaz a través de la cual el participante puede analizar la situación del proyecto, seleccionar acciones y emitir decisiones que influyen en el desarrollo de la simulación. Estas acciones pueden incluir la asignación de recursos, la priorización de tareas, la comunicación con el equipo o la adopción de medidas para mitigar riesgos.

Las decisiones del usuario se incorporan al modelo del sistema como nuevas entradas que modifican el estado del proyecto y afectan la generación de eventos futuros. De este modo, la simulación se desarrolla como una secuencia de ciclos de percepción, decisión y retroalimentación, en los que el participante interactúa activamente con el entorno simulado.

Este enfoque se alinea con los principios del aprendizaje experiencial, según los cuales el conocimiento se construye a partir de la acción, la reflexión y la experimentación en contextos significativos (Kolb, 2015; Dewey, 1938).

Registro y Análisis de Decisiones. El sistema registra de forma sistemática las decisiones tomadas por el usuario a lo largo de la simulación. Esta información permite reconstruir la secuencia de acciones realizadas, analizar los patrones de comportamiento del participante y evaluar el impacto de sus decisiones sobre el estado del proyecto.

El registro de estas interacciones constituye una fuente valiosa de datos para la evaluación pedagógica, ya que permite identificar estrategias de resolución de problemas, detectar errores recurrentes y comprender cómo el usuario interpreta las situaciones presentadas por el simulador.

Además, este registro abre la posibilidad de aplicar posteriormente técnicas de analítica del aprendizaje y minería de datos educativos para identificar patrones en el comportamiento de los participantes (Romero y Ventura, 2020; Han et al., 2012).

Evaluación del Desempeño y Retroalimentación. Una función fundamental del sistema consiste en evaluar el desempeño del participante dentro de la simulación y proporcionar retroalimentación orientada al aprendizaje. Esta evaluación se basa en el análisis de las decisiones tomadas por el usuario y en su impacto sobre las variables que describen el estado del proyecto.

El sistema puede generar informes de desempeño que reflejen aspectos como la eficiencia en la gestión del tiempo, la capacidad para anticipar riesgos, la efectividad en la resolución de conflictos o el equilibrio entre calidad, plazo y recursos. Este tipo de retroalimentación permite que el participante reflexione sobre su actuación y desarrolle una comprensión más profunda de los desafíos asociados a la gestión de proyectos (Rubio, 2003; Kirkpatrick y Kirkpatrick, 2006).

Componentes Fuera del Alcance de la Implementación Inicial. Si bien la arquitectura general del simulador permite la incorporación de mecanismos más avanzados de inteligencia artificial —como modelos probabilísticos de usuario, aprendizaje por refuerzo o sistemas conversacionales de lenguaje natural más complejos—, estos componentes no forman parte del núcleo funcional necesario para validar conceptualmente la propuesta presentada en este trabajo.

Su desarrollo requeriría una infraestructura adicional de datos, procesos de entrenamiento de modelos y ciclos prolongados de experimentación, por lo que se plantean como líneas de evolución futura del sistema. La implementación inicial del simulador se centra, por lo tanto, en garantizar una experiencia de simulación pedagógicamente coherente basada en el modelo dinámico del proyecto, la generación de eventos, la interacción con el usuario y la evaluación del desempeño.

En conjunto, estas capacidades funcionales permiten construir un entorno de simulación educativa capaz de recrear situaciones características de la gestión de proyectos de software y de analizar el comportamiento del participante dentro de ellas, estableciendo una base sólida para el desarrollo progresivo de sistemas de simulación inteligente más avanzados.

Capítulo 6 – Diseño de Simulador. Fase III: Modelado Lógico, Dinámica de Eventos y Procesamiento de Lenguaje Natural

6.1 Formulación del Problema Simbólico

El objetivo principal de esta etapa consiste en establecer formalmente el mecanismo lógico del simulador, entendiéndolo como un sistema diseñado para generar situaciones características de los procesos de toma de decisiones en la gestión de proyectos. El sistema no tiene como finalidad resolver los problemas de manera autónoma, sino producir eventos pseudoaleatorios coherentes que planteen desafíos al usuario, permitiéndole actuar, razonar y observar las repercusiones de sus elecciones a lo largo del tiempo.

Desde esta perspectiva, la inteligencia artificial cumple una función dual. Por un lado, deberá generar estados y eventos siguiendo una coherencia narrativa y lógica; por otro, evaluará y registrará los resultados de las decisiones del usuario, traduciendo sus consecuencias en métricas del proyecto y en retroalimentación pedagógica estructurada. De este modo, el simulador se formula como un problema de búsqueda donde las soluciones son producidas por el usuario humano, quien asume el rol de agente racional, mientras que la IA actúa como un entorno inteligente encargado de administrar los estados, las reglas de transición y las funciones de evaluación (Mitchell, 1997; Norvig y Russell, 2021).

6.1.1 Formalización del Sistema

En base a Norvig y Russell (1995), el comportamiento del simulador se representa mediante la formulación estándar de problemas de búsqueda y decisión en IA simbólica:

$$S = (E_i, A, f, G, C)$$

Donde:

E_i: estado inicial del proyecto y del perfil del usuario.

A: conjunto de acciones que el usuario puede expresar a través del lenguaje natural.

f: función de transición que aplica las consecuencias definidas por las reglas del modelo dinámico.

G: estado objetivo o situación de cierre del escenario.

C: función de costo o impacto global de las decisiones tomadas.

Estado Inicial ($\mathbf{E_i}$). En el contexto del simulador, el estado inicial (E_i) se establece con valores base para las variables globales del proyecto, como tiempo, presupuesto, moral del equipo, calidad, satisfacción del cliente; y para el perfil inicial del usuario: habilidades, nivel de experiencia y estilo de liderazgo. A partir de este punto, el motor de eventos pseudoaleatorios selecciona una situación a presentar, como un conflicto de prioridades, una renuncia inesperada o error en producción, cada una modelada como un problema simbólico con estados, operadores y consecuencias. Ejemplo:

Proyecto inicial: moral del equipo = 80, satisfacción del cliente = 70, presupuesto = 100%, tiempo restante = 30 días.

Perfil del jugador: fortalezas en comunicación y negociación, debilidades en planificación.

Este punto de partida determina la posición del agente (el usuario) en el espacio de estados, y condiciona la dificultad de los primeros eventos.

Conjunto de Acciones (A). Cada entrada textual representa una decisión simbólica que el motor de Procesamiento del Lenguaje Natural interpreta, clasifica y transforma en una acción formal dentro del dominio del simulador (Jurafsky y Martin, 2025):

1. El usuario recibe una situación o evento.
2. Escribe libremente cómo actuaría o qué decisión tomaría.
3. El módulo de Comprensión del lenguaje natural (*Natural Language Understanding*, por sus siglas en inglés, NLU) analiza el texto, identifica la intención principal y extrae entidades clave.
4. Esa intención se traduce internamente a una acción simbólica $a_i \in A$.

Ejemplo: Si ante un evento de Conflicto de prioridades entre cliente y equipo, el usuario escribe: “Voy a hablar con el cliente para explicar la situación y ofrecer una entrega parcial”; el sistema identifica la intención de negociar (acción base) y la intención de mantener la comunicación, clasificándola como acción compuesta:

$$a_i = \text{Negociar} + \text{Comunicar}$$

Cada acción simbólica así generada pasa al motor lógico, que ejecuta la función de transición: $E_{t+1} = f(E_t, a_i)$.

Función de Transición (f). La función de transición f describe cómo cambia el entorno del simulador una vez que el usuario expresa su decisión en lenguaje natural y esta es interpretada por el sistema mediante el módulo de Procesamiento del Lenguaje Natural, a partir del texto ingresado por el jugador. Formalmente, la transición puede expresarse como:

$$E_{t+1} = f\big(E_t, \text{Interpretar}(Texto)\big)$$

Donde:

E_t: estado actual del proyecto,

Interpretar$(Texto)$: proceso mediante el cual el subsistema NLU convierte la respuesta textual en una acción simbólica a_i,

E_{t+1}: nuevo estado del proyecto tras aplicar las reglas correspondientes a esa acción.

El resultado depende de una combinación de factores:

Reglas Determinísticas. Relaciones directas y previsibles entre acción e impacto, por ejemplo: si la acción implica posponer una entrega ⇒ tiempo +5, satisfacción del cliente –10):

$$satisfacion' = \begin{cases} satisfaccion - 10, & si\ retraso > 5 \\ satisfaccion, & en\ otro\ caso \end{cases}$$

Reglas Probabilísticas o Difusas. Relaciones inciertas o dependientes del contexto, por ejemplo una negociación agresiva puede mejorar la satisfacción en un caso, pero empeorarla en otro según la moral del equipo.

$$p_{éxito} = \sigma(\beta_0 + \beta_1 M \pm \beta_3 s + \beta_4 H)$$

Donde:

$p_{éxito}$: Probabilidad de que la acción del usuario produzca un resultado favorable. Se calcula en función del contexto actual del proyecto.

$\sigma(z)$: Función logística o sigmoide que transforma el valor total del argumento z en una probabilidad entre 0 y 1. Cuanto más alto el valor, mayor la probabilidad de éxito.

β_0: Término independiente o nivel base del entorno. Si es negativo, el escenario es más difícil por defecto; si es positivo, más favorable.

$\beta_1 M$: Influencia de la moral del equipo (M). Una moral alta aumenta las probabilidades de éxito; una moral baja las reduce.

$\beta_2 T$: Influencia de la confianza del cliente (T). Un cliente que confía más en el equipo tolera mejor los errores o retrasos.

$-\beta_3 s$: Penalización por la agresividad o dureza de la decisión (s). Cuanto más intensa o autoritaria sea la acción, menor es la probabilidad de que tenga éxito.

$\beta_4 H$: Factor de historial (H) del usuario. Si ha obtenido resultados favorables en el pasado, sus próximas acciones tienen mayor probabilidad de éxito.

Estas transiciones se almacenan internamente en una red semántica de estados, donde cada nodo representa un conjunto de variables del proyecto (tiempo, moral, calidad, etc.) y cada enlace corresponde a una decisión interpretada por la IA (Bishop, 2006; Murphy, 2022). De este modo, el simulador registra no solo qué acción fue tomada, sino también cómo fue expresada, integrando elementos lingüísticos, emocionales y estratégicos del usuario.

Estado Meta (G). El estado objetivo representa la condición de éxito del simulador y se define como un conjunto de condiciones deseadas que reflejan la madurez del proceso de gestión, más que un resultado numérico único; este estado puede adoptar diferentes formas según el escenario: (a) Cumplir con los objetivos del proyecto sin superar el límite de tiempo ni presupuesto; (b) Mantener la moral del equipo y la satisfacción del cliente por encima de un umbral o (c) Lograr un balance entre resultados y bienestar del equipo (objetivo ético-pedagógico). El sistema evalúa la distancia entre el estado actual y la meta mediante una función heurística que cuantifica el progreso del jugador y orienta la selección de eventos. Esta función combina indicadores cuantitativos. Como tiempo, moral, satisfacción del cliente; con métricas cualitativas derivadas del lenguaje natural, como tono, intención o coherencia comunicacional; este modelo se adapta a la representación de grafos de estado en IA clásica (Norvig y Russell, 1995).

En términos pedagógicos, alcanzar el estado meta implica que el usuario ha demostrado competencia integral en gestión, aplicando habilidades técnicas, sociales y estratégicas de forma coherente, el usuario logra este estado cuando demuestra decisiones consistentes, razonamiento ético y equilibrio entre resultados y bienestar del equipo, cumpliendo con los enfoques contemporáneos de evaluación de competencias complejas (Kirkpatrick y Kirkpatrick, 2006; PMBOK, 2021)

Función de Costo y Heurística (C). La función de costo $C(E_t, a_i)$ mide el impacto de la decisión interpretada por el sistema de PLN sobre las variables críticas del proyecto. Cada término representa la variación en un indicador clave ponderada por su importancia relativa (Mitchell, 1997; Norvig y Russell, 1995):

$$C(E_t, a_i) = \alpha \cdot \Delta T + \beta \cdot \Delta Presupuesto + \gamma \cdot \Delta Moral + \delta \cdot \Delta Calidad$$

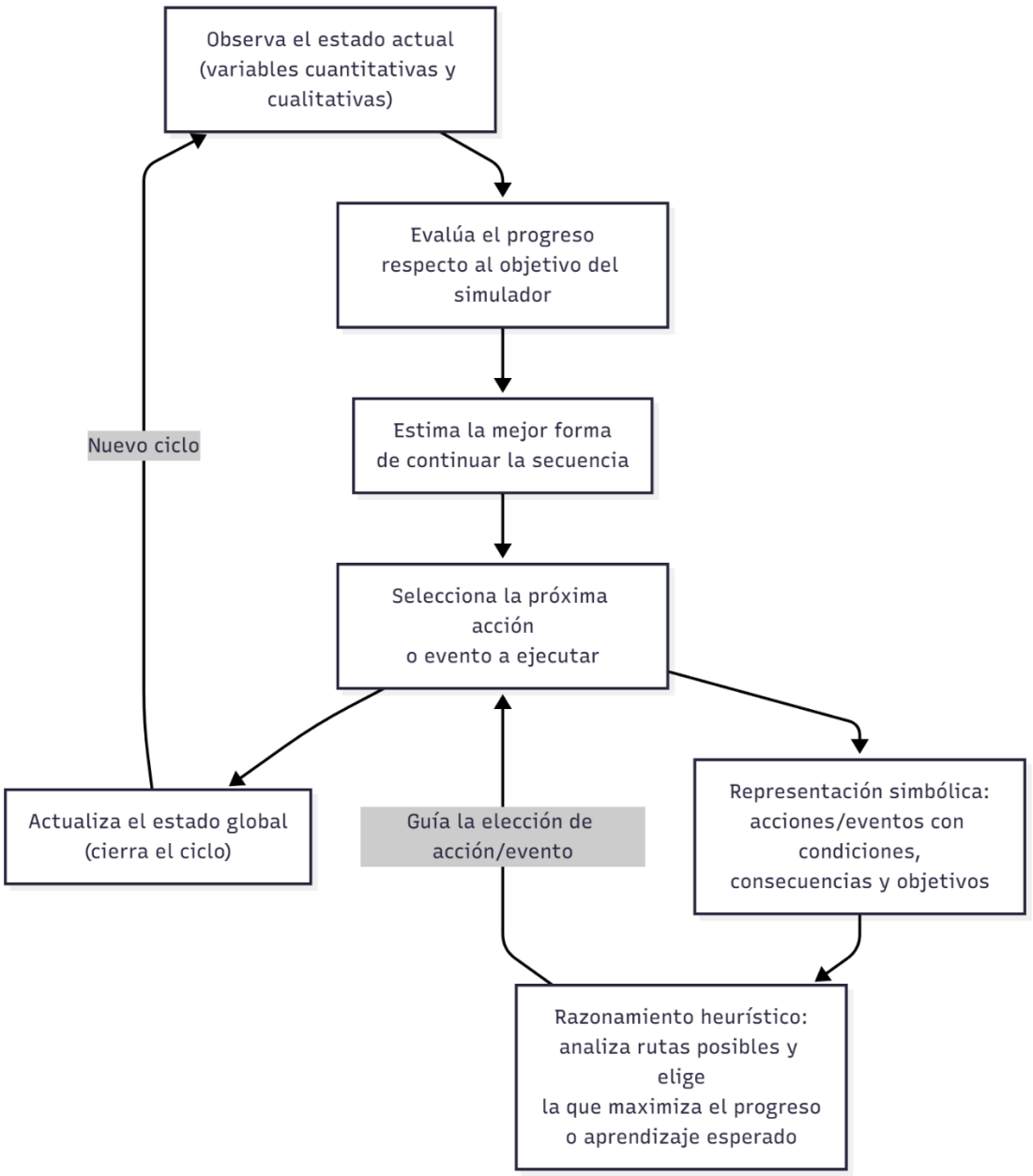
Donde los coeficientes $\alpha, \beta, \gamma, \delta$ representan la importancia relativa de cada métrica según el tipo de escenario, por ejemplo: en un entorno de crisis técnica, la calidad tiene más peso que el tiempo.

Espacio de Estados. El espacio de estados del simulador se compone de nodos que representan configuraciones del proyecto y enlaces que corresponden a decisiones expresadas en lenguaje natural. El tamaño del espacio crece exponencialmente con el número de eventos y decisiones, por lo que la IA emplea búsqueda heurística guiada para recorrerlo sin necesidad de evaluar todas las combinaciones posibles.

Gráficamente, el espacio se visualiza como un árbol de decisiones interactivo, donde cada rama conduce a un nuevo estado del proyecto y el conjunto de ramas define la narrativa emergente (Norvig y Russell, 1995); la principal diferencia de un sistema de opciones discretas con uno que hace uso de PLN, es que este último permite que el espacio se expanda de manera dinámica: cada texto ingresado genera un nuevo nodo simbólico, clasificado por su intención y tono, esto produce un entorno de simulación abierto, en el que las trayectorias posibles crecen con la creatividad del usuario, manteniendo la coherencia a través de las reglas heurísticas del modelo, la Figura 3 muestra una representación gráfica del ciclo de razonamiento del simulador, el cual observa el estado actual, evalúa su progreso, estima la mejor continuación y selecciona la acción siguiente. Cada decisión actualiza el estado global y expande dinámicamente el árbol de estados según las entradas del usuario y las reglas del modelo.

Figura 3

Representación en forma de Árbol de Decisiones del Espacio de Estados del Simulador



Nota: La figura muestra el ciclo de razonamiento del simulador: el sistema observa el estado actual, evalúa su progreso, estima la mejor continuación posible y selecciona la siguiente acción o evento. Cada decisión actualiza el estado global y genera nuevas ramas en el árbol de estados, permitiendo que el espacio crezca dinámicamente a partir de las entradas del usuario y de las reglas heurísticas del modelo. Figura de elaboración propia.

El Agente Planificador. El agente planificador del simulador, implementado en los módulos de Selector de Eventos y Adaptador Pedagógico, actúa como un mecanismo de

orquestación adaptativa encargado de seleccionar el siguiente evento en función del estado actual del proyecto, el perfil del usuario y los objetivos pedagógicos aún no consolidados. Desde una perspectiva formal, su función se define como una estrategia de selección heurística sobre el espacio de estados, donde el sistema prioriza aquellos eventos que maximizan el valor formativo esperado, en lugar de optimizar una función matemática global (Norvig y Russell, 1995). Esta distinción es relevante: el simulador no busca la solución óptima del proyecto, sino la secuencia de situaciones que favorezca el aprendizaje progresivo del usuario.

El agente combina heurísticas cuantitativas como el desempeño histórico, la variación de métricas del proyecto y la frecuencia de errores en habilidades específicas, con heurísticas cualitativas basadas en el análisis del tono, la coherencia comunicacional y señales de liderazgo o conflicto detectadas mediante procesamiento del lenguaje natural. Este enfoque se alinea con la arquitectura de los sistemas tutores inteligentes, en los cuales el módulo pedagógico selecciona intervenciones adaptativas en función del modelo del estudiante (Bourdeau et al., 2010; Park Woolf, 2009).

El simulador no busca que la IA encuentre una solución óptima, sino que genere un entorno de aprendizaje estructurado y coherente, en el cual el usuario tome decisiones en lenguaje natural. La función de costo cuantifica los efectos de cada acción, la heurística evalúa su valor educativo, el espacio de estados se expande dinámicamente a partir del texto ingresado, y el agente planificador selecciona los próximos desafíos para mantener continuidad y progresión pedagógica. Este enfoque convierte el simulador en un entorno simbólico de aprendizaje adaptativo, donde el razonamiento humano es el centro, y la IA cumple el rol de marco lógico, evaluador y narrador inteligente (Vygotsky, 1978; Kolb, 2015).

6.2 Operadores y Encadenamiento de Eventos

En el simulador, las acciones del usuario funcionan como operadores simbólicos que transforman el estado actual del sistema y, al hacerlo, desencadenan nuevos eventos dentro de la estructura narrativa; este proceso define la dinámica de encadenamiento, en la cual una decisión no actúa de forma aislada, sino como parte de una reacción en cadena que modifica las condiciones del entorno y genera nuevas situaciones de aprendizaje (Bourdeau et al., 2010; Mitchell, 1997).

6.2.1 Los Operadores como Acciones Simbólicas

Cada operador representa una acción interpretada a partir del lenguaje natural del usuario. El módulo de PLN analiza la intención expresada en el texto y la traduce en una acción simbólica estandarizada que se aplica dentro del modelo lógico del simulador. Estas acciones se integran al conjunto de operadores A, definidos formalmente como:

$$a_i : E_t \rightarrow E_{t+1}$$

Donde cada $a_i$transforma el estado actual E_t en un nuevo estado E_{t+1}, modificando las variables del sistema y las condiciones narrativas (Norvig y Russell, 1995).

Los operadores actúan según reglas fijas que establecen las consecuencias directas de cada tipo de acción, estas reglas determinan qué variables se ven afectadas, en qué magnitud y en qué condiciones contextuales. Por ejemplo, el sistema sabe que una acción asociada a delegar responsabilidades afecta positivamente la moral del equipo y negativamente el tiempo disponible; del mismo modo, una acción de presión o exigencia reduce el riesgo de atraso, pero a costa del bienestar del grupo.

Cada operador activa una o varias reglas de impacto condicional, que dependen del contexto del estado actual, de manera tal que las consecuencias no son lineales, sino que varíen según factores como el historial reciente del usuario, las variables actuales del proyecto y la combinación de eventos activos. La causalidad del sistema se define como:

$$C = \{(a_i, E_t) \rightarrow \Delta V_j\}$$

Donde cada par acción–estado produce un conjunto de variaciones ΔV_jen las variables del entorno. Estas variaciones generan nuevas condiciones de disparo (*triggers*) para otros eventos, creando así un flujo continuo de causas y efectos.

6.2.2 Encadenamiento de Eventos

El encadenamiento de eventos se produce cuando una o varias consecuencias activan otros eventos relacionados dentro del espacio de estados del simulador. Este mecanismo produce que la simulación evolucione de forma dinámica, generando narrativas emergentes a partir de las decisiones del usuario y reflejando la naturaleza acumulativa de la gestión de proyectos (Norvig y Russell, 1995; Mitchell, 1997).

En algunos casos, el encadenamiento adopta la forma de una reacción en cadena, donde una acción concreta desencadena una secuencia de eventos consecutivos. Cada evento se activa

como consecuencia directa del anterior, construyendo una progresión lógica y temporal que simula la escalada gradual de problemas o decisiones incorrectamente gestionadas. Este tipo de dinámica resulta representativo de procesos lineales, en los que la causa y el efecto pueden rastrearse de manera relativamente directa (Mitchell, 1997).

En otros escenarios, el encadenamiento responde a un fenómeno de masa crítica, que se manifiesta cuando la acumulación de múltiples condiciones simultáneas —como baja moral, plazos ajustados y alta presión— supera un umbral definido por el sistema y activa un evento crítico. En este caso, no es una acción aislada la que genera la consecuencia, sino la convergencia de varios factores a lo largo del tiempo, dando lugar a comportamientos no lineales y a la aparición súbita de crisis.

Ambos tipos de encadenamiento representan con mayor fidelidad la complejidad de los proyectos reales, donde los problemas pueden escalar de manera progresiva o emerger abruptamente tras una acumulación de tensiones, reforzando el realismo narrativo y pedagógico del simulador (Pearl, 1988; Murphy, 2022).

Rol del Agente Planificador. El agente planificador adaptativo supervisa todo el proceso de encadenamiento; su tarea es decidir qué eventos se habilitan tras una acción, verificando que la secuencia mantenga consistencia lógica y progresión pedagógica. Para ello, combina reglas fijas, guiadas por el determinismo causal; heurísticas de selección y criterios de balance narrativo. El planificador actúa como un director narrativo y pedagógico, asegurando que el simulador evolucione de forma verosímil, pero siempre orientado a los objetivos de aprendizaje establecidos (Bourdeau et al., 2010; Park Woolf, 2009).

6.3 Generación de Eventos Ponderados

La generación de eventos ponderados constituye el mecanismo mediante el cual el simulador define la aparición de nuevas situaciones o desafíos, garantizando variedad y realismo sin perder coherencia interna. En lugar de seleccionar eventos al azar puro, el sistema emplea un enfoque probabilístico y heurístico, basado en la combinación de un Generador de Números Pseudoaleatorios (*Pseudo-Random Number Generator*, por sus siglas en inglés, PRNG), con un sistema de reglas difusas (*Fuzzy Logic*) que pondera la relevancia, urgencia y adecuación pedagógica de cada posible evento. La variabilidad es necesaria para mantener la experiencia de aprendizaje activa y no predecible, sin embargo, el entorno debe conservar consistencia lógica y pedagógica, de modo que cada evento guarde relación con el estado actual del proyecto y las decisiones previas del usuario. Para ello, el sistema no “elige” eventos

directamente, sino que calcula probabilidades dinámicas para cada candidato en función de múltiples factores: el contexto actual del proyecto, el historial del jugador, las variables críticas (tiempo, moral, calidad, satisfacción) y las reglas de dependencia entre eventos (Chung, 2003).

6.3.1 El Método Monte Carlo

El enfoque pseudoaleatorio adoptado se apoya en los principios del Método Monte Carlo, un marco numérico basado en el muestreo aleatorio controlado para aproximar resultados de sistemas complejos o inciertos. El método Monte Carlo no busca eliminar la incertidumbre, sino modelarla estadísticamente, permitiendo obtener estimaciones a partir de múltiples ejecuciones con variación pseudoaleatoria, cada corrida representa una posible trayectoria del sistema, y la agregación de estas muestras produce una estimación estable del comportamiento general (Barto y Sutton, 2018; Chung, 2003).

$$E[f(x)] \approx \left(\frac{1}{N}\right) \cdot \sum_{(i=1 \to N)} f(x_i), \quad \text{donde: } x_i \sim P(x)$$

Donde:

$E[f(x)]$: Valor esperado (o esperanza matemática) de la función $f(x)$. Representa el resultado promedio teórico del sistema si se pudieran considerar todas las posibles configuraciones de x.

$f(x_i)$: Resultado de la función f evaluada para la muestra i. En el simulador, corresponde al resultado de una ejecución (por ejemplo, desempeño, costo o tiempo total).

x_i: Muestra pseudoaleatoria tomada de la distribución $P(x)$, cada x_i representa un conjunto de condiciones iniciales o parámetros de un escenario.

N: Número total de simulaciones o iteraciones ejecutadas, cuanto mayor sea N, más estable será la estimación.

$\Sigma(_{i=1} \to N)$: Suma sobre todas las muestras generadas.

$P(x)$: Distribución de probabilidad de las variables del sistema. Define cómo se eligen los escenarios o parámetros de simulación.

En el simulador, este enfoque se traduce en la generación de múltiples posibles escenarios de proyecto, donde cada evento tiene una probabilidad asociada y los resultados emergen de la interacción entre decisiones del usuario y factores pseudoaleatorios. Esta

pseudoaleatoriedad controlada cumple un papel esencial dentro del simulador, ya que permite mantener el equilibrio entre realismo e intencionalidad pedagógica: En primer lugar preserva la coherencia narrativa y causal, evitando resultados ilógicos y garantizando que los eventos se desarrollen dentro de un marco creíble; por ejemplo, el sistema impide que se presente una auditoría si el proyecto aún no ha comenzado.

Asimismo, posibilita controlar la progresión de la dificultad, ajustando las probabilidades de ocurrencia de ciertos eventos según la fase del simulador o el rendimiento del usuario. Otro aspecto clave es la garantía de replicabilidad, ya que al utilizar una misma semilla aleatoria es posible repetir escenarios idénticos, lo que facilita comparar estrategias o medir la efectividad de distintas intervenciones. Finalmente, esta metodología permite simular una incertidumbre realista, reproduciendo la imprevisibilidad propia de la gestión de proyectos, pero siempre dentro de límites pedagógicos cuidadosamente controlados.

Se opta por la pseudoaleatoriedad sobre la aleatoriedad pura para equilibrar incertidumbre y coherencia, permitiendo reproducir condiciones experimentales y modelar comportamientos realistas mediante técnicas inspiradas en el Método Monte Carlo, que combina el muestreo aleatorio con la estimación estadística de resultados esperados a partir de una cantidad N de ejecuciones:

$$\bar{\mathrm{E}}[J] = \left(\frac{1}{N}\right) \cdot \sum_{(i=1 \to N)} \mathrm{J}\left(\tau^{(i)}\right),$$

Donde:

$\bar{\mathrm{E}}[J]$: Estimación del valor esperado del desempeño del usuario o del proyecto, obtenida por Monte Carlo.

$\mathrm{J}\left(\tau^{(i)}\right)$: Valor o resultado obtenido en la ejecución i del simulador (por ejemplo, puntaje, nivel de éxito o satisfacción final).

$\tau^{(i)}$: Trayectoria completa de la simulación i, que incluye la secuencia de estados, acciones y eventos.

$\sum_{(i=1 \to N)}$: Suma sobre todas las simulaciones independientes.

N: Número total de ejecuciones pseudoaleatorias consideradas.

6.3.2 Mecanismo de Generación Pseudoaleatoria Guiada

La base de este proceso es un Generador de Números Pseudoaleatorios, que produce secuencias numéricas determinísticas con apariencia aleatoria. Los eventos se generan mediante modelos probabilísticos controlados, donde la aparición de cada tipo de suceso obedece a distribuciones predefinidas (por ejemplo, normal, exponencial o de Poisson) o a dependencias específicas entre variables (Aldrich, 2009; Bourdeau et al., 2010). La probabilidad de un evento depende del estado, la acción y sus ponderaciones de contexto:

$$P(e \mid S_t, a_t) = \frac{[\, C(e \mid S_t, a_t) \cdot w(e \mid S_t, a_t) \,]}{\Sigma_{(e' \in E)} [\, C(e' \mid S_t, a_t) \cdot w(e' \mid S_t, a_t) \,]}$$

Donde:

$P(e \mid , a_t)$: Probabilidad de que ocurra el evento e dado el estado actual S_t y la acción a_t del usuario.

S_t: Estado del sistema en el instante t (por ejemplo, avance del proyecto, moral del equipo, presupuesto disponible).

a_t: Acción tomada por el usuario en ese estado (por ejemplo, reasignar tareas, posponer una entrega, realizar una reunión).

e: Evento posible (por ejemplo, "retraso en desarrollo", "renuncia de un miembro del equipo", "satisfacción del cliente").

$C(e \mid S_t, a_t)$: Máscara de coherencia o validez. Toma valor 1 si el evento e puede ocurrir en ese contexto, y 0 si no (por ejemplo, no puede haber "entrega final" al inicio del proyecto).

$w(e \mid S_t, a_t)$: Peso o ponderación asociada al evento, que ajusta su probabilidad según la dificultad, el rendimiento del usuario o el punto de la simulación.

$\Sigma_{(e' \in E)}[\ldots]$: Normalización sobre todos los eventos posibles E, para asegurar que la suma total de probabilidades sea 1.

Formalmente, para un conjunto de eventos candidatos $E = \{e_1, e_2, \ldots, e_n\}$, se define la probabilidad ajustada de selección como:

$$P(e_i) = \frac{w_i}{\sum_{j=1}^{n} w_j}$$

Donde $w_i$representa el peso dinámico del evento e_i, calculado en tiempo real según su compatibilidad con el estado actual del sistema.

El proceso de selección se describe como:

1. El PRNG genera un valor pseudoaleatorio $r \in [0,1]$.
2. El sistema ordena los eventos por su probabilidad acumulada.
3. Se selecciona el evento $e_k$tal que:

$$\sum_{i=1}^{k-1} P(e_i) < r \leq \sum_{i=1}^{k} P(e_i)$$

De este modo, la generación conserva un componente aleatorio, pero con direccionalidad pedagógica: los eventos más coherentes con el contexto tienen mayor probabilidad de ser elegidos. Así, la pseudoaleatoriedad no implica arbitrariedad, sino la incorporación de variabilidad controlada para representar la complejidad e incertidumbre propias de la toma de decisiones en contextos reales. El estado siguiente surge de aplicar un evento pseudoaleatorio coherente, generado con una semilla reproducible:

$$S_{t+1} = T(S_t, a_t, e_t)$$
$$con \quad e_t = F_e^{-1}(u_t \mid S_t, a_t),$$
$$u_t = PRNG(seed, t)$$

Donde:

S_{t+1}: Estado del sistema en el siguiente paso temporal. Es el resultado de aplicar el evento y la acción al estado actual.

$T(S_t, a_t, e_t)$: Función de transición. Define cómo cambia el estado del sistema a partir del estado actual S_t, la acción a_t y el evento e_t.

e_t: Evento seleccionado de manera pseudoaleatoria, condicionado al contexto actual.

$F_e^{-1}(u_t \mid S_t, a_t)$: Función inversa de la distribución acumulada (CDF) del evento e. Permite convertir el número pseudoaleatorio u_t en un evento concreto, manteniendo la distribución de probabilidades correcta.

u_t: Número pseudoaleatorio generado por un PRNG.

$PRNG(seed, t)$: Algoritmo determinista que, a partir de una semilla fija y del tiempo t, genera una secuencia reproducible de valores pseudoaleatorios.

seed → Valor inicial del generador. Permite replicar exactamente la misma secuencia de eventos si se vuelve a usar.

6.3.3 Incorporación de Lógica Difusa

Para evitar la rigidez de la lógica binaria clásica de verdadero o falso, el sistema aplica Lógica Difusa (*Fuzzy Logic*) al cálculo de compatibilidades, de forma que cada variable del entorno se representa mediante valores lingüísticos en lugar de valores discretos. A diferencia de los enfoques binarios, la lógica difusa facilita un modelado más cercano al razonamiento humano, siendo utilizada en numerosos sistemas inteligentes aplicados a contextos complejos e inciertos (Banda, 2014; Norvig y Russell, 2021).

Cada evento posee un perfil de activación difuso, que describe las condiciones ideales bajo las cuales debe ocurrir. Por ejemplo, un evento de conflicto interno tiene mayor grado de activación cuando la moral es *baja* y la presión es *alta*, mientras que un evento de innovación del equipo se activa con moral *alta* y carga *media* (Banda, 2014; Baradó et al., 2013). Formalmente, el grado de activación de un evento e_i se define como:

$$\mu(e_i) = \min\{\mu_M(M), \mu_P(P), \mu_S(S)\}$$

Donde μ_M, μ_Py μ_S son las funciones de pertenencia difusa asociadas a moral, presión y satisfacción respectivamente. El resultado $\mu(e_i)$ representa la intensidad de compatibilidad del evento con el contexto actual.

El peso total del evento se obtiene combinando esta medida con otros factores heurísticos:

$$w_i = \lambda_1 \cdot \mu(e_i) + \lambda_2 \cdot h(e_i) + \lambda_3 \cdot r(e_i)$$

Donde:

$h(e_i)$: heurística de relevancia pedagógica (por ejemplo, si la habilidad aún no fue ejercitada),

$r(e_i)$: probabilidad residual o base de aparición,

$\lambda_1, \lambda_2, \lambda_3$: coeficientes de ponderación que equilibran los tres factores.

Sistema de Reglas Difusas

El módulo de Lógica Difusa se organiza como un conjunto de reglas del tipo IF–THEN, donde cada regla traduce conocimiento experto en relaciones lingüísticas. Ejemplo:

$$\begin{cases} Si\ (Moral = Baja) \wedge (Presión = Alta) \Rightarrow Evento = Conflicto \\ Si\ (Satisfacción = Alta) \wedge (Tiempo = Bajo) \Rightarrow Evento = Entrega\ Parcial \\ Si\ (Moral = Media) \wedge (Presión = Baja) \Rightarrow Evento = Innovación \end{cases}$$

Cada regla devuelve un grado de pertenencia y, tras el proceso de agregación, el sistema obtiene una distribución de probabilidad suavizada para los eventos candidatos. Este mecanismo permite razonar en condiciones de incertidumbre, integrando factores emocionales y contextuales sin depender de umbrales rígidos (Banda, 2014; Baradó et al., 2013).

6.3.4 Integración con el Agente Planificador

El agente planificador utiliza la salida del sistema de eventos ponderados como base para decidir el próximo escenario. En lugar de elegir directamente, el planificador interpreta los pesos $w_i$como indicadores de conveniencia y selecciona el evento que mejor balancea desafío, coherencia narrativa y valor formativo. Así, el planificador actúa como una capa de control que filtra el azar y lo transforma en intencionalidad pedagógica.

El sistema de operadores y encadenamiento de eventos constituye el núcleo dinámico del simulador, permite que cada interacción escrita por el usuario no solo modifique el estado actual, sino que genere consecuencias y nuevos escenarios en función de reglas causales y de umbrales de masa crítica. Este modelo combina la previsibilidad de las reglas fijas con la flexibilidad de la pseudoaleatoriedad, garantizando una experiencia de simulación que es al mismo tiempo estructurada, variada y formativamente significativa (Bourdeau et al., 2010; Park Woolf, 2009).

6.4 Entrada de Lenguaje Natural

El módulo de entrada de lenguaje natural PLN constituye el punto de interacción principal entre el usuario y el simulador, es a través de este canal que el usuario expresa sus decisiones, estrategias o razonamientos en forma de texto libre, y el sistema las interpreta mediante procesos de análisis lingüístico y semántico, transformándolas en acciones simbólicas que impactan en el estado del entorno. Este componente es fundamental para dotar al simulador de naturalidad comunicativa y profundidad cognitiva, permitiendo que la

experiencia se asemeje a un diálogo humano entre un profesional y su entorno laboral simulado (Jurafsky y Martin, 2025; Manning y Schütze, 1999).

6.4.1 Restricción de Dominio

El éxito de un sistema de PLN no depende únicamente de la complejidad de sus algoritmos, sino de la restricción adecuada de su dominio semántico, en este caso, el dominio se circunscribe al campo de la gestión de proyectos tecnológicos, lo que limita el rango de significados posibles y facilita la interpretación de las intenciones del usuario.

Restringir el dominio significa que el sistema no necesita comprender el lenguaje natural en su totalidad, sino únicamente un conjunto de términos, estructuras y expresiones típicas del contexto profesional simulado; esta limitación controlada permite reducir la ambigüedad y establecer un consenso interpretativo entre el usuario y el agente inteligente (Jurafsky y Martin, 2025; Manning y Schütze, 1999). En términos formales, el dominio léxico del simulador está compuesto por:

- Verbos de acción: decidir, comunicar, delegar, negociar, aprobar, priorizar, resolver, etc.
- Sustantivos de contexto: cliente, equipo, tarea, sprint, entregable, conflicto, retraso, etc.
- Adjetivos y modificadores: urgente, crítico, parcial, satisfactorio, ineficiente, etc.
- Conectores y marcadores de intención: “voy a”, “prefiero”, “deberíamos”, “considero que”, “propongo que”.

De esta manera, el sistema opera sobre un diccionario controlado y extensible, permitiendo procesar las frases con alto nivel de precisión sin requerir interpretación lingüística general.

6.4.2 Proceso de Interpretación

El procesamiento del texto ingresado por el usuario se organiza como un proceso secuencial compuesto por tres etapas diferenciadas, que permiten transformar lenguaje natural en acciones ejecutables dentro del simulador (Jurafsky y Martin, 2025; Manning y Schütze, 1999).

En una primera etapa se realiza el análisis léxico y etiquetado gramatical. El texto es tokenizado y se identifican sus componentes básicos, tales como verbos, sujetos, objetos y

modificadores. A partir de esta segmentación inicial, el sistema detecta palabras clave y patrones de expresión que funcionan como anclas semánticas, sirviendo de base para la interpretación posterior.

A continuación, el sistema lleva a cabo un análisis sintáctico y semántico, en el cual se examinan las relaciones entre los elementos previamente identificados. El objetivo de esta fase es inferir la intención principal del usuario —por ejemplo, negociar, comunicar o reorganizar— y el contexto en el que dicha intención se aplica, como una interacción con el cliente, con el equipo o durante una instancia crítica del proyecto.

Por último, en la etapa de mapeo simbólico y ejecución, la intención interpretada se traduce en una acción simbólica estandarizada α_i, compatible con el modelo lógico del simulador. Esta acción se integra al estado actual del sistema E_ty es procesada por el conjunto de reglas y operadores definidos, permitiendo calcular las consecuencias asociadas y actualizar las variables internas. De manera formal, este flujo se expresa como:

$$Texto_{usuario} \xrightarrow{PLN} \alpha_i \xrightarrow{f} E_t + 1$$

donde el módulo de PLN actúa como función de interpretación, y la función de transición f aplica las reglas y efectos definidos en el modelo dinámico del simulador (Jurafsky y Martin, 2025; Manning y Schütze, 1999).

6.4.3 Tipificación de Perceptos y Reducción de Ambigüedad

En los sistemas inteligentes, cada entrada textual se denomina percepto, una unidad de información que el agente recibe del entorno. El PLN transforma las palabras en perceptos tipificados, asignando etiquetas semánticas que describen su función dentro del contexto del simulador. Por ejemplo, el sistema distingue entre un percepto de acción (“voy a hablar con el cliente”) y un percepto emocional o valorativo (“estoy preocupado por el retraso”), lo que permite adaptar el retroalimentación del sistema y ajustar la evaluación del perfil del usuario (Jurafsky y Martin, 2025; Manning y Schütze, 1999).

La restricción del dominio facilita esta tipificación, ya que reduce el número de interpretaciones posibles y acota el análisis a categorías semánticas predefinidas (acción, intención, emoción, consecuencia, etc.). Dado que el lenguaje humano es inherentemente ambiguo, el simulador incorpora razonamiento difuso para manejar expresiones imprecisas, condicionales o subjetivas. Esto le permite interpretar correctamente enunciados como

“preferiría entregar algo incompleto antes que nada” o “podríamos extender el plazo si el cliente lo acepta” (Banda, 2014).

En estos casos, el sistema asigna grados de pertenencia semántica a diferentes categorías de acción (por ejemplo, entre “negociar” y “ceder”), y selecciona la interpretación con mayor compatibilidad según el contexto actual. De este modo, la ambigüedad no se elimina, sino que se modela como incertidumbre cuantificable, permitiendo al simulador razonar de forma flexible.

El módulo de PLN no solo traduce texto a acciones, sino que también extrae información pedagógica del discurso del usuario, analizando la claridad, el tono y la estructura del lenguaje para inferir rasgos de comunicación, empatía y liderazgo, retroalimentando el perfil del jugador. De esta manera, el simulador utiliza la entrada lingüística tanto para activar reglas del entorno como para evaluar competencias blandas, integrando aprendizaje técnico y emocional en un mismo proceso (Kirkpatrick y Kirkpatrick, 2006).

6.5 Procesamiento Semántico y Sintáctico

El procesamiento del lenguaje natural dentro del simulador se basa en un enfoque multicapa, que permite interpretar las respuestas del usuario desde la estructura gramatical hasta el sentido pragmático del mensaje. Esta arquitectura en niveles garantiza que el sistema no solo lea el texto, sino que comprenda su intención y significado dentro del contexto de la simulación. Para ello, el PLN combina análisis sintáctico, interpretación semántica y razonamiento pragmático, vinculando las expresiones lingüísticas con los conceptos, relaciones y acciones definidas en el modelo de conocimiento del simulador (Jurafsky y Martin, 2025).

6.5.1 Capas de Procesamiento

El sistema de Procesamiento del Lenguaje Natural se organiza en tres capas de análisis complementarias que actúan de manera secuencial y cooperativa, permitiendo transformar el lenguaje natural del usuario en una interpretación significativa dentro del simulador.

En primer lugar, la capa sintáctica se encarga de descomponer la oración en sus componentes gramaticales fundamentales, tales como sujeto, verbo, objeto y modificadores. Para ello utiliza técnicas de etiquetado gramatical y análisis de dependencias, con el objetivo de identificar quién realiza la acción y sobre qué elemento recae. En esta etapa el sistema no

busca aún comprender el significado del mensaje, sino asegurar una estructuración coherente del texto que habilite los análisis posteriores (Manning y Schütze, 1999).

Sobre esta base opera la capa semántica, responsable de interpretar el significado de las palabras y las relaciones que mantienen entre sí. En este nivel se aplica el mapeo semántico que vincula los términos utilizados por el usuario con los conceptos propios del dominio del simulador, como cliente, equipo, entrega, plazo o riesgo. Esta capa permite reconocer sinónimos, equivalencias conceptuales y expresiones dependientes del contexto, estableciendo un puente entre el lenguaje humano y el conocimiento formal del sistema (Jurafsky y Martin, 2025).

La capa pragmática incorpora el contexto, la intención y el tono del mensaje, evaluando cómo el significado se modifica según la situación y la forma de expresión. Esta capa conecta el lenguaje con la realidad simulada, permitiendo interpretar la intención comunicativa del usuario, ya sea una decisión firme, una duda, una advertencia o un intento de persuasión. El análisis pragmático se apoya en el historial del usuario, el estado actual del proyecto y señales emocionales derivadas del texto, integrando la comunicación dentro del flujo narrativo y pedagógico del simulador (Norvig y Russell, 1995).

6.5.2 Integración con el Modelo de Conocimiento

Un sistema de PLN robusto requiere no solo capacidades lingüísticas, sino también conocimiento del mundo o del dominio en el que opera. En este simulador, dicho conocimiento se representa en la ontología y la base de reglas desarrolladas en los capítulos previos, donde se definen entidades, relaciones, eventos y consecuencias. Cada oración procesada se traduce en una estructura lógica compatible con esa base de conocimiento, lo que permite al sistema razonar simbólicamente sobre el texto (Baradó et al., 2013).

Por ejemplo, si el usuario menciona “reorganizar tareas”, el PLN reconoce la acción “reorganizar” como una decisión de planificación y la vincula con el conjunto de reglas y efectos asociados al área de gestión del equipo.

Este proceso de integración semántica se formaliza mediante la correspondencia:

$$Lenguaje \leftrightarrow Conocimiento\ del\ Domino \leftrightarrow Acción\ Simbolica$$

Esta relación garantiza que el significado del texto ingresado por el usuario se conecte directamente con las variables, entidades y relaciones que definen la realidad simulada,

evitando interpretaciones aisladas o desconectadas del contexto del proyecto (Baradó et al., 2013).

Dado que el lenguaje natural es inherentemente ambiguo, el sistema incorpora mecanismos específicos para reducir la incertidumbre semántica y asegurar interpretaciones consistentes. En primer lugar, se aplica una restricción de dominio, limitando el vocabulario y los contextos posibles al ámbito de la gestión de proyectos, lo que reduce significativamente la cantidad de significados alternativos. En segundo lugar, el sistema recurre a la desambiguación basada en contexto, utilizando el estado actual del simulador y el historial de interacciones del usuario para seleccionar la interpretación más probable entre varias opciones posibles. Finalmente, cuando la ambigüedad persiste, se emplea razonamiento difuso, asignando grados de pertenencia a distintos significados y eligiendo aquel que presenta mayor compatibilidad con el entorno y la situación (Norvig y Russell, 1995; Banda, 2014).

De este modo, el procesamiento semántico no se limita a comprender el texto, sino que establece una relación directa entre el lenguaje y la dinámica del simulador. Cada expresión escrita por el usuario se interpreta como un acto simbólico dentro del modelo de simulación, con consecuencias mensurables y coherentes. A partir de esta correspondencia, el sistema actualiza su estado interno y genera respuestas acordes, convirtiendo al lenguaje en el principal medio a través del cual el usuario actúa sobre el mundo simulado.

6.5.3 Arquitectura y Relación con Sistemas Expertos

El modelo de procesamiento del lenguaje implementado se inspira en la arquitectura clásica de los sistemas expertos, donde el razonamiento simbólico se basa en una base de conocimiento y un motor de inferencia. En este caso, el motor de PLN actúa como la capa perceptiva del agente inteligente, mediante recibir entradas lingüísticas, analizarlas sintáctica y semánticamente, vincularlas con la base de conocimiento y producir acciones o inferencias en función de las reglas definidas (Baradó et al., 2013).

Esta integración entre lenguaje y razonamiento experto convierte al simulador en un entorno cognitivo híbrido, donde el componente lingüístico y el componente lógico trabajan de manera complementaria para sostener la coherencia y la adaptabilidad del sistema.

Capítulo 7 – Diseño de Simulador. Fase IV: Diseño del Sistema de Inteligencia Artificial

7.1 Arquitectura del Agente de Aprendizaje

El agente de aprendizaje constituye el núcleo funcional del sistema de inteligencia artificial del simulador, representando la entidad que dota al entorno virtual de capacidades adaptativas, reactivas y orientadas a la optimización del proceso de enseñanza-aprendizaje. Este agente no es un componente aislado sino un sistema cognitivo integrado que percibe razona, actúa, evalúa y aprende de manera continua. Según Norvig y Russell (1995) el diseño del agente se fundamenta en la teoría clásica de agentes inteligentes, donde una entidad autónoma interactúa con su entorno mediante un ciclo de percepción-acción; en términos formales, un agente se define como cualquier entidad que:

- Percibe su entorno a través de sensores físicos o virtuales
- Actúa sobre ese entorno mediante actuadores
- Mantiene un estado interno que representa su conocimiento
- Implementa una función de agente que mapea percepciones a acciones: $f: P^* \rightarrow A$
- Exhibe comportamiento dirigido a objetivos

En entornos educativos, el agente se encuentra con desafíos específicos, como equilibrar las múltiples dimensiones del proceso educativo: la narrativa coherente, la evolución pedagógica, el involucramiento, y la transmisión de conocimientos, tratando de optimizar las funciones objetivo, las que solo observa de forma parcial, ya que el aprendizaje verdadero no es fácil de medir. Además, necesita adaptarse a lo siempre cambiante, ya que el alumno siempre está evolucionando.

7.1.1 Estructura Dual: Elementos de Desempeño y Aprendizaje

El agente inteligente del simulador se organiza a partir de una estructura dual, compuesta por un elemento de desempeño y un elemento de aprendizaje, que operan de manera complementaria. Mientras el primero gobierna la interacción inmediata con el usuario y la evolución del entorno simulado, el segundo cumple una función metacognitiva, orientada a mejorar progresivamente el comportamiento del sistema a partir de la experiencia acumulada (Norvig y Russell, 1995).

El elemento de desempeño es responsable de seleccionar los eventos, evaluar las decisiones del usuario y generar retroalimentación en tiempo de ejecución. Su funcionamiento

comienza con un selector de eventos, que determina cuál de los eventos candidatos debe presentarse al usuario mediante una función de utilidad ponderada (Norvig y Russell, 1995; Mitchell, 1997):

$$U(e_i \mid S) = w_1 \cdot C(e_i, S) + w_2 \cdot P(e_i, H) + w_3 \cdot D(e_i, U) + w_4 \cdot V(e_i, E) + w_5 \cdot T(e_i, t)$$

Donde:

$C(e_i, S)$ Es la coherencia narrativa: evalúa si el evento tiene sentido dentro del estado actual del relato o contexto.

$P(e_i, H)$ Es la pertinencia pedagógica: Mide cuánto contribuye el evento a los objetivos de aprendizaje pendientes

$D(e_i, U)$ Es la dificultad adecuada: Comprueba si el evento es apropiado al nivel o perfil del usuario.

$V(e_i, E)$ Es la variedad experiencial: Favorece la diversidad de situaciones, evitando repeticiones o monotonía.

$T(e_i, t)$ Es el timing narrativo: Considera si el momento temporal o narrativo es oportuno para ese evento.

Los coeficientes w_j representan pesos adaptativos ajustados dinámicamente por el sistema. El evento seleccionado corresponde a aquel que maximiza la función de utilidad:

$$e^* = \underset{e_i}{argmax} U(e_i \mid S)$$

Una vez presentado el evento, el evaluador de consecuencias integra el análisis semántico del texto ingresado por el usuario con heurísticas de evaluación y reglas del dominio, determinando el impacto de la decisión sobre las principales variables del proyecto, el proceso finaliza con el generador de retroalimentación, que ajusta el tono, el nivel de detalle y el estilo del mensaje en función del perfil del usuario y del contexto de la decisión (Bourdeau et al., 2010; Park Woolf, 2009).

En paralelo, el elemento de aprendizaje opera como un mecanismo de mejora continua del sistema, mantiene una observación y registro estructurados de las interacciones entre el usuario y el simulador, incluyendo eventos presentados, decisiones tomadas, consecuencias generadas, niveles de compromiso y progreso alcanzado. Sobre esta base, el sistema realiza una evaluación de efectividad, analizando la calidad de sus propias decisiones mediante

preguntas tales como si el evento fue apropiado, si la dificultad resultó adecuada o si la retroalimentación produjo el efecto esperado (Park Woolf, 2009).

A partir de este análisis, el agente ajusta sus parámetros, modifica los pesos w_j, los umbrales de dificultad y las heurísticas internas según los patrones detectados; refinando el perfil cognitivo del usuario, que contempla fortalezas, debilidades, sesgos, preferencias y velocidad de aprendizaje. Este perfil retroalimenta al elemento de desempeño, cerrando el ciclo adaptativo entre acción, evaluación y mejora. Esta estructura dual permite que el simulador no solo reaccione de forma inmediata a las decisiones del usuario, sino que también aprenda de la experiencia, afinando progresivamente su comportamiento para ofrecer interacciones cada vez más coherentes, personalizadas y pedagógicamente efectivas.

7.1.2 Ciclo Operativo del Agente

El agente de aprendizaje funciona dentro de un bucle operativo continuo, a través del cual percibe su entorno, interpreta la información obtenida, toma decisiones, actúa sobre el sistema, evalúa los resultados y ajusta su comportamiento en función de la experiencia. Este ciclo iterativo constituye el núcleo de su inteligencia adaptativa y garantiza su capacidad de evolución constante a lo largo del tiempo (Norvig y Russell, 1995). El agente opera en un bucle continuo:

```
[Percepción] → [Interpretación] → [Decisión] → [Acción] → [Evaluación] → [Aprendizaje]
```

Durante la fase de percepción, el agente recopila información a través de sensores virtuales organizados en distintos niveles jerárquicos. En el nivel bajo, se registran variables cuantitativas del proyecto, tales como tiempo, moral del equipo, satisfacción del cliente o estado de las tareas; en el nivel medio, el agente detecta tendencias, proporciones y relaciones entre variables, permitiéndole identificar desviaciones o patrones emergentes; en el nivel alto, realiza un análisis semántico del lenguaje del usuario, extrayendo información contextual, emocional y cognitiva de las interacciones. Finalmente, en el nivel metacognitivo, evalúa los aspectos pedagógicos de la experiencia, identificando el progreso del jugador, sus estilos de razonamiento y los posibles puntos de intervención educativa (Bourdeau et al., 201; Norvig y Russell, 1995).

La etapa de interpretación consiste en transformar la información percibida en conocimiento significativo, el agente combina análisis semántico, evaluación heurística y contextualización temporal para construir una representación coherente del estado actual del

entorno y del comportamiento del usuario. El proceso culmina con una síntesis interpretativa, que integra los distintos niveles de información en una comprensión unificada de la situación.

En la fase de decisión, el agente aplica un modelo de optimización multiobjetivo, que opera simultáneamente en tres horizontes temporales. A corto plazo, ejecuta decisiones operativas inmediatas, destinadas a mantener la coherencia del simulador o responder a contingencias locales; a mediano plazo, adopta decisiones tácticas por sesión, que buscan equilibrar el desafío, la motivación y el progreso pedagógico; a largo plazo, planifica decisiones estratégicas entre cada sesión, orientadas a mejorar la experiencia global del usuario, su perfil de aprendizaje y la eficacia de los mecanismos internos del sistema (Bourdeau et al., 201; Norvig y Russell, 1995).

La acción del agente se materializa mediante actuadores virtuales, que le permiten intervenir sobre el entorno de diversas maneras: modificar variables del proyecto, activar o desactivar eventos, generar mensajes de retroalimentación, ajustar parámetros de dificultad o actualizar ponderaciones internas. Estas acciones no son arbitrarias, sino el resultado de un razonamiento orientado a objetivos y condicionado por el conocimiento acumulado del agente (Bourdeau et al., 2010).

En la etapa de evaluación, el agente compara los resultados observados con los resultados esperados, calculando señales de error cuantificables que reflejan la eficacia de sus decisiones; este proceso de contraste continuo le permite detectar desviaciones, medir impactos y preparar la información necesaria para su aprendizaje interno (Bourdeau et al., 2010).

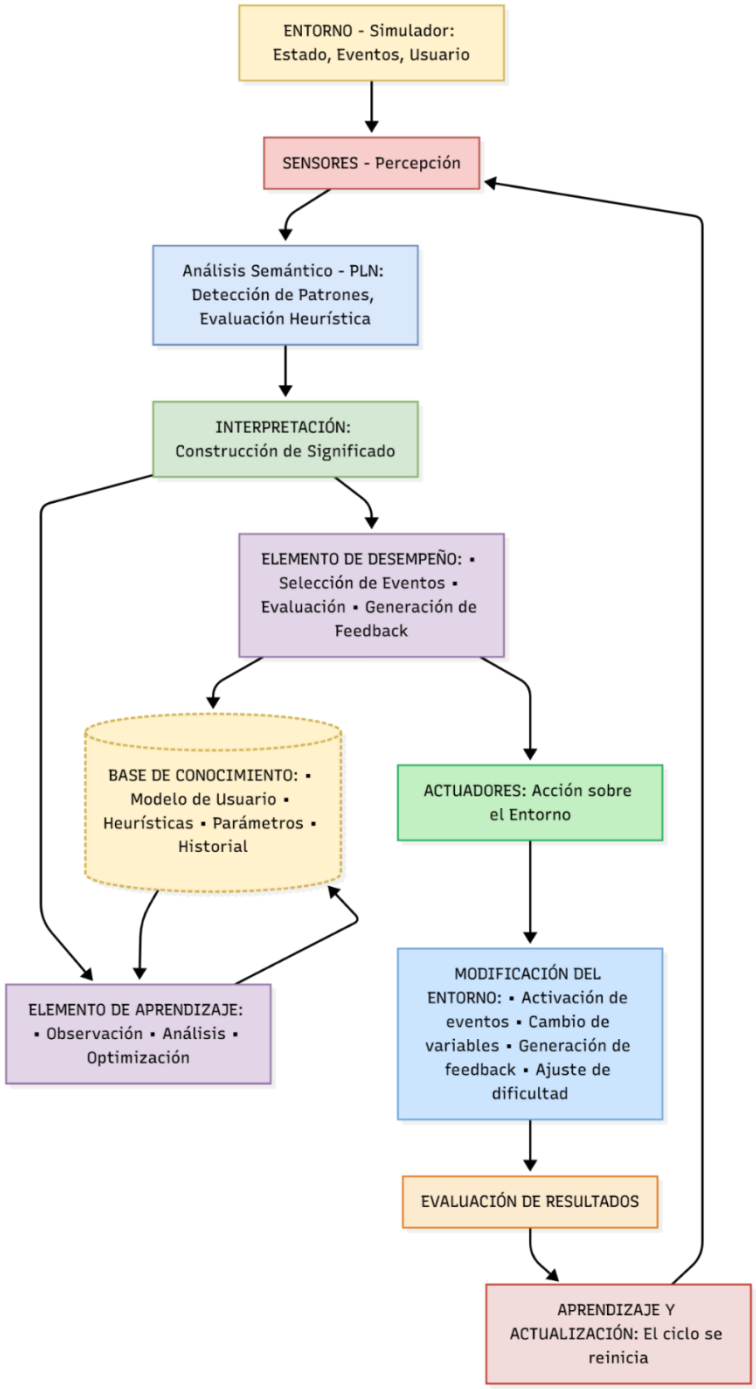
Finalmente, la fase de aprendizaje actualiza el conocimiento del agente en múltiples niveles. En el nivel reactivo, el sistema ajusta su respuesta inmediata ante estímulos repetidos o inesperados; en el nivel adaptativo, modifica sus reglas de decisión y heurísticas en función de los patrones observados en sesiones anteriores; en el nivel metacognitivo, revisa los propios mecanismos de aprendizaje, optimizando la forma en que evalúa, razona y actúa; y en el nivel transferible, generaliza lo aprendido para aplicarlo en nuevos contextos, reforzando su capacidad de ajuste ante usuarios o escenarios distintos.

De esta arquitectura cíclica emergen propiedades que definen al agente como un sistema inteligente completo, en la Figura 4 se puede ver un resumen grafico de esta arquitectura. El agente posee autonomía, al operar sin intervención externa; persistencia, al mantener continuidad entre sesiones; y reactividad, al responder en tiempo real a los cambios del entorno. Posee proactividad, para anticipar situaciones futuras mediante predicción

heurística; razonamiento simbólico, al interpretar conceptos abstractos en lugar de simples datos; y finalidad pedagógica, subordinando todas sus decisiones al objetivo educativo central del simulador. Finalmente, posee adaptabilidad multimodal, que le permite ajustar tanto estrategias cuantitativas como cualitativas, y coherencia comportamental, que asegura la consistencia lógica de sus acciones a lo largo de la experiencia de aprendizaje (Norvig y Russell, 1995).

Figura 4

Ciclo operativo del agente del simulador



Nota. La figura resume el flujo de procesamiento del agente: percibe el entorno, interpreta los eventos mediante análisis semántico, selecciona acciones apoyado en su base de conocimiento y ajusta el estado del simulador. El ciclo se repite de forma continua para mantener coherencia y adaptación durante la experiencia. Figura de elaboración propia.

7.2 Función Objetivo del Sistema

La función objetivo es el componente que define qué debe optimizar el agente, en contextos pedagógicos, esta función es inherentemente multidimensional y compleja, ya que el éxito educativo no puede reducirse a una métrica única.

7.2.1 Naturaleza Multidimensional de la Función Objetivo

La función objetivo constituye el núcleo decisional del simulador, ya que define qué debe optimizar el agente inteligente al interactuar con el usuario. A diferencia de sistemas clásicos de optimización —como aquellos orientados a maximizar beneficios económicos o minimizar error de predicción—, el aprendizaje implica equilibrar dimensiones cognitivas, motivacionales y experienciales que pueden resultar parcialmente conflictivas entre sí (Kolb, 2015; Kirkpatrick y Kirkpatrick, 2006).

Desde esta perspectiva, la función objetivo del sistema se plantea como la maximización del aprendizaje profundo, sujeta a una serie de restricciones que aseguran una experiencia pedagógica sostenible y creíble (Mitchell, 1997):

```
F_objetivo = Maximizar[Aprendizaje_profundo]

             sujeto a:

               Compromiso_sostenido ≥ umbral_mínimo

               Diversidad_experiencial ≥ umbral_variedad

               Coherencia_narrativa ≥ umbral_credibilidad

               Frustración ≤ umbral_tolerable
```

7.2.2 Descomposición de la Función Objetivo.

Para operacionalizar este enfoque, la función objetivo se expresa como una combinación ponderada de componentes evaluados dinámicamente en el tiempo (Aldrich, 2005; Kolb,2015; Park Woolf, 2009):

$$F(t) = w^1 \cdot Progreso(t) + w^2 \cdot Engagement(t) + w^3 \cdot Diversidad(t) + w^4 \cdot Coherencia(t) + w^5 \cdot Transferencia(t) - w^6 \cdot Frustración(t)$$

Cada término representa una dimensión clave del proceso de aprendizaje y contribuye de forma diferenciada al valor global del sistema.

El progreso pedagógico, $Progreso(t)$, mide el incremento real de las competencias objetivo del usuario y se define como:

$$Progreso(t) = \sum_i [competencia_i(t) - competencia_i(t-1)] \,/\, n_competencias$$

donde cada competencia se estima considerando el desempeño consistente en eventos relacionados, la capacidad de aplicar conocimientos en contextos variados y la profundidad de razonamiento evidenciada en las respuestas del usuario.

El compromiso sostenido, $Engagement(t)$, evalúa señales indirectas de involucramiento cognitivo:

$$Engagement(t) = \alpha \cdot f_{longitud_{respuestas(t)}} + \beta \cdot f_{tiempo_{apropiado(t)}} + \gamma \cdot \mathrm{f}_{\mathrm{completitud_{interacción(t)}}}$$

donde se bonifican respuestas elaboradas dentro de un rango óptimo de extensión, se penalizan tiempos excesivamente breves o prolongados, y se detectan abandonos o saltos reiterados de eventos.

La diversidad experiencial, $Diversidad(t)$, incentiva la exposición balanceada a distintos tipos de situaciones:

$$Diversidad(t) = 0.6 \cdot \left(\frac{tipos_{eventos\ unicos}}{total_{tipos}}\right) + 0.4 \cdot \left(\frac{áreas_{ejercitadas}}{total_{áreas}}\right)$$

Este componente posee un peso decreciente a lo largo del tiempo: es más relevante en las fases iniciales del aprendizaje y se reduce progresivamente cuando el sistema prioriza la profundización en áreas específicas.

La coherencia narrativa, $Coherencia(t)$, garantiza la credibilidad del entorno simulado y se calcula como:

$$Coherencia(t) = 1 - \left(\frac{violaciones_{causalidad}}{total_{transiciones}}\right)$$

verificando que los eventos respeten la cronología lógica, las dependencias causales y la continuidad del estado del proyecto.

La transferencia de conocimientos, $Transferencia(t)$, bonifica la aplicación espontánea de conceptos aprendidos en situaciones nuevas:

$$Transferencia(t) = \Sigma_j \, \delta_j \cdot similitud_{conceptual(decisión\ actual, concepto\ aprendido\ j)}$$

Donde $\delta_j = 1$ si el usuario aplica el concepto j sin que el evento lo sugiera explícitamente.

Por último, la frustración, $Frustración(t)$, actúa como término de penalización y se estima mediante indicadores combinados:

$$Frustración(t) = \lambda^1 \cdot indicador_{respuestas\ breves\ tras\ complejidad} + \lambda^2 \cdot indicador_{tiempo\ excesivo} + \lambda^3 \cdot indicador_{lenguaje\ emocional\ negativo} + \lambda^4 \cdot indicador_{abandono\ sesion}$$

Representaciones posibles de la Función Objetivo. La función objetivo puede implementarse mediante distintos formalismos. En escenarios donde las relaciones entre variables son aproximadamente lineales o cuadráticas, puede utilizarse una representación polinomial (Mitchell, 1997):

$$F(x) = \Sigma_i \, w_i \cdot x_i + \Sigma_{ij} \, w_{ij} \cdot x_i \cdot x_j + b$$

Esta aproximación ofrece alta interpretabilidad, aunque resulta limitada para capturar relaciones no lineales complejas.

Alternativamente, puede emplearse una representación mediante redes neuronales:

$$F(x) = \sigma(W^3 \cdot \sigma(W^2 \cdot \sigma(W^1 \cdot x + b^1) + b^2) + b^3)$$

Donde σ representa funciones de activación no lineales. Esta opción permite una mayor capacidad de aproximación, a costa de menor interpretabilidad.

Identificación de fortalezas y debilidades. La función objetivo no solo evalúa el estado global del usuario, sino que permite descomponer el desempeño por áreas específicas:

$$Fortalezas = \{área_i : competencia(área_i) > percentil_{75}\}$$

$$Debilidades = \{área_j : competencia(área_j) < percentil_{25}\}$$

Esta información habilita al agente a reducir la exposición a áreas dominadas, reforzar aquellas deficitarias, ajustar la dificultad de forma diferenciada y generar retroalimentación focalizada sobre conceptos no comprendidos (Park Woolf, 2009).

7.3 Modelo de Usuario

El modelo de usuario es la representación interna del agente de cada estudiante, en este modelo se incluyen aspectos cognitivos, preferencias, fortalezas y debilidades, y así el sistema logra comprender y anticipar su conducta. Este modelo es crucial para personalizar el aprendizaje efectivamente, siendo la base para que el simulador ajuste su respuesta a cada estudiante individualmente, en el marco de los Sistemas Tutores Inteligentes, el modelo del estudiante es uno de los componentes estructurales fundamentales, junto con el modelo del dominio y el modelo pedagógico (Park Woolf, 2009; Bourdeau et al., 2010)

Desde una perspectiva funcional, el modelo de usuario cumple tres roles complementarios: (a) una función diagnóstica, identificando carencias de conocimiento y áreas de mejora; (b) una función predictiva, estimando la probabilidad de éxito o dificultad del usuario ante distintos tipos de eventos; y (c) una función adaptativa, guiando las decisiones del simulador respecto a qué mostrar, cuándo intervenir y cómo acompañar el proceso de aprendizaje.

El modelo se apoya en un historial de interacciones que actúa como memoria temporal del sistema y como fuente empírica para la actualización continua del resto de sus componentes. De este modo, el perfil de competencias, las características cognitivas y el estado afectivo no operan como elementos aislados, sino como vistas complementarias de un proceso adaptativo unificado (Norvig y Russell, 1995).

7.3.1 Componentes del Modelo

Perfil de Competencias. El perfil de competencias representa el nivel estimado de dominio del usuario en cada una de las áreas clave de la gestión de proyectos. Se modela como un vector continuo:

$$C = [c_{tiempo}, c_{presupuesto}, c_{stakeholders}, c_{riesgos}, c_{equipo}, c_{calidad}]$$

Donde cada $c_i \in [0, 1]$ representa el nivel de competencia estimado:

0.0 - 0.3: Novato (comprensión superficial)

0.3 - 0.6: Intermedio (comprensión aplicable)

0.6 - 0.9: Avanzado (dominio consistente)

0.9 - 1.0: Experto (transferencia creativa)

Este perfil no se utiliza únicamente con fines descriptivos, sino como insumo operativo para la selección de eventos, el ajuste de dificultad y la ponderación de la función objetivo del sistema. Las variaciones en cada competencia influyen directamente en la exposición futura a desafíos específicos, permitiendo una progresión diferenciada por área (Park Woolf, 2009).

Actualización del Perfil. El nivel de cada competencia se actualiza de forma incremental en función del desempeño observado (Mitchell, 1997):

$$c_i(t+1) = c_i(t) + \eta \cdot [desempeño_{observado} - c_i(t)]$$

Donde η es tasa de aprendizaje que decrece con el tiempo, favoreciendo ajustes rápidos en las primeras interacciones y actualizaciones más conservadoras en fases avanzadas.

Historial de Interacciones. Constituye un registro estructurado que documenta de forma detallada la relación del usuario con el simulador. En él se almacenan los eventos experimentados junto con sus características, las decisiones tomadas por el usuario, las evaluaciones obtenidas en cada caso (como el desempeño asociado a cada evento), así como el tiempo de respuesta y la longitud de los textos ingresados; este registro permite seguir la evolución temporal de las competencias del usuario, ofreciendo una visión longitudinal de su progreso a lo largo de múltiples sesiones (Park Woolf, 2009).

El análisis de este historial proporciona información valiosa para comprender el comportamiento del jugador: Permite identificar patrones recurrentes, como la tendencia a priorizar el costo por encima del tiempo; detectar sesgos sistemáticos, por ejemplo, si el usuario subestima de forma constante los riesgos y evaluar la evolución de sus habilidades, como determinar si ha mejorado en la gestión de los interesados a partir de la tercera sesión. En conjunto, este componente dota al sistema de la capacidad de realizar un seguimiento adaptativo y personalizado, ajustando el desarrollo de la simulación a la trayectoria real del aprendizaje del usuario (Park Woolf, 2009).

Características Cognitivas. Las características cognitivas describen el estilo de aprendizaje y procesamiento del usuario mediante variables cualitativas y semicuantitativas:

- `Orientación: {analítico, intuitivo, experimental},`
- `Velocidad_aprendizaje: {lenta, media, rápida},`

- `Tolerancia_ambigüedad: [0, 1],`
- `Preferencia_Retroalimentacion: {conciso, balanceado, detallado},`
- `Nivel_reflexión: {superficial, intermedio, profundo}`

Estas variables no modifican directamente las métricas de desempeño del proyecto, pero condicionan la forma en que el sistema presenta la información, regula el ritmo de la simulación y define el estilo de retroalimentación pedagógica. De este modo, el simulador adapta la experiencia a distintos perfiles cognitivos sin alterar los objetivos de aprendizaje subyacentes (Bourdeau et al., 2010).

Estado Afectivo Estimado. El estado afectivo estimado representa una inferencia continua del estado emocional del usuario, construida a partir de señales indirectas observables durante la interacción. Se modela mediante las siguientes variables:

- `Compromiso_actual: [0, 1],`
- `Nivel_frustración: [0, 1],`
- `Confianza_percibida: [0, 1],`
- `Motivación_intrínseca: [0, 1]`

Estas estimaciones cumplen un rol regulador y preventivo dentro del sistema. Sus valores se integran a los mecanismos de detección temprana de frustración excesiva, desmotivación o riesgo de abandono, habilitando intervenciones adaptativas automáticas —como la provisión de ayudas contextuales o el ajuste de dificultad— destinadas a sostener el compromiso sin reducir el nivel de desafío cognitivo (Park Woolf, 2009).

7.3.2 Técnicas de Modelado

Redes Neuronales Artificiales para Clustering. Las Redes Neuronales Artificiales constituyen una técnica especialmente conveniente para agrupar usuarios que poseen características similares mediante procesos de *clustering*. La arquitectura de la red de *clustering* propuesta opera en tres capas diferenciadas: la capa de entrada recibe un vector de características del usuario que integra sus competencias en diferentes áreas, su estilo cognitivo y su historial de interacciones; la capa oculta implementa un mecanismo de reducción dimensional mediante un *autoencoder* que comprime la información en una representación más compacta pero preservando las características esenciales; finalmente, la capa de salida asigna al usuario a uno de los k clusters predefinidos según la similitud de su representación latente con los centroides de cada grupo (Haykin, 2009; Bishop, 2006).

Esta aproximación ofrece múltiples ventajas significativas para el sistema: en primer lugar permite la identificación automática de arquetipos de usuarios sin necesidad de definir manualmente las categorías, descubriendo patrones naturales en los datos; en segundo lugar facilita la transferencia de conocimiento entre usuarios similares: si un usuario pertenece a un *clúster* donde cierta estrategia pedagógica ha demostrado ser efectiva, el sistema aplica esa estrategia con mayor confianza; en tercer lugar, proporciona una inicialización más informada para nuevos usuarios que aún no tienen historial suficiente, evitando el problema del arranque en frío que caracteriza a los sistemas de recomendación.

El proceso de implementación sigue una secuencia de cuatro pasos (Goodfellow et al., 2016):

1. Se entrena un *autoencoder* utilizando datos históricos de múltiples usuarios, permitiendo que la red aprenda a representar usuarios en un espacio latente de menor dimensionalidad donde las características más relevantes están resaltadas.
2. Se aplica el algoritmo *k-means* sobre las representaciones latentes obtenidas, identificando grupos naturales de usuarios con perfiles similares.
3. Cuando un nuevo usuario ingresa al sistema, se proyecta su vector de características al espacio latente mediante el *encoder* entrenado y se le asigna al *clúster* cuyo centroide está más cercano según alguna métrica de distancia (típicamente euclidiana o coseno).
4. Los parámetros iniciales del agente de aprendizaje para este nuevo usuario se configuran según las características promedio del *clúster* al que fue asignado, proporcionando un punto de partida sensato que será refinado mediante la interacción personalizada subsecuente.

Redes Bayesianas para Relaciones Causales. Las Redes de Creencia Bayesianas (*Bayesian Belief Networks*, por sus siglas en inglés, BBN) modelan relaciones probabilísticas entre el conocimiento del usuario y su comportamiento observable dentro del simulador, a diferencia de sistemas determinísticos que asumen certeza absoluta, las BBN reconocen que el estado cognitivo real del usuario es inherentemente incierto y solo se determina mediante evidencia indirecta como su desempeño en eventos específicos (Pearl, 1988; Norvig y Russell, 2020).

El diferencial de las BBN radica en su capacidad de realizar inferencia bidireccional, es decir que, por un lado permiten predicción hacia adelante: dado que conocemos (o

estimamos) el nivel de competencia del usuario, podemos calcular la probabilidad de que tenga buen desempeño en el próximo evento; y por otro el lado permiten diagnóstico hacia atrás: si observamos que el usuario tuvo desempeño deficiente, podemos actualizar nuestra creencia sobre su verdadero nivel de competencia mediante el Teorema de Bayes. Esta actualización bayesiana permite que el modelo del usuario se refine continuamente con cada nueva interacción. En términos prácticos, las BBN permiten al agente de aprendizaje tomar decisiones bajo incertidumbre de manera racional.

Por ejemplo, si la red estima que existe 85% de probabilidad de que el usuario necesite asistencia en el próximo evento (basándose en su historial de desempeño y nivel de competencia inferido), el sistema puede activar proactivamente una pista de nivel apropiado. Si el usuario tiene éxito sin usar la pista, esta evidencia actualiza las probabilidades y reduce la estimación de necesidad de ayuda futura, evitando así proporcionar soporte innecesario que podría resultar condescendiente (Pearl, 1988; Norvig y Russell, 2020).

7.3.3 Funcionamiento del Sistema de Inferencia Bayesiana

El sistema de inferencia bayesiana implementado en el agente de aprendizaje opera mediante un ciclo continuo de predicción, observación y actualización que permite refinar progresivamente el modelo del usuario, este proceso se fundamenta en la teoría de la probabilidad condicional y el Teorema de Bayes, proporcionando un marco matemáticamente riguroso para el razonamiento bajo incertidumbre (Pearl, 1988; Norvig y Russell, 2020).

El ciclo comienza con la fase de inferencia predictiva, donde el agente determina proactivamente si el usuario requerirá asistencia antes de enfrentar un nuevo evento. El sistema consulta dos variables clave: la competencia estimada en el área específica y el desempeño promedio en los últimos tres eventos similares; estas variables alimentan una tabla de probabilidad condicional (*Conditional Probability Table*, por sus siglas en inglés, CPT) que codifica el conocimiento experto sobre la relación entre competencia, desempeño histórico y necesidad de ayuda (Pearl, 1988). La CPT establece que si la competencia es inferior a 0.3 y el desempeño reciente menor a 0.4, la probabilidad de necesitar asistencia es 85%; si la competencia está entre 0.3 y 0.6 con desempeño menor a 0.6, la probabilidad es 55%; en otros casos, apenas 15%. Si la probabilidad calculada supera el umbral de 70%, el agente activa proactivamente un pista de nivel 2, fundamentado en la teoría de la zona de desarrollo próximo de Vygotsky (1978).

La fase de actualización de creencias se activa después de que el usuario completa el evento. El sistema implementa el Teorema de Bayes para actualizar sus creencias sobre el estado cognitivo del usuario basándose en el desempeño observado. Formalmente:

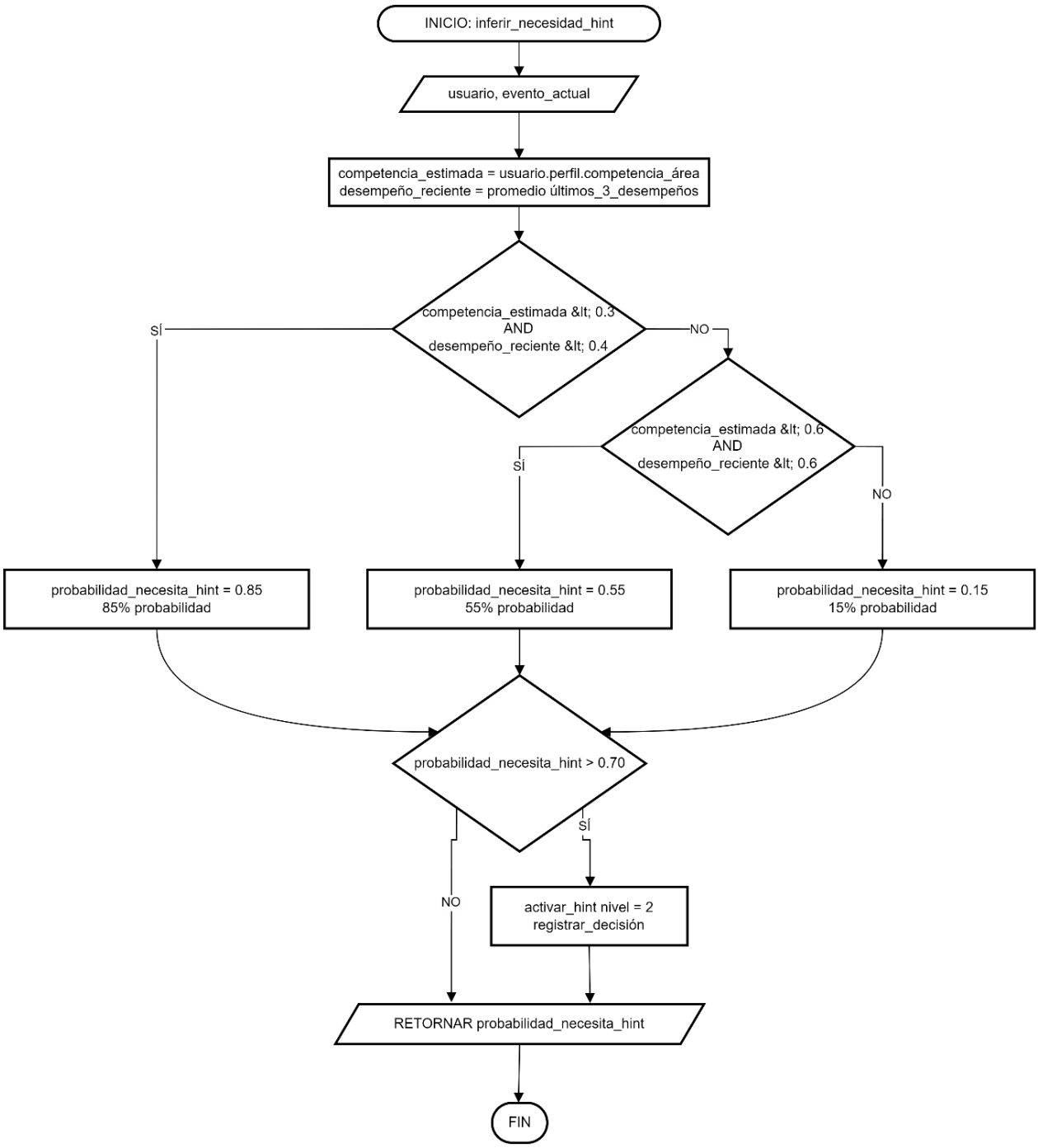
$$P(C|D) = \frac{P(D|C) \cdot P(C)}{P(D)}$$

Donde C representa competencia y D el desempeño observado. El sistema calcula la verosimilitud $P(D|C)$, combina con el prior $P(C)$ y normaliza mediante la evidencia $P(D)$ para obtener la probabilidad posterior $P(C|D)$, que representa la creencia actualizada incorporando la nueva evidencia (Jaynes, 2003). Esta competencia posterior reemplaza la estimación previa en el perfil del usuario, completando el ciclo de actualización.

El ciclo completo de retroalimentación integra ambas fases en un proceso continuo donde cada iteración produce: una predicción más precisa basada en un modelo refinado, una intervención pedagógica apropiada, y una actualización del modelo con evidencia reciente. Este enfoque bayesiano maneja explícitamente la incertidumbre, actualiza creencias de manera óptima, permite combinar múltiples fuentes de evidencia coherentemente, y proporciona trazabilidad sobre las decisiones del sistema. La Figura 5 ilustra cómo el sistema estima la probabilidad de que el usuario necesite asistencia evaluando su competencia y desempeño reciente mediante reglas basadas en probabilidades condicionales, la Figura 6 muestra cómo el agente actualiza esa competencia estimada aplicando el Teorema de Bayes, combinando la evidencia observada con sus creencias previas; finalmente, la Figura 7 integra la inferencia predictiva, la interacción con el usuario y la actualización bayesiana dentro de un ciclo continuo que ajusta el modelo del agente en cada iteración.

Figura 5

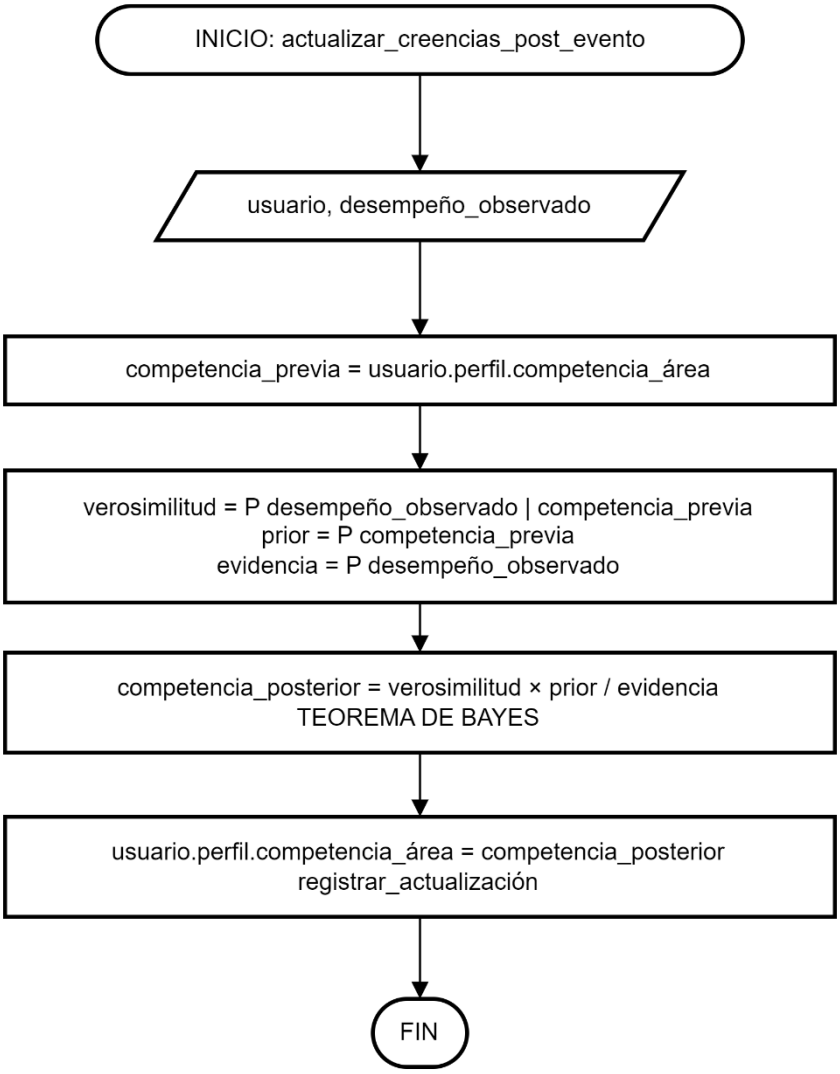
Diagrama de Flujo del Proceso de Inferencia de Necesidad de Asistencia



Nota. El diagrama muestra cómo el sistema estima la probabilidad de que el usuario requiera asistencia, evaluando su competencia y desempeño reciente mediante reglas basadas en probabilidades condicionales. Figura de elaboración propia.

Figura 6

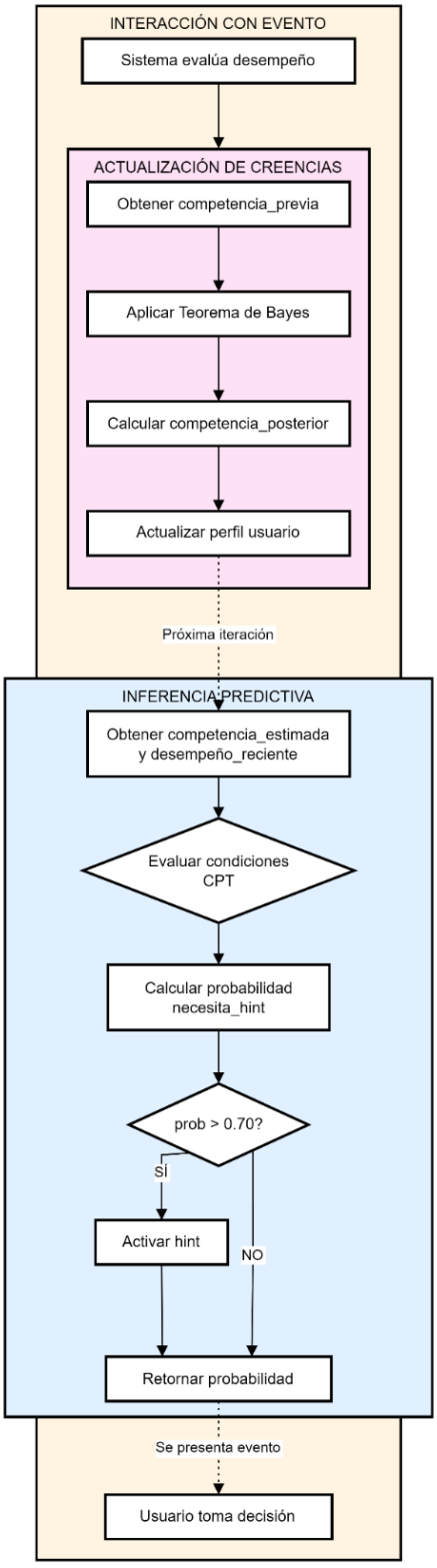
Diagrama de Flujo del Proceso de Actualización Bayesiana de Creencias



Nota. El diagrama ilustra cómo el agente actualiza la competencia estimada del usuario aplicando el Teorema de Bayes para combinar evidencia observada con creencias previas. Figura de elaboración propia.

Figura 7

Ciclo Completo de retroalimentación del Sistema de Aprendizaje Adaptativo



Nota. El diagrama integra la inferencia predictiva, la interacción con el usuario y la actualización bayesiana en un ciclo continuo que ajusta el modelo del agente en cada iteración. Figura de elaboración propia.

7.4 Experiencia de Entrenamiento

La experiencia de entrenamiento constituye el conjunto de datos que el agente utiliza para aprender y mejorar su comportamiento mediante la interacción continua con usuarios, en el simulador propuesto, esta experiencia no se encuentra predefinida de forma estática, sino que se genera dinámicamente a partir de la interacción continua con los usuarios, integrándose de manera directa con el modelo de usuario y su historial de interacciones (Barto y Sutton, 2018).

La experiencia adopta dos modalidades complementarias. La experiencia directa, la cual se adquiere durante la interacción con el usuario actual, registrando secuencias de tuplas del tipo $\{estado_t, acción_t, resultado_t, estado_{t+1}\}$ que capturan el estado previo del simulador, la decisión tomada, la recompensa obtenida según la función objetivo y el estado resultante tras la interacción. Esta modalidad permite una personalización inmediata y específica, aunque presenta una curva de aprendizaje inicial más lenta y el riesgo de desempeño subóptimo en las primeras etapas debido al problema del arranque en frío (Barto y Sutton, 2018).

De manera complementaria, la experiencia indirecta se construye a partir de datos históricos de interacciones previas con múltiples usuarios. Esta información permite inicializar el comportamiento del agente de forma informada, evitando errores ya conocidos y acelerando el proceso de adaptación. No obstante, esta modalidad conlleva el riesgo de introducir sesgos si los datos históricos no resultan representativos del perfil del usuario actual (Barto y Sutton, 2018).

El enfoque adoptado por el simulador combina ambas modalidades mediante un esquema de aprendizaje mixto, que utiliza datos históricos para un preentrenamiento inicial y posteriormente refina el comportamiento del sistema a través de aprendizaje en línea (*online fine-tuning*) durante la interacción real. Esta estrategia maximiza las ventajas de cada aproximación mientras mitiga sus limitaciones, permitiendo un equilibrio entre estabilidad inicial y personalización progresiva.

El simulador emplea el aprendizaje por refuerzo como paradigma principal, requiriendo únicamente señales de recompensa diferidas en lugar de etiquetas explícitas de calidad para cada interacción. Este enfoque reduce significativamente el costo de anotación experta y resulta especialmente adecuado para dominios complejos y abiertos como la gestión de proyectos. La complejidad del aprendizaje depende críticamente de tres factores: la tolerancia al error ε(con

complejidad proporcional a $1/\varepsilon^2$), el número de características F(con complejidad exponencial en el peor caso debido a la maldición de la dimensionalidad) y el tamaño del conjunto de reglas Rnecesarias para expresar el conocimiento objetivo (Barto y Sutton, 2018).

La gestión de esta complejidad se aborda mediante técnicas de reducción dimensional y selección cuidadosa de características relevantes, priorizando aquellas variables del modelo de usuario y del historial de interacciones que aportan mayor valor predictivo y pedagógico. De este modo, el sistema mantiene un balance entre expresividad, adaptabilidad y tratabilidad computacional, garantizando un aprendizaje eficiente y escalable (Friedman et al., 2009).

7.5 Detección de Patrones

La detección de patrones constituye la base del razonamiento inteligente del simulador: es el proceso mediante el cual el agente analiza los datos de interacción para reconocer regularidades y relaciones significativas que expliquen el comportamiento de los usuarios, este mecanismo permite que el sistema evolucione desde un modelo reactivo, que responde ante eventos aislados, hacia un modelo adaptativo capaz de aprender principios generales sobre cómo aprenden los individuos, aplicándolos en nuevos contextos. En otras palabras, el simulador no solo observa, sino que razona sobre lo que observa, aprende de los patrones y los utiliza para anticipar comportamientos, personalizar la experiencia educativa y optimizar la toma de decisiones pedagógicas (Han et al., 2012).

7.5.1 Patrones Predictivos y Anticipación

Los patrones alcanzan su máximo valor cuando se traducen en predicciones verificables, el sistema utiliza los datos recolectados para estimar el comportamiento futuro de los usuarios y decidir cuándo, cómo y con qué intensidad intervenir, por ejemplo, mediante análisis de regresión o modelos probabilísticos, el simulador estima el desempeño futuro a partir de las primeras interacciones:

$$Desempeño_{\{sesión\ 10\}} = f\left(características_{\{sesiones\ 1-3\}}\right)$$

Entre las variables predictivas más relevantes se encuentran la diversidad conceptual, la consistencia en la aplicación de buenas prácticas, la velocidad de mejora y la receptividad a la retroalimentación. Con base en ellas, el sistema detecta usuarios en riesgo de estancamiento y activar estrategias compensatorias como andamiaje adicional, ejemplos guiados o refuerzo de conceptos críticos antes de que se pierda la motivación (Han et al., 2012).

Asimismo, el modelo estima probabilidades de intervención adaptativa $P(NecesitaHint_{proximoturno})$ a partir de variables como el tiempo de respuesta, la longitud y complejidad de los textos ingresados, el desempeño reciente y la dificultad del evento actual. Cuando dicha probabilidad supera un umbral determinado, el simulador interviene de manera automática, ofreciendo ayuda contextual sin requerir una solicitud explícita por parte del usuario. Este mecanismo asiste oportunamente al estudiante, evitando bloqueos cognitivos o frustración excesiva (Park Woolf, 2009).

De manera complementaria, las predicciones sobre áreas problemáticas habilitan la personalización progresiva de la secuencia de aprendizaje. Por ejemplo, si el sistema identifica un perfil técnico con baja competencia en negociación, ajusta gradualmente los desafíos para fortalecer la confianza del usuario antes de exponerlo a eventos de alta carga interpersonal. Este razonamiento predictivo transforma al simulador en un agente proactivo, capaz de anticipar dificultades, prevenir errores recurrentes y maximizar la efectividad del proceso de enseñanza antes de que los problemas se manifiesten (Mitchell, 1997).

El conocimiento extraído de estas predicciones no se limita a correlaciones estadísticas, sino que se estructura como razonamiento pedagógico basado en patrones, materializado en un motor de decisiones justificadas. Este enfoque incrementa la robustez del simulador frente a variaciones de contexto y diversidad de perfiles de usuario. Para ello, el sistema combina tres formas principales de inferencia.

En primer lugar, emplea razonamiento inductivo, mediante el cual generaliza reglas a partir de datos observados. Por ejemplo, si una proporción elevada de usuarios con baja competencia en gestión de riesgos falla sistemáticamente en eventos de crisis, el agente infiere una relación general que ajusta automáticamente la dificultad futura (Mitchell, 1997).

En segundo término, recurre al razonamiento por analogía, identificando similitudes entre el comportamiento del usuario actual y el de usuarios anteriores. Cuando se detecta que un nuevo usuario sigue un patrón similar al de un perfil histórico que mejoró tras una determinada secuencia de eventos, el sistema replica dicha trayectoria como estrategia pedagógica (Han et al., 2012).

El simulador incorpora razonamiento causal, orientado a comprender relaciones de causa y efecto que fundamentan las decisiones pedagógicas. Por ejemplo, si se observa que promover explícitamente la comunicación proactiva incrementa la satisfacción del cliente, el

sistema no solo registra la correlación, sino que refuerza ese concepto en eventos futuros, explicando el porqué de su impacto. El conjunto de estos mecanismos permite que el simulador no solo detecte patrones de comportamiento, sino que los interprete y utilice de forma explicable para guiar al usuario, consolidando su rol como un agente educativo adaptativo, predictivo y pedagógicamente fundamentado. Desde esta perspectiva, el sistema deja de reaccionar a errores para anticipar trayectorias de aprendizaje.

7.5.2 Minería de Datos Educacionales

La minería de datos educacionales constituye la infraestructura analítica del sistema, permitiéndole procesar grandes volúmenes de información provenientes de las interacciones del usuario; como decisiones tomadas; los tiempos de respuesta; la receptividad al retroalimentación y la evolución del desempeño, para extraer patrones estadísticamente relevantes que fundamenten decisiones pedagógicas informadas (Romero y Ventura, 2020).

Los resultados de este análisis sistemático se organizan en cuatro categorías con valor pedagógico directo: (a) Los patrones de éxito, que identifican factores comunes entre usuarios de alto rendimiento, revelando características como la aplicación temprana de buenas prácticas, la diversidad conceptual en decisiones iniciales o la rápida adaptación observable tras recibir retroalimentación, esta información permite al sistema reconocer perfiles prometedores y calibrar apropiadamente el nivel de desafío para mantenerlos en su zona de desarrollo próximo; (b) los patrones de dificultad, que señalan puntos críticos donde la mayoría de los usuarios comete errores sistemáticos, identificando cuellos de botella que justifican el rediseño de la experiencia mediante andamiaje adicional o la introducción de ayudas contextuales preventivas; (c) las secuencias efectivas, que describen órdenes específicos de presentación de eventos que maximizan el aprendizaje, revelando dependencias implícitas entre competencias que no fueron evidentes durante el diseño inicial del sistema, por ejemplo, el análisis podría demostrar que abordar planificación antes que gestión de interesados facilita posteriormente la comprensión de análisis de riesgos y (d) las señales tempranas, que constituyen indicadores detectables en las primeras sesiones que predicen con precisión estadística el rendimiento futuro o el riesgo de abandono, habilitando intervenciones proactivas antes de que problemas potenciales se manifiesten plenamente. Con estas métricas empíricas, el agente toma decisiones fundamentadas en evidencia acumulada, garantizando que cada usuario reciba un recorrido educativo ajustado a su evolución real y no a un esquema pedagógico rígido predefinido (Han et al., 2012; Romero y Ventura, 2020).

7.5.3 Clustering, Clasificación e Identificación de Arquetipos de Usuario

El simulador incorpora técnicas de análisis de patrones para identificar regularidades en el comportamiento de los usuarios y utilizar ese conocimiento con fines adaptativos y pedagógicos. Entre estas técnicas, el *clustering* cumple un rol central al permitir agrupar usuarios con estrategias, estilos cognitivos y patrones de decisión similares sin requerir etiquetas previas. A diferencia de los enfoques de clasificación supervisada, el *clustering* descubre estructuras latentes directamente a partir de los datos de interacción, lo que resulta especialmente adecuado en contextos educativos abiertos y exploratorios (Jain, 2010).

El algoritmo más utilizado en este contexto es *k-means*, el cual itera asignando usuarios a centroides en función de la distancia euclidiana entre sus vectores de características —como tiempos de respuesta, preferencias decisionales, desempeño por área y estilo de comunicación— y recalcula dichos centroides hasta alcanzar estabilidad. Aplicado al simulador, este enfoque permite identificar arquetipos cognitivos emergentes, tales como usuarios analíticos y sistemáticos, caracterizados por decisiones precisas pero lentas y guiadas por datos cuantitativos; usuarios intuitivos rápidos, que priorizan la velocidad de respuesta con escaso análisis previo; y perfiles experimentales, que exploran múltiples rutas antes de consolidar una estrategia estable (Jain, 2010).

La identificación de estos arquetipos habilita múltiples mecanismos adaptativos. En primer lugar, permite inicializar el modelo de nuevos usuarios a partir del clúster más cercano, mitigando el problema del arranque en frío y reduciendo la exposición inicial a eventos mal calibrados. En segundo lugar, posibilita la transferencia de conocimiento entre usuarios con comportamientos similares, reutilizando secuencias de eventos y estrategias que demostraron ser efectivas en perfiles equivalentes. Finalmente, habilita intervenciones pedagógicas diferenciadas a nivel grupal, ajustando el grado de autonomía, el tipo de retroalimentación y la complejidad de los desafíos presentados (Jain, 2010).

De manera complementaria, el sistema emplea técnicas de clasificación y perfilado supervisado para organizar a los usuarios en categorías predefinidas de desempeño y progreso. Estas técnicas construyen una taxonomía explícita que facilita tanto la interpretación del comportamiento del usuario como la explicación pedagógica de las decisiones del sistema. Los árboles de decisión resultan particularmente valiosos en este sentido, ya que permiten segmentar usuarios mediante reglas claras y transparentes (Quinlan, 1993), mientras que las

redes neuronales ofrecen una mayor capacidad para capturar relaciones complejas entre variables, aunque a costa de una menor interpretabilidad.

En conjunto, el *clustering* y la clasificación operan de forma complementaria dentro del simulador, mientras el *clustering* explora y revela patrones emergentes no anticipados, la clasificación estructura y formaliza el conocimiento obtenido, permitiendo una adaptación más precisa, justificable y coherente con los objetivos pedagógicos del sistema.

7.6 Módulo de Adaptación y Tutoría

El módulo de tutoría es el componente que genera las interacciones pedagógicas adaptadas al usuario, utilizando el diagnóstico del modelo de usuario para ajustar la instrucción según sus necesidades específicas.

7.6.1 Arquitectura del Módulo Tutor

El módulo de tutoría constituye el componente responsable de generar interacciones pedagógicas personalizadas, utilizando el diagnóstico del modelo de usuario para ajustar dinámicamente la instrucción según necesidades específicas identificadas. Este módulo transforma la evaluación técnica del desempeño en comunicación educativa efectiva que promueve reflexión metacognitiva y desarrollo de competencias (Bourdeau et al., 2010; Park Woolf, 2009).

El tutor integra cuatro componentes funcionales que operan secuencialmente. El Planificador de Intervenciones determina cuándo y qué tipo de intervención pedagógica es necesaria basándose en señales del modelo de usuario: si detecta confusión persistente, activa andamiaje adicional; si identifica dominio consolidado, reduce soporte para promover autonomía; si reconoce estancamiento, introduce variabilidad estratégica. Este componente implementa lógica de alto nivel sobre timing y tipo de intervención apropiada, se basa en los principios de andamiaje progresivo (Vygotsky, 1978).

El Generador de Contenido Adaptativo produce el contenido específico de la intervención: retroalimentación explicativo que conecta decisiones con consecuencias mediante relaciones causales, Pistas Graduadas que proporcionan ayuda progresiva sin revelar respuestas directamente, explicaciones conceptuales que abordan malentendidos fundamentales, o refuerzo positivo que reconoce aplicación correcta de principios. Este componente transforma intenciones pedagógicas en texto concreto contextualizado al escenario actual (Park Woolf, 2009).

El Selector de Estrategia Pedagógica determina cómo comunicar el contenido según el perfil cognitivo del usuario: estilo interrogativo que promueve descubrimiento mediante preguntas orientadoras para usuarios reflexivos, estilo explicativo con relaciones causales detalladas para usuarios analíticos, estilo sugestivo con ejemplos concretos para usuarios prácticos, o método socrático que expone contradicciones para usuarios con exceso de confianza. Esta selección estratégica maximiza la probabilidad de que el mensaje pedagógico sea recibido efectivamente según preferencias individuales de aprendizaje (Graesser et al., 2005).

El Gestor de Diálogo mantiene coherencia conversacional a lo largo de múltiples interacciones, recordando temas previamente discutidos, evitando repetición innecesaria de conceptos ya explicados, y construyendo progresivamente sobre comprensiones establecidas. Este componente garantiza que el retroalimentación no sea una serie de mensajes desconectados sino una narrativa pedagógica cohesiva que evoluciona con el usuario (Graesser et al., 2005).

7.6.2 Tipos de Intervenciones Tutoriales

El módulo tutor implementa un conjunto acotado de intervenciones pedagógicas que se activan en función del estado del usuario, su desempeño reciente y los objetivos pedagógicos pendientes. Estas intervenciones se inspiran en prácticas documentadas en Sistemas Tutores Inteligentes, donde la retroalimentación y el soporte se ajustan de manera progresiva para promover aprendizaje activo sin generar dependencia (Graesser et al., 2005; Park Woolf, 2009).

retroalimentación Correctiva. La retroalimentación correctiva informa consecuencias observables de una decisión y explica, de forma causal, por qué se produjo ese efecto dentro del modelo del simulador. Su finalidad es conectar la acción con la teoría subyacente y promover reflexión metacognitiva. En entornos tutoriales con diálogo, este tipo de feedback resulta clave para transformar un resultado en aprendizaje, evitando que la experiencia se reduzca a un "juego de métricas" (Graesser et al., 2005).

Estructura.

$$[Resultado\ observable] + [Explicación\ causal] + [Pregunta\ reflexiva]$$

Ejemplo. La moral del equipo disminuyó un 15% tras tu decisión. Esto ocurrió porque delegar una tarea crítica sin supervisión a un recurso junior genera inseguridad en el equipo.

¿Qué acciones podrías haber tomado para mitigar este riesgo mientras delegas la responsabilidad?

Pistas Graduadas. Las pistas graduadas constituyen un mecanismo de ayuda progresiva ante señales de dificultad, con el objetivo de sostener el avance sin revelar la "respuesta correcta" de forma inmediata. Este enfoque es consistente con el principio de andamiaje en ITS, donde la asistencia se incrementa por niveles y se ajusta según la respuesta del estudiante (Park Woolf, 2009).

Nivel 1 - Orientación General. "Recuerda considerar no solo los aspectos técnicos sino también el impacto en las personas involucradas."

Nivel 2 - Pregunta Orientadora. ¿Cómo crees que esta decisión afectará la relación con el interesado principal del proyecto?

Nivel 3 - Ejemplo Análogo. En una situación similar anterior, usuarios exitosos mencionaron la importancia de comunicación anticipada con interesados antes de implementar cambios.

Nivel 4 - Explicación Directa. Es fundamental comunicar proactivamente con el cliente cuando hay cambios que afectan el alcance. Esto mantiene confianza y permite ajustar expectativas.

Andamiaje. El andamiaje (*Scaffolding*) es un mecanismo de soporte temporal mediante el cual el sistema guía al usuario en las primeras etapas del aprendizaje y retira gradualmente dicha ayuda conforme aumenta su autonomía, su propósito es facilitar la adquisición de competencias sin generar dependencia, promoviendo la transición del control externo al autocontrol cognitivo. Este principio se encuentra estrechamente vinculado a la Zona de Desarrollo Próximo, donde el aprendizaje se maximiza cuando el apoyo se ajusta a la brecha entre lo que el estudiante puede hacer solo y lo que puede lograr con guía (Vygotsky, 1978).

En la fase inicial, el agente proporciona una estructura explícita que guía la reflexión del usuario paso a paso, indicando de forma clara los criterios que deben considerarse antes de decidir, como el impacto en el tiempo, el presupuesto y los interesados. A medida que el usuario avanza hacia la fase intermedia, esta orientación se reduce progresivamente y se limita a recordatorios generales que fomentan una participación más activa y consciente, incentivando al estudiante a integrar por sí mismo las distintas dimensiones del problema. Finalmente, en la fase avanzada, el soporte desaparece por completo y el usuario toma decisiones de manera

autónoma, aplicando los principios interiorizados en las etapas previas sin intervención directa del sistema. Este esquema gradual asegura una transición natural desde la guía estructurada hacia la autonomía plena, consolidando un aprendizaje adaptativo, reflexivo y autorregulado.

Refuerzo Positivo. Reconocimiento explícito de decisiones correctas: "Excelente consideración de los interesados. Mencionaste específicamente cómo la decisión afecta al cliente, al equipo y al sponsor. Este análisis multidimensional es característico de gerentes efectivos", el refuerzo debe ser específico (qué hizo bien) no genérico ("bien hecho"), cuando se detecten conceptos mal entendidos, se debe intervenir con explicación y redefinición de los conceptos (Kirkpatrick y Kirkpatrick, 2006):

```
SI usuario.errores_consecutivos_concepto_X ≥ 3

ENTONCES Activar_módulo_remediación:

      - Explicación_conceptual_básica(concepto_X)

      - Ejemplo_concreto(concepto_X)

      - Evento_simplificado_que_ejercite(concepto_X)

      - Verificar_comprensión()
```

7.6.3 Estrategias Pedagógicas

Además del tipo de intervención, el tutor selecciona una estrategia pedagógica acorde al nivel y estilo del usuario. Esta selección es consistente con el enfoque de modelos tutoriales adaptativos, donde el cómo se enseña es tan relevante como el qué se enseña.

Enseñanza Directa. Implica una explicación clara y estructurada de conceptos, debe usarse en casos de usuarios completamente novatos en un área; para explicar un concepto fundamental que requiere poseer una base sólida o tras múltiples intentos fallidos (Graesser et al., 2005; Park Woolf, 2009).

Ejemplo: "El triángulo de hierro de gestión de proyectos establece que tiempo, costo y alcance están interrelacionados. Si modificas uno, inevitablemente afectas a los otros dos..."

Aprendizaje por Descubrimiento. Significa guiar al usuario mediante preguntas para que este construya conocimiento, debe usarse en casos de usuarios con conocimiento previo suficiente; para promover el pensamiento crítico o para fomentar transferencia activa (Graesser et al., 2005; Park Woolf, 2009).

Ejemplo: "Observaste que el cliente está insatisfecho a pesar de que el proyecto va a tiempo. ¿Qué factores además del cronograma podrían estar influyendo en su percepción?"

Aprendizaje por Casos. Se basa en presentar situaciones reales análogas, recomendables cuando el usuario es intermedio pero necesita contexto; cuando se quiera ilustrar la aplicación práctica de algunos conceptos o para facilitar la transferencia de conceptos abstractos a contextos reales (Graesser et al., 2005; Park Woolf, 2009).

Ejemplo: "En el proyecto X de desarrollo de software, el equipo enfrentó una situación similar. Decidieron... ¿Qué elementos de esa solución podrían ser relevantes para tu situación actual?"

Aprendizaje Socrático. Basado en un diálogo mediante generar preguntas que expongan contradicciones, aplicable para usuarios avanzados con confianza excesiva; para desafiar determinadas suposiciones implícitas o para promover una reflexión más profunda (Graesser et al., 2005; Park Woolf, 2009).

Ejemplo:

Usuario: "Incrementaría el equipo para acelerar el proyecto."

Sistema: "Mencionas que el proyecto va retrasado por falta de recursos. Sin embargo, también indicaste que el equipo actual tiene problemas de coordinación. ¿Cómo crees que agregar más personas afectará esos problemas de coordinación?"

7.6.4 Reglas de Acción ante Respuestas del Usuario

El tutor del simulador debe contar con estrategias de intervención predefinidas que se activen en función del tipo de respuesta proporcionada por el usuario. Cuando la respuesta es correcta y completa, el sistema aplica un refuerzo positivo, incrementa levemente la dificultad de los desafíos posteriores y reduce la frecuencia de pistas, promoviendo mayor autonomía (Park Woolf, 2009). En los casos de respuestas parcialmente correctas, el tutor reconoce los aciertos logrados, ofrece retroalimentación en forma de pregunta orientadora y genera consecuencias mixtas que reflejan tanto los aspectos correctamente resueltos como aquellos que requieren ajuste.

Si la respuesta es incorrecta pero presenta un razonamiento coherente, el sistema valida el proceso cognitivo del usuario, propone una corrección guiada mediante pistas de nivel intermedio y brinda la oportunidad de revisar la decisión tomada (Graesser et al., 2005). En

cambio, ante respuestas incorrectas y superficiales, el tutor interviene con retroalimentación directa, activa pistas de mayor nivel, aplica consecuencias negativas dentro de la simulación y reduce la dificultad o inicia mecanismos de remediación. Cuando el usuario manifiesta incapacidad para responder, el sistema proporciona una pista explícita, simplifica el evento y evalúa los conocimientos previos necesarios para continuar. Si la respuesta resulta fuera de contexto, el tutor aclara el enunciado, redirige el foco de atención y da la oportunidad al usuario reformular su respuesta, asegurando que la interacción continúe de manera constructiva y pedagógicamente coherente (Vygotsky, 1978).

7.6.5 Personalización de la Tutoría

Basándose en el modelo cognitivo del usuario y en principios de instrucción adaptativa (Park Woolf, 2009). el sistema modifica su comportamiento al perfil cognitivo y emocional del usuario, ajustando el tipo de retroalimentación, el tono comunicativo y la dificultad de las tareas: En los casos de usuarios analíticos, reciben datos cuantitativos, explicaciones causales y más tiempo de reflexión, acompañados por gráficos o visualizaciones; para usuarios intuitivos, se mostraran mensajes breves y orientados a la acción, con ejemplos prácticos y un ritmo más dinámico; para el caso de los usuarios con alta confianza, estos serán desafiados mediante preguntas socráticas, escenarios complejos y retroalimentación que promueve la autocrítica; finalmente, los usuarios con baja confianza recibirán refuerzo positivo frecuente, andamiaje explícito y una progresión de dificultad gradual.

De este modo, el sistema adapta su estilo pedagógico para maximizar la eficacia del aprendizaje en función del perfil del jugador.

7.6.6 Proactividad y Comportamiento Social

El sistema reacciona no solo ante las acciones del usuario, sino que anticipa necesidades y mantiene una comunicación empática mediante intervenciones proactivas, detecta patrones de inactividad, omisiones o frustración potencial y actúa preventivamente, ofreciendo aclaraciones o ajustando la complejidad antes de que el problema se manifieste. En su dimensión social, el sistema se comunica de manera personalizada y natural, utilizando el nombre del usuario, reconociendo su esfuerzo, expresando comprensión y celebrando los progresos genuinos. Su lenguaje mantiene un tono humano, evitando respuestas rígidas o impersonales (Park Woolf, 2009).

Ejemplo: “Veo que te enfocaste en los aspectos técnicos, lo cual es importante. Sin embargo, el desafío principal era la gestión de expectativas. ¿Cómo podrías abordar también esa dimensión?”

Este enfoque transforma la tutoría en un diálogo genuino, manteniendo el compromiso y la motivación del usuario.

7.6.7 Resumen y Síntesis de Sesión

Al finalizar cada sesión, el sistema genera un resumen personalizado que consolida el progreso y orienta el aprendizaje futuro, el informe incluye un reconocimiento general del desempeño, las fortalezas evidenciadas durante la sesión, las áreas de mejora detectadas, la evolución respecto a sesiones anteriores y algunos objetivos sugeridos para la siguiente interacción (Park Woolf, 2009). Este resumen refuerza la autoconciencia del usuario y ajusta dinámicamente el plan pedagógico, por ejemplo:

Completaste 12 eventos mostrando consistencia en tu enfoque. Mejoraste en análisis de riesgos y control presupuestario, aunque la gestión de interesados sigue siendo un punto que reforzar. Tu competencia general aumentó de 0.58 a 0.64 (+10%). En la próxima sesión abordaremos estrategias de comunicación y negociación de expectativas.

7.7 Algoritmos de Aprendizaje por Refuerzo

Una vez establecidos la arquitectura del agente, las funciones objetivo, el modelo de usuario y el módulo de tutoría, resulta necesario especificar los algoritmos concretos que permiten al agente aprender de la experiencia y optimizar su comportamiento pedagógico mediante interacción continua con usuarios.

7.7.1 Formalización como Proceso de Decisión de Markov

El problema de aprendizaje del agente se modela formalmente como un Proceso de Decisión de Markov (*Markov Decision Process*, por sus siglas en inglés, MDP), estructura matemática que formaliza la toma de decisiones secuenciales bajo incertidumbre que se define mediante la tupla $MDP = (S, A, P, R, \gamma)$, donde cada componente representa un aspecto fundamental del problema de aprendizaje (Barto y Sutton, 2018).

El espacio de estados S captura toda la información relevante sobre el contexto actual mediante aproximadamente 25 dimensiones organizadas en cuatro categorías: seis variables

del proyecto que describen el estado de la simulación (tiempo restante, presupuesto disponible, moral del equipo, satisfacción del cliente, nivel de riesgo, avance de funcionalidades); diez variables del perfil del usuario que representan competencias en diferentes áreas de gestión, nivel de experiencia acumulada, compromiso histórico y preferencias de aprendizaje; cuatro variables de contexto temporal que indican la fase actual del proyecto, turnos transcurridos desde eventos significativos y rachas de desempeño; y cinco dimensiones de historial reciente que codifican mediante *embeddings* los últimos eventos experimentados, capturando patrones temporales relevantes para decisiones actuales (Barto y Sutton, 2018).

El espacio de acciones A define las intervenciones que el agente ejecuta, incluyendo la selección entre aproximadamente 18 categorías de eventos clasificados por área temática y nivel de complejidad, así como ajustes paramétricos como modificación de dificultad, frecuencia de pistas y nivel de andamiaje proporcionado. La función de transición $P(s'|s,a)$ especifica la probabilidad de alcanzar el estado s' tras ejecutar la acción a en el estado s; en el simulador, esta función es determinística para variables del proyecto (las consecuencias de eventos están definidas por reglas del dominio) pero estocástica para el comportamiento del usuario (su respuesta ante eventos tiene componente aleatorio impredecible). La función de recompensa $R(s,a,s')$ cuantifica la deseabilidad de cada transición según la función objetivo multidimensional definida previamente, balanceando progreso pedagógico, compromiso, diversidad experiencial y coherencia narrativa. Finalmente, el factor de descuento γ (típicamente 0.95) modula el balance entre optimizar recompensas inmediatas versus acumuladas a largo plazo, reflejando que el aprendizaje profundo requiere planificación temporal extendida más allá del evento actual (Barto y Sutton, 2018).

7.7.2 Algoritmo Actor-Critic

La implementación emplea un algoritmo Actor-Critic, arquitectura que descompone el problema de aprendizaje en dos componentes complementarios que aprenden simultáneamente. El Actor mantiene la política $\pi(a|s,\theta)$ que mapea estados a distribuciones de probabilidad sobre acciones, donde θ representa parámetros aprendibles (típicamente pesos de una red neuronal); esta política define cómo el agente selecciona acciones en cada situación. El Critic mantiene la función de valor $V(s,w)$ que estima la recompensa acumulada esperada desde cada estado, donde w son parámetros aprendibles; esta función proporciona una línea base para evaluar si acciones específicas fueron mejores o peores que el promedio esperado.

El proceso de actualización opera en cinco pasos tras cada interacción. Primero, el sistema observa la transición completa (s_t, a_t, r_t, s_{t+1}) que incluye el estado inicial, la acción ejecutada, la recompensa inmediata obtenida y el estado resultante. Segundo, el Critic estima los valores $v_t = V(s_t, w) y v_{t+1} = V(s_{t+1}, w)$ para cuantificar cuán buenos eran los estados antes y después de la acción. Tercero, se calcula el error de diferencia temporal $\delta_t = r_t + \gamma \cdot v_{t+1} - v_t$, que mide la discrepancia entre la recompensa realmente obtenida más el valor futuro estimado versus el valor que el Critic había predicho originalmente; este error representa la "sorpresa" del sistema ante el resultado observado. Cuarto, el Critic actualiza sus parámetros mediante $w \leftarrow w + \alpha_{critic} \cdot \delta_t \cdot \nabla_w V(s_t, w)$ para reducir el error de predicción, donde α_{critic} es la tasa de aprendizaje y ∇_w denota el gradiente que indica la dirección de ajuste óptima. Quinto, el Actor actualiza su política mediante $\theta \leftarrow \theta + \alpha_{actor} \cdot \delta_t \cdot \nabla_\theta \log \pi(a_t|s_t, \theta)$ incrementando la probabilidad de acciones con ventaja positiva ($\delta_t > 0$, indicando que la acción fue mejor que el promedio) y decrementando la probabilidad de acciones con ventaja negativa ($\delta_t < 0$, indicando que fue peor que el promedio) (Barto y Sutton, 2018).

7.7.3 Balance Exploración-Explotación

El agente enfrenta continuamente el dilema entre exploración, es decir, probar acciones nuevas o poco frecuentes para descubrir estrategias potencialmente superiores, y explotación, es decir, utilizar acciones que el conocimiento actual indica como efectivas. La estrategia *epsilon-greedy* con decaimiento implementa este balance mediante una probabilidad de exploración que decrece con el tiempo: $\varepsilon(t) = \max(\varepsilon_{min}, \varepsilon_{inicial} \cdot decay^t)$. En cada decisión, el sistema genera un número aleatorio; si este es menor que $\varepsilon(t)$, ejecuta exploración seleccionando una acción aleatoria, de lo contrario ejecuta explotación seleccionando $\arg\max_a \pi(a|s)$, la acción de máxima probabilidad según la política actual. Típicamente se establece $\varepsilon_{inicial} = 0.2$ para explorar activamente durante aprendizaje inicial, $\varepsilon_{min} = 0.05$ para mantener exploración mínima que previene estancamiento en óptimos locales, y $\text{decay} = 0.995$ para reducción gradual que refleja confianza creciente en el conocimiento acumulado (Barto y Sutton, 2018; Park Woolf, 2009).

7.7.4 Aprendizaje por Currículo

La técnica de Aprendizaje por currículo (*curriculum learning*) estructura la progresión de dificultad en niveles secuenciales que el usuario debe dominar antes de avanzar, emulando cómo currículos educativos tradicionales organizan contenido de simple a complejo (Bengio et al., 2009). El Nivel 1 (Fundamentos) dura mínimo tres sesiones, presenta únicamente eventos básicos e introductorios con complejidad máxima 0.4, y requiere que las competencias básicas superen 0.6 para avanzar. El Nivel 2 (Intermedio) dura mínimo cinco sesiones, incluye eventos básicos e intermedios con complejidad máxima 0.7, y exige desempeño consistente superior a 0.65. El Nivel 3 (Avanzado) no tiene límite de duración, habilita todos los eventos incluyendo los de máxima complejidad 1.0, y representa el régimen de operación normal donde el agente adapta continuamente dificultad según evolución del usuario. Esta progresión estructurada previene frustración temprana al garantizar que usuarios adquieran fundamentos sólidos antes de enfrentar desafíos complejos.

7.7.5 Aprendizaje por Transferencia

El Aprendizaje por Transferencia (*transfer learning*) aborda el problema del arranque en frío inicializando nuevos usuarios con conocimiento extraído de usuarios previos (Pan y Yang, 2010). Cuando ingresa un usuario sin historial, el sistema copia los pesos θ y w de un agente pre-entrenado que ha acumulado experiencia con múltiples usuarios anteriores, proporcionando políticas sensatas desde la primera interacción. Adicionalmente, se duplica temporalmente la tasa de aprendizaje ($\alpha_{nuevo} = 2.0 \cdot \alpha_{normal}$) durante sesiones iniciales para que el sistema se adapte rápidamente a las particularidades del nuevo individuo, esta estrategia evita que usuarios experimenten las políticas subóptimas o aleatorias que caracterizarían un agente sin conocimiento previo, mejorando significativamente la calidad de la experiencia inicial.

7.7.6 Adaptación a la No Estacionariedad

Los usuarios evolucionan genuinamente mediante aprendizaje real, generando un entorno no estacionario donde las relaciones entre estados y recompensas cambian con el tiempo; el sistema implementa un detector de cambios de concepto que compara el desempeño promedio de las últimas 10 interacciones (ventana actual) contra las 10 previas (ventana anterior). Si la diferencia absoluta entre ambas medias supera 0.15, indica cambio significativo

que requiere ajuste estratégico; s la media actual excede la previa, señala mejora genuina del usuario, justificando incremento de dificultad del 15% para mantenerlo en zona de desarrollo próximo; si la media actual es inferior, señala deterioro que demanda reducción de dificultad del 10% e incremento de soporte mediante andamiaje adicional. Esta adaptación continua garantiza que el agente permanezca calibrado a la competencia actual del usuario, no a su estado histórico ya superado o deteriorado (Barto y Sutton, 2018; Norvig y Russell, 2021).

7.8 Integración Sistémica y Consideraciones Finales

El sistema de inteligencia artificial del simulador no constituye una colección de módulos independientes sino un organismo cognitivo integrado donde cada componente se entrelaza funcionalmente con los demás, generando propiedades emergentes que trascienden las capacidades de sus partes individuales. Esta integración sistémica transforma al simulador de una herramienta educativa estática en una experiencia de aprendizaje genuinamente adaptativa que evoluciona continuamente en respuesta al comportamiento único de cada usuario (Bourdeau et al., 2010; Park Woolf, 2009).

7.8.1 Arquitectura de Integración Funcional

El flujo operativo del sistema comienza cuando el usuario interactúa con la interfaz del simulador, tomando decisiones mediante texto libre, esta entrada textual es capturada inmediatamente por el módulo de procesamiento de lenguaje natural (PLN), que transforma la cadena de caracteres en una representación semántica estructurada mediante tokenización, análisis sintáctico, reconocimiento de entidades nombradas y clasificación de intención comunicativa (Norvig y Russell, 1995). El resultado de este procesamiento resulta en un vector de características lingüísticas enriquecido con metadatos semánticos, el cual se transmite al sistema de heurísticas de evaluación, que aplica reglas expertas del dominio de gestión de proyectos para identificar buenas prácticas mencionadas, antipatrones presentes, nivel de reflexión demostrado y completitud conceptual de la respuesta.

La evaluación heurística alimenta directamente al agente de aprendizaje, específicamente a su elemento de desempeño, que debe determinar las consecuencias de la decisión del usuario sobre el estado del proyecto simulado. El agente consulta su modelo interno del usuario, representado mediante un modelo probabilístico mantenido mediante redes bayesianas y perfiles de competencia, para contextualizar la evaluación: una respuesta que sería excelente para un novato puede ser apenas satisfactoria para un experto, reflejando expectativas diferenciadas según el nivel de desarrollo (Park Woolf, 2009). Con base en esta evaluación

contextualizada, el agente calcula los cambios en las variables del proyecto (tiempo, presupuesto, moral, riesgo, satisfacción del cliente) mediante las reglas del sistema de eventos, que codifica las dinámicas causales del dominio.

Simultáneamente, el módulo de tutoría genera retroalimentación personalizado que comunica al usuario las consecuencias de su decisión, este retroalimentación no es meramente informativo sino que esta pedagógicamente estructurado: el módulo consulta el modelo del usuario para determinar el estilo de comunicación más efectivo (detallado versus conciso, interrogativo versus explicativo, técnico versus conceptual) y ajusta el tono según el estado emocional inferido (refuerzo positivo si detecta desánimo, desafío constructivo si detecta exceso de confianza). El retroalimentación resultante se presenta al usuario a través de la interfaz, cerrando el ciclo de interacción inmediata (Graesser et al., 2005).

Paralelamente, el elemento de aprendizaje del agente opera en un nivel metacognitivo, analizando la efectividad de las decisiones que el propio sistema tomó: ¿El evento presentado fue pedagógicamente apropiado? ¿El nivel de dificultad fue el correcto? ¿El retroalimentación generó reflexión o confusión? Estas evaluaciones alimentan el módulo de detección de patrones, que identifica regularidades en el comportamiento del usuario mediante técnicas de minería de datos, aplicando clustering para identificar su arquetipo cognitivo, árboles de decisión para clasificar su nivel de competencia y análisis de secuencias para detectar progresión o estancamiento. Los patrones detectados se integran al modelo del usuario mediante actualización bayesiana, refinando continuamente las creencias del sistema sobre el estado cognitivo real del individuo (Han et al., 2012; Romero y Ventura, 2020).

Esta información actualizada modifica los parámetros que guían futuras decisiones del agente, desde los pesos de la función de utilidad, que determina qué evento presentar a continuación; los umbrales que activan pistas automáticos o las probabilidades de exploración versus explotación en el algoritmo Actor-Critic. El algoritmo de aprendizaje por refuerzo ajusta estos parámetros mediante gradiente descendente, buscando maximizar la función objetivo multidimensional que balancea progreso pedagógico, compromiso sostenido, diversidad experiencial y coherencia narrativa. El factor de descuento temporal ($\gamma = 0.95$) en el algoritmo garantiza que el sistema optimice no solo la interacción inmediata sino la trayectoria de aprendizaje a largo plazo (Barto y Sutton, 2018).

El sistema de eventos proporciona al agente el repertorio de situaciones que puede presentar, pero el agente determina dinámicamente cuándo y cómo presentar cada evento

mediante la función de utilidad ponderada. Esta función considera múltiples dimensiones: coherencia narrativa, para que el evento tenga sentido causal dentro del proyecto simulado; pertinencia pedagógica, ya que debe ejercitar competencias que el usuario aún no domina; dificultad apropiada, porque debe estar en la zona de desarrollo próximo del usuario según su nivel actual; diversidad experiencial para aportar variedad evitando repetición monótona y timing narrativo, es decir que debe presentarse en el momento dramáticamente apropiado de la simulación (Vygotsky, 1978).

La integración alcanza su máxima sofisticación en el curriculum learning, donde el sistema estructura la progresión de dificultad en niveles secuenciales que el usuario debe dominar antes de avanzar. El agente monitorea continuamente indicadores de dominio (competencias básicas superiores a 0.6 para avanzar del Nivel 1 al 2, desempeño consistente superior a 0.65 para avanzar del Nivel 2 al 3) y ajusta automáticamente el pool de eventos disponibles según el nivel actual, esta progresión estructurada, combinada con la adaptación continua dentro de cada nivel, genera una experiencia que es simultáneamente personalizada y estructurada (Bengio et al., 2009).

El transfer learning añade otra capa de integración temporal, permitiendo que el conocimiento acumulado con usuarios previos beneficie a nuevos usuarios. Cuando un individuo sin historial ingresa al sistema, el agente inicializa sus parámetros con los pesos de un agente pre-entrenado que ha interactuado con cientos de usuarios anteriores, proporcionando políticas sensatas desde la primera sesión. Sin embargo esta inicialización no es rígida, ya que incluye tasas de aprendizaje elevadas durante sesiones iniciales que permiten una rápida adaptación a las particularidades del nuevo usuario, convergiendo hacia una política genuinamente personalizada que preserva conocimiento generalizable pero ajusta aspectos idiosincráticos.

La adaptación a no estacionariedad representa quizás el aspecto más sofisticado de la integración, reconociendo que el usuario evoluciona genuinamente mediante aprendizaje real, en este punto el detector de cambios de concepto compara ventanas temporales de desempeño para identificar mejoras o deterioros significativos, ajustando proactivamente la dificultad y el nivel de soporte. Si el usuario mejora consistentemente, el sistema incrementa la complejidad de eventos presentados, reduce la frecuencia de pistas, y activa modos de operación más desafiantes. Si el desempeño se deteriora, el sistema reduce temporalmente la dificultad,

incrementa el andamiaje, e incluso activar módulos de remediación que revisan conceptos fundamentales aparentemente no consolidados.

7.8.2 Propiedades Emergentes de la Integración

La integración de los módulos descritos no produce únicamente una suma funcional de componentes, sino un comportamiento sistémico con propiedades emergentes. Estas propiedades no están presentes en ningún módulo de forma aislada, sino que surgen de la interacción coordinada entre procesamiento lingüístico, modelo de usuario, aprendizaje por refuerzo, inferencia bayesiana y tutoría adaptativa. En términos de teoría de sistemas complejos, se trata de fenómenos que emergen de la organización estructural y de los ciclos de retroalimentación internos del sistema (Norvig y Russell, 2021).

Coherencia Adaptativa: El sistema mantiene una narrativa del proyecto internamente consistente, es decir que las consecuencias de decisiones previas afectan situaciones futuras, los interesados recuerdan interacciones pasadas, los recursos evolucionan según su uso, mientras adapta continuamente la dificultad y el tipo de desafíos presentados según la evolución del usuario. Esta combinación de coherencia narrativa y adaptabilidad pedagógica distingue al simulador de experiencias educativas puramente algorítmicas que sacrifican credibilidad por personalización, o de simulaciones realistas pero estáticas que ignoran las necesidades específicas del aprendiz (Norvig y Russell, 2021).

Aprendizaje Bidireccional. No solo el usuario aprende sobre gestión de proyectos mediante la interacción con el simulador, sino que el sistema aprende sobre el usuario mediante la observación de su comportamiento. Esta dualidad genera un fenómeno de coevolución donde las capacidades del usuario y las estrategias del agente se refinan mutuamente; a medida que el usuario desarrolla competencias, el agente descubre qué intervenciones pedagógicas resultan más efectivas para ese perfil específico; a medida que el agente refina sus estrategias, el usuario enfrenta desafíos cada vez mejor calibrados que maximizan su progreso. Esta coevolución acelera el aprendizaje más allá de lo que sería posible con tutorización estática o con práctica no guiada (Barto y Sutton, 2018).

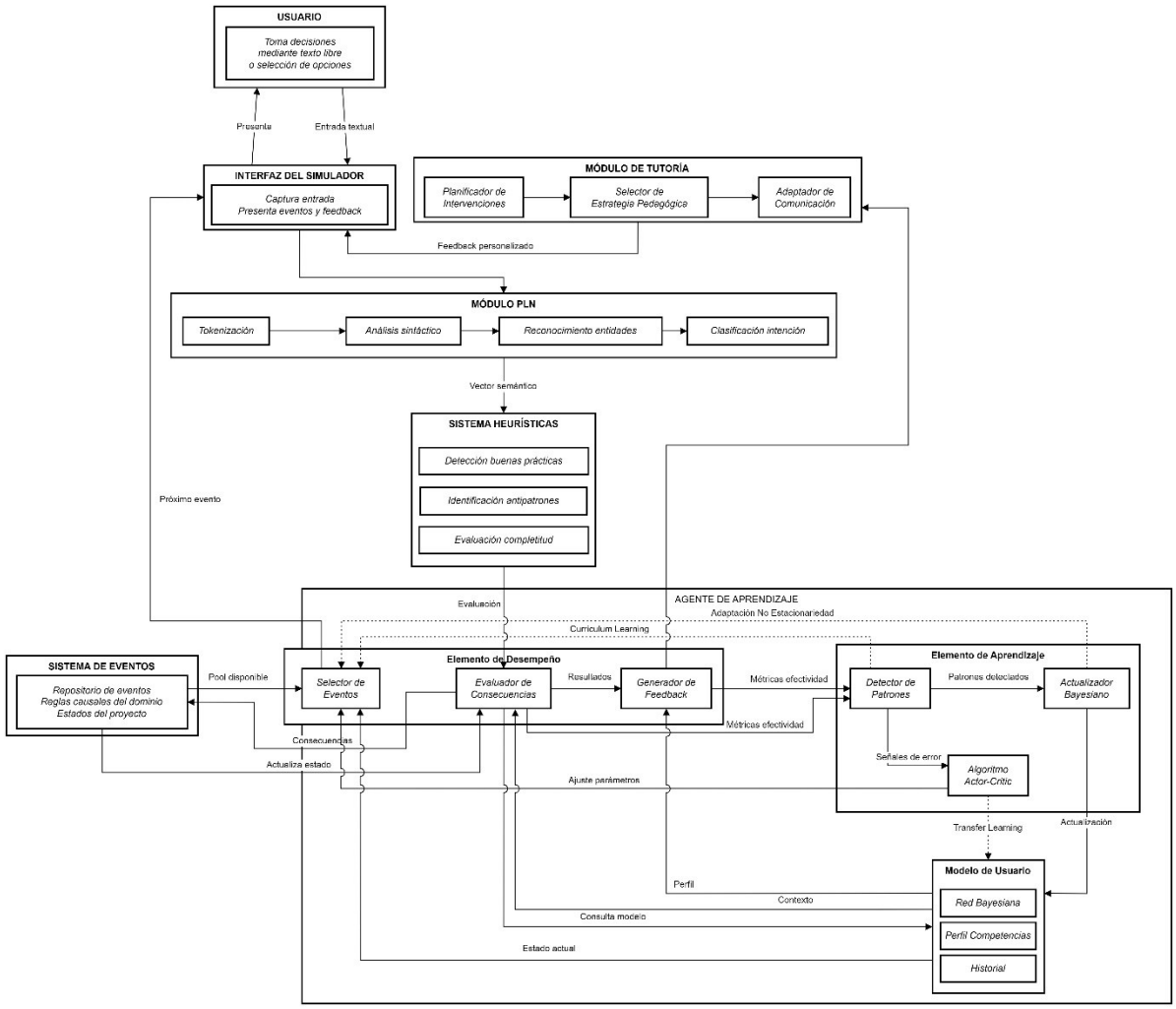
Robustez Mediante Redundancia Funcional. Múltiples componentes del sistema contribuyen a los mismos objetivos mediante mecanismos complementarios, generando robustez ante fallos parciales. La personalización se logra mediante el modelo bayesiano del usuario, mediante clustering que identifica arquetipos, o mediante aprendizaje por refuerzo que optimiza políticas individuales; si uno de estos mecanismos falla o proporciona información

poco confiable, los otros compensan. La evaluación de respuestas integra análisis semántico mediante PLN, aplicación de heurísticas expertas, y comparación con el modelo de competencia del usuario; la convergencia de estas evaluaciones independientes proporciona mayor confiabilidad que cualquier método aislado. Esta redundancia funcional garantiza que el sistema mantenga efectividad pedagógica incluso cuando componentes individuales operan con información incompleta o ruidosa, característica inevitable de sistemas que intentan inferir estados cognitivos internos mediante observación de comportamiento externo (Norvig y Russell, 2021).

La Figura 8 sintetiza visualmente la arquitectura integrada del sistema de inteligencia artificial, ilustrando los flujos de información entre componentes y los ciclos de retroalimentación que permiten aprendizaje continuo tanto del usuario como del agente.

Figura 8

Diagrama integrador completo



Nota. El diagrama sintetiza la arquitectura integrada del sistema de inteligencia artificial, mostrando los flujos de información entre módulos y los ciclos de retroalimentación que permiten el aprendizaje continuo del agente. Las flechas sólidas indican procesos principales y las punteadas representan mecanismos de aprendizaje avanzado. Figura de elaboración propia.

Capítulo 8 – Diseño de Simulador. Fase V: Consideraciones Teóricas de Despliegue y Validación

Este capítulo aborda las consideraciones finales necesarias para materializar el diseño conceptual propuesto en los capítulos anteriores. Aunque el presente trabajo no incluye implementación de software, resulta fundamental establecer el marco metodológico para validar si el sistema cumple sus objetivos pedagógicos, analizar qué componentes del diseño son reutilizables en otros contextos, definir cómo se desplegara el sistema en infraestructura real y especificar qué capacidad de procesamiento demandara su operación.

8.1 Diseño de la Metodología de Validación

La validación del simulador debe responder dos preguntas fundamentales: ¿el sistema enseña efectivamente lo que pretende enseñar? y ¿los estudiantes adquieren habilidades transferibles a situaciones reales? Este proceso se estructura en dos fases temporales diferenciadas que combinan métodos cualitativos y cuantitativos (Kirkpatrick y Kirkpatrick, 2006).

8.1.1 Validación en Dos Fases

La fase formativa ocurre durante el diseño, antes de cualquier implementación. Un panel de cinco a siete gerentes de proyectos certificados evalúa los escenarios diseñados mediante rúbricas que miden autenticidad situacional, complejidad apropiada y relevancia pedagógica en escalas de uno a cinco. Paralelamente, veinte a treinta usuarios piloto interactúan con prototipos para detectar problemas de usabilidad o comprensión que los expertos podrían no anticipar, proporcionando retroalimentación sobre claridad del retroalimentación y dificultad percibida (Campbell y Stanley, 1963; Kirkpatrick y Kirkpatrick, 2006).

La fase sumativa requiere implementación completa y emplea diseño cuasiexperimental robusto con cien participantes divididos aleatoriamente en grupos experimental y control (Campbell y Stanley, 1963). El grupo experimental recibe formación mediante el simulador durante cuatro semanas (doce horas totales); el grupo control recibe formación tradicional con casos escritos e igual carga horaria. Se administran *pre-test* inicial, *post-test* inmediato, *post-test* diferido tres meses después para evaluar retención, y evaluación de transferencia mediante casos completamente nuevos.

8.1.2 Validación del Modelo de Usuario

El modelo bayesiano que infiere conocimiento del usuario debe validarse independientemente comparando sus estimaciones contra evaluaciones independientes ("*ground truth*") administradas por expertos humanos (Anderson y Corbett, 1994). Después de que un usuario completa varios escenarios, el sistema habrá actualizado creencias sobre cinco competencias principales (planificación, riesgos, negociación, resolución de conflictos, comunicación). Paralelamente, evaluadores humanos administran pruebas prácticas que miden las mismas competencias mediante casos escritos calificados con rúbricas específicas. El coeficiente de correlación de Pearson entre ambas evaluaciones debe superar 0.7 para validez predictiva aceptable; correlaciones inferiores revelan patrones de error sistemático que informan ajustes a las probabilidades condicionales de la red bayesiana.

8.1.3 Validación del Componente PLN

El procesamiento de lenguaje natural requiere verificar coherencia narrativa, pertinencia del retroalimentación y robustez ante inputs inusuales. La coherencia se valida ejecutando cien interacciones con contextos definidos, con evaluadores marcando inconsistencias; el objetivo es tasa inferior al 5%. La pertinencia se mide creando cincuenta "decisiones de referencia" para las cuales expertos escriben retroalimentación ideal; la similitud semántica entre retroalimentación del sistema y de expertos (medida mediante coseno de embeddings) debe superar 0.75. La robustez se valida introduciendo cincuenta inputs problemáticos (sin sentido, fuera de dominio, maliciosos); el sistema debe poseer mecanismos de tolerancia a fallos, solicitando aclaración en lugar de generar respuestas erróneas (Jurafsky y Martin, 2025; Manning y Schütze, 1999).

8.1.4 Métricas Cuantitativas de Éxito

Los umbrales predefinidos incluyen: ganancia de conocimiento mínima del 25% (post-test vs. pre-test), diferencia estadísticamente significativa entre grupos ($p < 0.05p$), satisfacción subjetiva superior a 70% positiva, correlación del modelo de usuario $r > 0.7$, coherencia narrativa con menos de 5% inconsistencias, pertinencia del retroalimentación con similitud superior a 75%, tasa de completitud superior a 80%, y latencia de respuesta inferior a cinco segundos para mantener interacción fluida. La Tabla 1 presenta un conjunto de métricas de éxito que establecen los criterios cuantitativos para evaluar el aprendizaje, la coherencia

narrativa, la calidad de la retroalimentación y el desempeño técnico del sistema (Aldrich, 2009; Park Woolf, 2009).

Tabla 1

Métricas utilizadas para medir el éxito del sistema

Categoría	Métrica	Objetivo
Aprendizaje	Ganancia conocimiento	>25%
Aprendizaje	Diferencia experimental vs control	p<0.05
Aprendizaje	Satisfacción (escala 1-5)	>4.0
Modelo estudiante	Correlación con evaluaciones externas	r>0.7
PLN	Coherencia narrativa	<5% inconsistencias
PLN	Pertinencia retroalimentación	>75% similitud con expertos
Experiencia	Tasa completitud escenarios	>80%
Técnica	Latencia respuesta IA	<5 segundos

Nota. Las métricas reflejan criterios cuantitativos utilizados para evaluar aprendizaje, coherencia narrativa, calidad del retroalimentación y desempeño técnico del sistema.

8.2 Arquitectura de Despliegue

El diseño conceptual del simulador debe materializarse eventualmente en una infraestructura computacional real. Esta sección examina consideraciones arquitectónicas fundamentales que determinaran cómo se distribuyen los componentes del sistema, cómo se comunican entre sí y cómo escalan para soportar múltiples usuarios simultáneos.

8.2.1 Arquitectura Cliente-Servidor Estratificada

La arquitectura propuesta para el simulador es un esquema cliente–servidor estratificado en múltiples capas, donde cada capa tiene responsabilidades delimitadas y puede evolucionar de forma independiente. Este enfoque resulta consistente con arquitecturas típicas de sistemas educativos inteligentes, en las cuales se separan los módulos de interacción, razonamiento, tutoría y persistencia para garantizar mantenibilidad y escalabilidad funcional

(Bourdeau et al., 2010; Park Woolf, 2009). La capa de presentación corresponde a la interfaz de usuario ejecutándose en el navegador web del estudiante, esta capa es responsable de renderizar escenarios visualmente atractivos, capturar las decisiones del usuario mediante formularios o interfaces conversacionales, mostrar visualizaciones del estado del proyecto, como gráficos de presupuesto consumido, cronogramas o indicadores de riesgo; y presentar el retroalimentación generado por el sistema de manera clara y motivadora (Myers, 1985).

La capa de aplicación constituye el backend del sistema y contiene toda la lógica de negocio del simulador, esta capa implementa el motor de escenarios que gestiona el flujo de la simulación, coordina las llamadas al modelo de lenguaje para generar respuestas de interesados, calcula las consecuencias de las decisiones del usuario, actualiza el modelo bayesiano del estudiante y determina qué escenario presentar a continuación basándose en la estrategia pedagógica diseñada en el Módulo Tutorial. La separación entre frontend y backend permite que equipos diferentes trabajen en cada capa y que se escalen independientemente según demanda (Bourdeau et al., 2010; Park Woolf, 2009).

La capa de datos persiste toda la información que debe sobrevivir entre sesiones: perfiles de usuarios, biblioteca completa de escenarios y sus configuraciones, historial detallado de todas las simulaciones ejecutadas (cada decisión tomada, cada respuesta generada, cada actualización del modelo del estudiante) y métricas agregadas que permiten análisis del uso del sistema. Esta capa típicamente se implementa mediante sistemas de gestión de bases de datos relacionales que garantizan consistencia y durabilidad de la información (Korth et al., 2019).

Finalmente, la capa de inteligencia artificial constituye un servicio especializado responsable de la inferencia del modelo de lenguaje. Dado que los LLM son computacionalmente demandantes y requieren hardware especializado, resulta crucial aislarlos en una capa independiente, se recomienda implementar esta capa mediante infraestructura propia con GPUs de alto rendimiento o, más pragmáticamente para volúmenes moderados de uso, mediante el consumo de APIs comerciales de proveedores especializados como OpenAI o Anthropic que ya resolvieron los desafíos de escalar modelos de lenguaje eficientemente (OpenAI, s.f.; Anthropic, s.f.).

La comunicación entre estas capas se realizará mediante protocolos estándar de internet, haciendo uso de peticiones HTTP a una API REST o GraphQL, el backend consulta

la base de datos mediante manejadores estándar que implementan el protocolo de comunicación del sistema de bases de datos elegido, y el backend invoca el servicio de inteligencia artificial mediante peticiones HTTP a su API. Esta arquitectura basada en protocolos estándar maximiza la flexibilidad: cualquier capa puede reemplazarse por una implementación alternativa siempre que respete la misma interfaz de comunicación (Bourdeau et al., 2010; Mitchell, 1997).

8.2.2 Escalabilidad Horizontal mediante Replicación

Un desafío crítico es cómo el sistema maneja el crecimiento en el número de usuarios simultáneos. Inicialmente, durante una fase piloto con decenas de usuarios, una arquitectura centralizada simple donde todos los componentes residen en un único servidor puede ser suficiente, sin embargo, si el simulador se despliega institucionalmente, en una universidad con miles de estudiantes; o comercialmente en una plataforma de capacitación profesional, se requiere que la arquitectura escale sin degradar el rendimiento.

La estrategia fundamental de escalabilidad es la replicación horizontal: en lugar de aumentar la capacidad de un único servidor, se despliegan múltiples instancias idénticas del backend que operan en paralelo. Un balanceador de carga distribuye las peticiones entrantes entre estas instancias de manera que ninguna se sobrecargue. Para que esta estrategia funcione, el backend debe diseñarse sin estado interno: cada petición debe contener toda la información necesaria para procesarse, sin asumir que peticiones previas del mismo usuario fueron procesadas por la misma instancia, el estado persistente (quién es el usuario, qué escenario está ejecutando, cuál es su historial) reside exclusivamente en la capa de datos, que todas las instancias consultan (Chung, 2003).

La capa de presentación escala trivialmente porque servir archivos estáticos a navegadores es una operación ligera que puede distribuirse globalmente mediante *Content Delivery Networks*. La capa de datos presenta desafíos de escalabilidad más complejos, ya que consta de múltiples instancias del backend consultando y modificando simultáneamente la base de datos, lo que puede crear conflictos de concurrencia. Las soluciones incluyen el uso de bases de datos diseñadas para alta concurrencia, la implementación de réplicas de lectura y, en casos extremos de escala masiva, técnicas de *sharding* donde diferentes subconjuntos de usuarios se almacenan en bases de datos físicamente separadas (Korth et al., 2019).

La capa de inteligencia artificial presenta los mayores desafíos de escalabilidad porque la inferencia de modelos de lenguaje de gran tamaño es intrínsecamente costosa, si se implementa con infraestructura propia, se requieren múltiples GPUs operando en paralelo para atender suficientes peticiones simultáneamente. Las colas de procesamiento son esenciales, dado que cuando llega una petición de inferencia, se encola y los procesos especializados que monitorean la cola la procesan en cuanto tienen capacidad disponible. El frontend debe diseñarse para manejar esta latencia variable mostrando indicadores de progreso apropiados al usuario (Myers, 1985). Si se utilizan APIs comerciales, el proveedor gestiona la escalabilidad internamente, pero el sistema debe monitorear el costo económico por consumo para garantizar sostenibilidad operativa (OpenAI, s.f.; Anthropic, s.f.; Amazon Web Services, s.f.; Google Cloud, s.f.; Microsoft Azure, s.f.).

8.2.3 Integración con Ecosistemas Educativos Institucionales

El despliegue del simulador en instituciones educativas requiere integración transparente con la infraestructura tecnológica existente, particularmente *con Learning Management Systems* (LMS) como Moodle, Canvas o Blackboard, que constituyen la columna vertebral de la educación digital institucional al gestionar cursos, estudiantes, contenidos y calificaciones (IMS Global Learning Consortium, 2019).

La integración se implementa mediante el estándar LTI, protocolo que permite que herramientas externas se embeben dentro del LMS sin que el usuario perciba discontinuidad en su experiencia. Cuando un estudiante accede a una actividad del simulador, el LMS transmite información contextual (identidad del estudiante, curso, permisos) mediante tokens de autenticación seguros; el simulador autentica al usuario sin credenciales adicionales (Single Sign-On) y, al finalizar, reporta calificaciones o progreso al LMS mediante el mismo protocolo estandarizado, generando una experiencia fluida dentro del entorno habitual del estudiante. Desde una perspectiva pedagógica, la integración con el LMS no es meramente técnica, sino estratégica: permite que el simulador forme parte del ecosistema formativo institucional, alineando resultados de aprendizaje, evaluación y seguimiento del progreso con los marcos educativos definidos por el curso o programa (Bourdeau et al., 2010; Park Woolf, 2009).

La privacidad de datos constituye consideración crítica regulada por legislación específica como FERPA en Estados Unidos o GDPR en Europa. Los datos educativos—identidad del estudiante, decisiones tomadas, desempeño registrado—son información sensible

que requiere protección rigurosa. La arquitectura debe garantizar que los datos se almacenan encriptados tanto en reposo como en tránsito, que solo personal autorizado accede a ellos mediante control de acceso granular, que los estudiantes ejercen derechos de revisión y exportación de sus propios datos, y que existe eliminación permanente bajo solicitud del usuario o tras períodos de retención legalmente definidos. El sistema implementa niveles diferenciados de acceso: instructores visualizan desempeño agregado de su clase sin acceder al historial detallado individual sin consentimiento explícito; administradores institucionales acceden a datos anonimizados para investigación sin información identificable no autorizada. Esta arquitectura de seguridad y privacidad protege derechos individuales mientras habilita usos legítimos educativos e investigativos de los datos generados (Romero & Ventura, 2020; UNESCO, 2021).

8.3 Especificación de Plataforma de Procesamiento

La factibilidad técnica del simulador depende fundamentalmente de la disponibilidad de hardware con capacidad suficiente para ejecutar sus componentes más demandantes dentro de latencias aceptables para interacción humana en tiempo real.

8.3.1 Disparidad entre Procesamiento Tradicional y Procesamiento de IA

Los componentes del simulador presentan perfiles computacionales radicalmente diferentes: el procesamiento tradicional del backend, como validar entradas, consultar bases de datos, ejecutar lógica de negocio y actualizar estructuras en memoria, es relativamente ligero; un servidor moderno con cuatro núcleos de CPU y cuatro gigabytes de RAM puede manejar entre cincuenta y cien usuarios simultáneos sin dificultad, completando operaciones en milisegundos.

En contraste, la inferencia del modelo de lenguaje que simula interesados es considerablemente más demandante, los LLM contienen miles de millones de parámetros organizados en docenas de capas neuronales; generar una respuesta requiere propagar la entrada a través de todas estas capas realizando multiplicaciones masivas de matrices en cada una. Las CPUs, diseñadas para instrucciones secuenciales, ejecutan esta inferencia excesivamente lento (de treinta a sesenta segundos por respuesta) (Bengio et al., 2016; Géron, 2022). Las GPUs, con miles de núcleos paralelos diseñados originalmente para gráficos 3D, reducen esta latencia a dos o cinco segundos, lo cual resulta más aceptable para interacción

humana con retroalimentación visual adecuado, mostrando un mensaje como "El cliente está pensando...". (NVIDIA, s.f.).

Esta disparidad crea una decisión arquitectónica fundamental, ya que implementar infraestructura propia requiere una inversión significativa en hardware especializado de alto rendimiento, con GPUs diseñadas específicamente para inferencia de modelos de lenguaje (como la serie NVIDIA A100 Tensor Core), que representan inversiones de capital sustanciales por unidad, con consumo energético considerable bajo carga operativa y capacidad limitada de inferencias simultáneas por dispositivo (NVIDIA, s.f.). Para soportar volúmenes elevados de usuarios concurrentes se necesitarán múltiples unidades de este hardware especializado, representando inversión inicial considerable más costos operativos continuos de electricidad, refrigeración y mantenimiento.

La alternativa pragmática para volúmenes pequeños a moderados es utilizar APIs comerciales de proveedores especializados, tales como OpenAI, Anthropic o Google, que ya resolvieron estos problemas de escala mediante infraestructura distribuida masiva. El simulador envía peticiones HTTP con el *prompt* y recibe respuestas generadas; el proveedor maneja toda la complejidad de *clústers* de GPUs, balanceo de carga y optimización de latencia. De esta manera, el costo se transfiere de inversión de capital a gasto operativo por uso, con modelos de *pricing* típicamente basados en *tokens* procesados (OpenAI, 2024; Anthropic, 2024). Para implementaciones con uso moderado diario, este enfoque resulta significativamente más manejable económicamente que infraestructura propia, particularmente durante fases iniciales de despliegue donde el volumen de usuarios es incierto.

8.3.2 Requisitos de Almacenamiento y Base de Datos

El simulador genera volúmenes considerables de datos, aunque modestos comparados con aplicaciones multimedia. El contenido estático, como la biblioteca de escenarios, las descripciones narrativas, configuraciones, plantillas de retroalimentación y recursos multimedia, ocuparían espacio de almacenamiento moderado medido en gigabytes. Los datos por usuario activo incluyen perfil, modelo del estudiante (representado en una red bayesiana) y su historial completo de simulaciones con cada escenario ejecutado, decisiones tomadas con timestamp, respuestas generadas, actualizaciones de estado y retroalimentación proporcionado. Este nivel de granularidad es esencial para análisis posterior y genera volúmenes de datos proporcionales al nivel de actividad e interacción de cada usuario.

Para implementación institucional con miles de usuarios activos, el almacenamiento total requerido es manejable con infraestructura de nube contemporánea. Con los modelos de pricing actuales de proveedores principales de almacenamiento en nube con redundancia y backups automáticos (Amazon Web Services, s.f.; Google Cloud, s.f.), el costo de almacenamiento representa una fracción menor del presupuesto operativo total. El desafío real no es el volumen de almacenamiento sino diseñar el esquema de base de datos para que las consultas frecuentes ejecuten eficientemente incluso con millones de registros acumulados a lo largo del tiempo.

Las bases de datos relacionales como PostgreSQL son apropiadas porque los datos tienen estructura clara y relaciones estrictamente definidas: un usuario tiene múltiples simulaciones, cada simulación tiene múltiples decisiones, cada decisión generó actualizaciones del modelo (PostgreSQL Global Development Group, s.f.). Las consultas se expresan naturalmente en SQL y ejecutan eficientemente con índices apropiados; la capacidad de garantizar integridad referencial previene inconsistencias que podrían causar errores difíciles de diagnosticar (Korth et al., 2019).

8.3.3 Latencia, Tasa de Procesamiento y Experiencia del Usuario

La experiencia humana de interacción es extremadamente sensible a la latencia. La investigación en interacción humano-computadora ha establecido umbrales perceptuales ampliamente documentados: respuestas menores a 100 milisegundos se perciben como instantáneas, entre 100 y 300 milisegundos son aceptables para la mayoría de interacciones, latencias de 300 milisegundos a un segundo son perceptiblemente lentas pero tolerables, respuestas entre uno y tres segundos son aceptables únicamente con retroalimentación visual clara de que el sistema está procesando, y latencias superiores a cinco segundos generan frustración significativa en usuarios (Nielsen, 1993; Card et al., 1991).

El simulador tiene operaciones con diferentes requisitos de latencia. Actualizar visualizaciones del estado del proyecto debe ser instantáneo (menor a 100 milisegundos) para mantener la percepción de respuesta inmediata. Las consultas de base de datos deben completarse en menos de 300 milisegundos, latencia típica de sistemas diseñados con índices apropiados. La generación de respuestas mediante modelo de lenguaje representa el componente de mayor latencia, tomando varios segundos debido a la complejidad inherente del procesamiento de lenguaje natural a gran escala. La estrategia de diseño debe aceptar esta latencia inevitable pero mitigar su impacto perceptual mediante indicadores animados

atractivos que mantienen al usuario informado de que el sistema está procesando activamente su solicitud, técnica que la investigación ha demostrado mejora significativamente la tolerancia del usuario a latencias elevadas (Myers, 1985).

La Tasa de Procesamiento (capacidad de procesamiento) se refiere a cuántas operaciones por segundo puede manejar el sistema. El backend tradicional procesa volúmenes elevados de peticiones por segundo sin saturación. Una base de datos relacional adecuadamente optimizada permite miles de transacciones por segundo mediante paralelización y caché efectivo. El cuello de botella principal es la inferencia del modelo de lenguaje: cada inferencia consume tiempo considerable, limitando significativamente el capacidad de procesamiento por unidad de hardware de procesamiento especializado.

8.3.4 Estimación de Capacidad para Diferentes Volúmenes

Para dimensionar apropiadamente la infraestructura, resulta necesario considerar escenarios representativos de uso que permitan proyectar demanda, latencia esperada y costo operativo. Esta estimación se basa en principios de planificación de capacidad y evaluación progresiva de carga, habituales en sistemas de simulación y plataformas digitales educativas (Chung, 2003; Bourdeau et al., 2010). Un escenario pequeño correspondiente a un curso universitario individual de 100 estudiantes con un pico de 20 simultáneos puede soportarse con una arquitectura simple de un único servidor backend de especificaciones moderadas y uso de API comercial para el modelo de lenguaje, representando inversión mensual manejable para instituciones educativas (OpenAI, s.f.; Anthropic, s.f.).

Un escenario mediano correspondiente a implementación departamental o institucional con 1,000 usuarios activos con un pico de 200 simultáneos requeriría una arquitectura distribuida en múltiples instancias de backend detrás de un balanceador de carga para distribuir peticiones, base de datos con réplicas de lectura para distribuir consultas, y cuota apropiada en servicios de API comerciales. Esta escala representa presupuestos operativos típicos de programas académicos consolidados o iniciativas corporativas de capacitación (Korth et al., 2019).

Un escenario grande correspondiente a despliegue masivo institucional o plataforma comercial con 10,000 usuarios con picos de 1,000 simultáneos requeriría infraestructura robusta, de múltiples instancias backend con arquitectura de microservicios, base de datos con sharding horizontal para distribuir carga entre múltiples servidores, y decisión estratégica crítica sobre el modelo de lenguaje (Korth et al., 2019). En este volumen, mantener el uso de

APIs comerciales representa gasto operativo considerable, mientras que la inversión en infraestructura propia con hardware especializado puede justificarse económicamente considerando el retorno sobre inversión a mediano plazo tras amortizar el capital inicial (NVIDIA, s.f.; OpenAI, s.f.; Anthropic, s.f.).

8.3.5 Gestión de Riesgos y Contingencia

La gestión de riesgos constituye un componente esencial del plan de implementación, orientado a anticipar, evaluar y mitigar los factores que puedan comprometer la ejecución técnica, económica o académica del proyecto. Su objetivo principal es asegurar la continuidad del desarrollo y la calidad del producto final, incluso ante escenarios de incertidumbre o cambio. El enfoque adoptado se basa en las recomendaciones del Project Management Institute (PMI, 2021), combinando una evaluación cualitativa de la probabilidad de ocurrencia y del impacto potencial de cada riesgo, con estrategias de mitigación proporcionales a su criticidad. De esta manera, el plan integra tres dimensiones clave: prevención (anticipación de fallas), contención (procedimientos de respuesta) y resiliencia (capacidad de recuperación).

En el plano técnico, los riesgos más relevantes se relacionan con la estabilidad del entorno de desarrollo y la integración entre los distintos módulos del simulador. El entrenamiento del agente de aprendizaje automático, por ejemplo, puede enfrentar dificultades derivadas de conjuntos de datos insuficientes, errores de configuración o sobreajuste de los modelos. Para reducir estas probabilidades, se prevé un proceso de validación cruzada, el uso de datasets incrementales y la aplicación de técnicas de *tuning* iterativo. Asimismo, el riesgo de incompatibilidades entre el motor de IA, el módulo de procesamiento del lenguaje natural y el sistema de eventos se aborda mediante un esquema de integración continua (CI/CD), con pruebas unitarias y funcionales en cada iteración. Otro aspecto crítico es la disponibilidad de la infraestructura tecnológica, ya que eventuales interrupciones en servicios en la nube o pérdida de datos podrían afectar el cronograma de trabajo. Para ello, se establecen mecanismos de respaldo automático, snapshots de servidores, almacenamiento redundante y políticas de versionado en repositorios distribuidos (Géron, 2022; Mitchell, 1997).

En cuanto a los riesgos humanos, el plan reconoce la posible rotación o indisponibilidad de miembros clave del equipo técnico. Dado que el desarrollo del simulador depende de perfiles altamente especializados en inteligencia artificial y procesamiento del lenguaje natural, la ausencia prolongada de un integrante podría impactar significativamente en la continuidad del

proyecto. Para mitigar esta situación, se prevé una documentación técnica exhaustiva, la asignación de responsabilidades cruzadas y la implementación de un protocolo de transferencia de conocimiento dentro del equipo (Beck et al., 2001; Schwaber & Sutherland, 2020).

En conjunto, este plan de gestión de riesgos busca una ejecución previsible, controlada y resiliente, capaz de adaptarse a los cambios tecnológicos o institucionales que pudieran surgir durante el proceso. La incorporación de mecanismos de monitoreo, respaldo y control de calidad contribuye a reducir la probabilidad de fallas críticas y a sostener la integridad técnica y pedagógica del simulador inteligente a lo largo de toda su implementación (Aldrich, 2009; Bourdeau et al., 2010).

Capítulo 9 – Plan de Implementación

9.1 Recursos Humanos

La implementación del simulador inteligente, tal como fue conceptualizado en los capítulos anteriores, requiere un equipo interdisciplinario con competencias en ingeniería de software, inteligencia artificial, diseño instruccional y gestión de proyectos. La estructura aquí descrita corresponde al escenario de implementación integral del sistema en su versión completa. Su desarrollo debe abordarse de manera progresiva por etapas, incorporando recursos especializados conforme aumente el nivel de sofisticación técnica del producto.

La conformación del equipo responde a un modelo multidisciplinario alineado con buenas prácticas en el desarrollo de sistemas inteligentes interactivos (Bourdeau et al., 2010; Park Woolf, 2009) y con marcos de gestión de proyectos tecnológicos de carácter iterativo (PMI, 2021; Schwaber & Sutherland, 2020). El equipo de trabajo contempla los siguientes perfiles principales:

Gerente de proyectos. Asume la dirección general del proyecto, coordinando las tareas de planificación, ejecución y supervisión técnica. Su función principal consiste en garantizar la coherencia entre los componentes del sistema, definir prioridades y asegurar la correcta integración entre los módulos de simulación, inteligencia artificial y procesamiento del lenguaje natural. Este rol mantiene una participación constante a lo largo de todo el ciclo de implementación.

Desarrolladores Backend. Son responsables del diseño y construcción de la lógica del simulador, incluyendo la creación de servicios API, el control de eventos y la gestión de datos. Sobre ellos recae la implementación de los módulos centrales de simulación, de los mecanismos de registro de acciones y del sistema de puntuación. Su participación es intensiva durante las fases de desarrollo del proyecto.

Desarrolladores Frontend. Se encargan de la interfaz de usuario y de los flujos de interacción con el simulador. Su trabajo incluye la visualización del estado del proyecto, el diseño de los tableros de control y la integración de los elementos pedagógicos interactivos. Este grupo participa principalmente en las etapas intermedias y finales del proceso de desarrollo.

Especialista en IA y PLN. Este perfil resulta esencial en etapas intermedias y avanzadas del proyecto. Es responsable de la implementación del modelo de usuario, la

integración de modelos de lenguaje y la validación de algoritmos de aprendizaje automático y aprendizaje por refuerzo (Barto y Sutton, 2018; Murphy, 2022; Pearl, 1988). En fases iniciales, su participación puede limitarse a la configuración de APIs externas de modelos preentrenados, reduciendo complejidad técnica y requerimientos de infraestructura.

Especialista en Experiencia de Usuario Educativa. Colabora en la definición de la experiencia de aprendizaje y en la elaboración de los escenarios narrativos del simulador. Además, se encarga de diseñar el flujo pedagógico, asegurando que las decisiones de diseño técnico mantengan coherencia con los objetivos formativos definidos. Este rol garantiza que las decisiones técnicas mantengan consistencia con principios de aprendizaje experiencial y tutoría inteligente (Kolb, 2015; Park Woolf, 2009). Su participación es clave en la fase de diseño conceptual y prototipado.

Especialista en Control de Calidad. Interviene en las etapas finales, realizando pruebas de funcionalidad, coherencia pedagógica y usabilidad. Es responsable de diseñar casos de prueba, verificar la estabilidad del sistema y asegurar que cada módulo cumpla con los requisitos establecidos antes de ser liberado.

Consultor de Dominio. Cumple una función de supervisión externa, validando la alineación entre los objetivos teóricos del simulador y los resultados obtenidos. Participa principalmente en las instancias de evaluación y cierre, aportando observaciones sobre la pertinencia educativa y la calidad del producto final.

La composición del equipo responde a un modelo multidisciplinario y de rotación adaptativa, en el que algunos recursos, como los desarrolladores, participan de manera continua, mientras que los especialistas (IA, Experiencia de Usuario, Calidad) se incorporan de forma focalizada en fases específicas. La estrategia de asignación de recursos responde a un esquema escalonado. En etapas iniciales puede operar un núcleo reducido compuesto por gerente de proyecto y desarrolladores, incorporando progresivamente especialistas en IA, experiencia educativa y control de calidad a medida que el sistema evoluciona hacia versiones más sofisticadas. Este enfoque modular permite controlar costos, validar hipótesis pedagógicas de manera incremental y reducir riesgos asociados a desarrollos altamente complejos en etapas tempranas (PMI, 2021).

9.2 Presupuesto

El presente presupuesto corresponde al escenario de implementación integral del simulador inteligente, conforme a la arquitectura completa descrita en los capítulos anteriores. No obstante, resulta fundamental aclarar que dicho monto representa una estimación para el desarrollo del sistema en su versión avanzada, incluyendo modelado adaptativo, aprendizaje por refuerzo y procesamiento conversacional completo. En línea con el enfoque progresivo propuesto en este plan de implementación, el desarrollo debe estructurarse en fases incrementales, comenzando con un prototipo funcional mínimo (MVP) de menor complejidad técnica y menor requerimiento presupuestario, e incorporando progresivamente los módulos de mayor sofisticación.

Las estimaciones salariales se calcularon utilizando datos del mercado tecnológico argentino correspondientes al segundo trimestre de 2025, relevados por Openqube, PageGroup, Bumeran Tech y Deel, que proveen estadísticas actualizadas por rol y seniority. La metodología de estimación incorpora márgenes de contingencia conforme a las recomendaciones del Project Management Institute (PMI, 2021).

9.2.1 Costo Total Estimado del Equipo Humano

Tomando los valores medios de cada rango salarial y la duración proyectada de participación por perfil, se realizó una estimación detallada del costo de cada rol involucrado en el proyecto. La Tabla 2 presenta la distribución completa del presupuesto de recursos humanos, especificando para cada perfil el costo mensual promedio estimado, la cantidad de meses de participación, el número de recursos asignados y el costo total resultante por rol. Esta desagregación permite visualizar la estructura de inversión y comprender el peso relativo de cada especialidad dentro del desarrollo integral del simulador.

Tabla 2

Distribución aproximada del presupuesto

Perfil	Costo Promedio	Duración	Recursos	Costo Total Estimado
Gerente de proyectos	$ 4.150.000	8	1	$ 33.200.000
Desarrolladores Backend	$ 3.400.000	8	3	$ 81.600.000
Desarrollador Frontend	$ 3.150.000	7	2	$ 44.100.000

Perfil	Costo Promedio	Duración	Recursos	Costo Total Estimado
Ingeniero en ML	$ 4.600.000	8	2	$ 73.600.000
Especialista PLN	$ 4.600.000	6	1	$ 27.600.000
Ingeniero de Datos	$ 4.200.000	6	1	$ 25.200.000
DevOps	$ 4.200.000	6	1	$ 25.200.000
Diseñador UX	$ 2.250.000	6	1	$ 13.500.000
Analista QA	$ 2.000.000	6	1	$ 12.000.000
Consultor de Dominio	$ 1.000.000	3	1	$ 3.000.000
Subtotal				$ 339.000.000
+ 19 % contingencia				$ 64.410.000
+ Cargas Sociales y Beneficios (~20%)				$ 67.800.000
Total estimado final				≈ $ 471.210.000

Nota: Los valores de *Costo promedio* y *Costo total estimado* están expresados en pesos argentinos (ARS). Los valores de *Duración* representan meses de trabajo proyectado.

El costo total proyectado de recursos humanos representa aproximadamente el 88 % del presupuesto global de implementación, proporción típica en proyectos de base tecnológica con alta participación de perfiles especializados. Los valores adoptados aseguran la viabilidad del desarrollo, contemplando remuneraciones competitivas, estabilidad del equipo y sostenibilidad durante la ejecución del plan. La aplicación del margen de contingencia y de los costos indirectos asociados (beneficios, impuestos y cargas sociales) garantiza un escenario financiero realista frente a las variaciones del contexto económico argentino.

9.2.2 Recursos Tecnológicos y Costos de Infraestructura

La infraestructura tecnológica requerida para la implementación del simulador inteligente se definió considerando un enfoque híbrido que combina recursos locales de desarrollo (hardware de los integrantes del equipo) y servicios en la nube para pruebas, almacenamiento y despliegue del entorno productivo. El dimensionamiento se realizó siguiendo los lineamientos de AWS Well-Architected Framework y las recomendaciones del

PMBOK® (PMI, 2021) para la planificación de adquisiciones tecnológicas, priorizando escalabilidad, resiliencia y control de costos.

Equipamiento de Desarrollo Local. Cada integrante del equipo requiere equipamiento que garantice la ejecución fluida de entornos de desarrollo integrados (IDE), contenedores, herramientas de virtualización y modelos de IA. La Tabla 3 detalla la configuración mínima recomendada para cada puesto de desarrollo, incluyendo equipamiento informático y licencias de software base necesarias para garantizar un entorno de trabajo adecuado para tareas de programación, entrenamiento de modelos y pruebas de integración. Se incluye el costo estimado unitario por componente y el costo promedio total por puesto, a fin de dimensionar la inversión en infraestructura local requerida para el equipo completo.

Tabla 3

Costo estimado de los recursos de Hardware y Licencias de Software

Componente	Especificación Mínima	Costo Estimado
Laptop / Workstation	CPU Intel i7 / Ryzen 7, 32 GB RAM, SSD 1 TB	$ 1.900.000
Monitor externo (27")	Resolución 2K o superior	$ 430.000
Periféricos (teclado, mouse, headset)	Ergonómicos, calidad profesional	$ 250.000
Licencias de software base (SO, Office, IDEs)	Windows Pro, Visual Studio, JetBrains	$ 400.000
Costo por puesto de desarrollo (promedio)		≈ $ 2.980.000
Total (12 puestos)		≈ $ 35.760.000

Nota: Los valores de Costo estimado están expresados en pesos argentinos (ARS). Los precios de hardware corresponden a relevamientos realizados en octubre de 2025 en proveedores nacionales (Mercado Libre, Compugamer, Frávega y Garbarino) y reflejan configuraciones de gama alta utilizadas en entornos de desarrollo profesional.

Infraestructura en la Nube. Para el entorno de despliegue, pruebas y entrenamiento de modelos de IA, se propone la utilización de servicios en la nube bajo modalidad PaaS e IaaS, priorizando plataformas con disponibilidad global y soporte para procesamiento de lenguaje natural. La Tabla 4 presenta la estimación de costos asociados a los principales

servicios en la nube necesarios durante el período de desarrollo, incluyendo almacenamiento, instancias de GPU para entrenamiento de modelos, servicios de balanceo de carga y dominio web. Los valores se expresan en dólares estadounidenses, dado que la mayoría de los proveedores de infraestructura tecnológica operan bajo esa denominación.

Tabla 4

Costos estimados de los servicios de la nube a contratar

Servicio	Proveedor Sugerido	Descripción	Costo Estimado	Costo Total
Almacenamiento de archivos y logs (S3 / Blob Storage)	AWS / Azure	500 GB mensuales promedio	$ 40	$ 320
Entrenamiento de modelos IA (GPU Instance)	AWS (g5.xlarge) / Google Cloud	GPU NVIDIA A10 / T4, 2 instancias promedio activas	$ 400	$ 6.400
API Gateway y balanceo de carga	AWS / Azure	2.000.000 peticiones mensuales	$ 35	$ 560
Dominio y certificados SSL	Namecheap / AWS	Certificado anual + dominio .com/.edu	$ 25	$ 25
Total estimado				$ 9.600

Nota: Los valores de Costo estimado y Costo total están expresados en dólares estadounidenses (USD).

En caso de optar por Google Cloud o Azure, los costos serían equivalentes, con una variación del ±10 %. Se prevé el uso de créditos educativos o planes gratuitos en etapas tempranas para reducir costos iniciales de infraestructura.

Servicios Complementarios. Además de los recursos principales, el proyecto contempla la contratación de herramientas colaborativas y de control de versiones, necesarias para la coordinación del equipo multidisciplinario y el versionado seguro del código fuente, la Tabla 5 resume los costos estimados de las herramientas colaborativas y de soporte al desarrollo necesarias para la coordinación del equipo multidisciplinario, la gestión de documentación, el diseño de interfaces y el versionado seguro del código fuente. Se detallan los valores mensuales por usuario, la duración proyectada de uso y el costo total acumulado por herramienta.

Tabla 5

Costos estimados de los servicios complementarios

Servicio	Proveedor	Uso Principal	Costo Mensual	Duración	Costo Total
GitHub Enterprise	GitHub	Repositorios privados y CI/CD	$ 20	8	$ 2.080
Confluence	Atlassian	Documentación y gestión colaborativa	$ 10	8	$ 1.040
Figma	Figma	Diseño de Interfaces	$ 15	6	$ 90
Google Workspace	Google	Correo, almacenamiento y colaboración	$ 12	8	$ 1.248
Total estimado					≈ $ 4.500

Nota: Los valores de Costo mensual y Costo total están expresados en dólares estadounidenses (USD). Los valores de la columna Duración están expresados en meses.

El margen de contingencia del 19 % se encuentra incorporado en la estimación de recursos humanos presentada en la sección anterior. En el caso de la infraestructura tecnológica, los valores fueron estimados con criterios conservadores de dimensionamiento, considerando variaciones de mercado y posibles incrementos en consumo de servicios en la nube. La Tabla 6 consolida todos los componentes presupuestarios previamente detallados, integrando los costos de recursos humanos, equipamiento de desarrollo, infraestructura en la nube y servicios complementarios. Esta visión unificada permite dimensionar el esfuerzo económico total requerido para la implementación integral del simulador en su versión avanzada.

Tabla 6

Costos totales estimados de implementación del sistema

Concepto	Costo estimado
Equipo de desarrollo (Recursos Humanos)	$ 471.210.000
Equipamiento de desarrollo (hardware y software base)	$ 35.760.000
Infraestructura en la nube (servidores, IA, almacenamiento)	$ 13.872.000
Servicios colaborativos y licencias complementarias	$ 6.502.500
Total estimado final	≈ $ 527.344.500

Nota: Los valores de la columna Costo estimado están expresados en pesos argentinos (ARS). Tipo de cambio de referencia: ARS 1.445 = USD 1, aproximado a octubre de 2025. Fuente de consulta: Dólar Histórico https://dolarhistorico.com/.

La combinación de recursos locales y servicios en la nube permite un equilibrio adecuado entre control operativo, rendimiento y escalabilidad, sin requerir inversiones iniciales en infraestructura propia de alto costo. Este enfoque híbrido facilita una implementación progresiva, especialmente en las etapas iniciales del proyecto, manteniendo flexibilidad para escalar según el crecimiento de la demanda y la madurez del sistema.

El presupuesto total estimado del proyecto asciende aproximadamente a USD 365.000, considerando un tipo de cambio de referencia de ARS 1.445 por dólar (octubre 2025). Del total, los recursos tecnológicos (hardware, servicios en la nube y herramientas colaborativas) representan aproximadamente el 11 % del presupuesto global, mientras que el 89 % corresponde a inversión en capital humano especializado. Esta distribución resulta coherente con proyectos intensivos en desarrollo de software e investigación aplicada en inteligencia artificial, donde el principal factor de costo es el conocimiento técnico requerido para el diseño, implementación y validación del sistema.

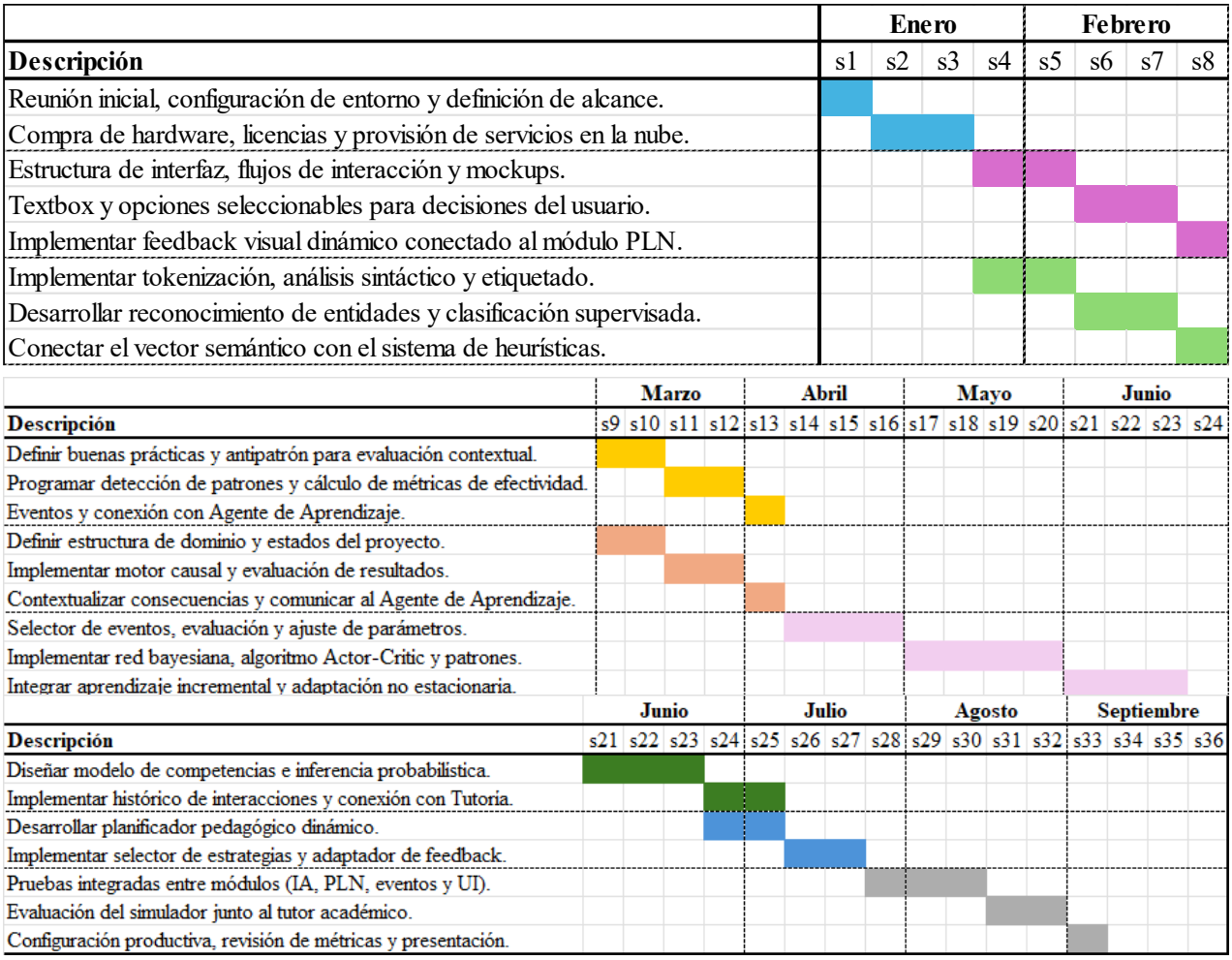
9.3 Cronograma Tentativo

La secuencia temporal prioriza la construcción de los subsistemas lógicos y cognitivos antes de la interfaz de usuario, asegurando la solidez del modelo de decisión y del motor de IA antes de su despliegue operativo. El plan contempla una duración total de aproximadamente ocho meses y medio (35 semanas), con una carga distribuida entre desarrollo backend,

inteligencia artificial y diseño instruccional. Las actividades se organizan en fases consecutivas con dependencias explícitas, siguiendo la filosofía ágil adaptativa, en la que cada etapa produce un incremento funcional verificable. La Figura 9 ilustra visualmente la secuencia de ejecución propuesta, las relaciones de dependencia y la duración estimada de cada tarea, mediante un diagrama de Gantt tentativo para el proyecto:

Figura 9

Diagrama de Gantt Tentativo



Nota. El diagrama muestra la planificación temporal preliminar del proyecto, indicando la secuencia de tareas, sus dependencias y la distribución estimada a lo largo de las semanas. Figura de elaboración propia.

9.4 Plan de Control y Seguimiento del Proyecto

El control y seguimiento del proyecto constituyen el mecanismo operativo mediante el cual se supervisa el avance del desarrollo, se validan los resultados parciales y se asegura la coherencia con los objetivos técnicos, pedagógicos y de investigación definidos. Este plan

establece los procedimientos, herramientas y métricas necesarios para garantizar que el simulador evolucione de forma ordenada, previsible y medible a lo largo de su ciclo de implementación. El Gerente de proyectos asume la responsabilidad directa del control operativo, supervisando semanalmente el progreso de los módulos en desarrollo y la correcta aplicación de los procedimientos de integración continua (CI/CD). El control de calidad técnica se realiza mediante indicadores cuantitativos y cualitativos. Entre los indicadores de desempeño se incluyen:

- Cumplimiento de hitos: porcentaje de tareas completadas dentro del plazo previsto.
- Estabilidad funcional: número de errores críticos detectados durante las pruebas unitarias o integrales.
- Cobertura de validación de IA: nivel de desempeño del agente de aprendizaje en las pruebas de entrenamiento y precisión del modelo bayesiano.
- Disponibilidad del entorno de simulación: tiempo de actividad del sistema y estabilidad de los servicios cloud.

De forma complementaria, se realiza un seguimiento pedagógico orientado a evaluar la coherencia didáctica del simulador, verificando que las decisiones del agente y los eventos generados se alineen con los objetivos de aprendizaje establecidos. Esta revisión está a cargo del Tutor Académico, quienes participan en pruebas controladas de simulación en etapas clave del desarrollo.

El flujo de control de cambios se gestiona mediante herramientas colaborativas donde se registran las tareas activas, las dependencias entre módulos y los reportes de validación. Cada cambio relevante se documenta en el repositorio del proyecto, manteniendo trazabilidad sobre versiones y revisiones de código, asegurando así la reproducibilidad del proceso de desarrollo.

Factores Externos Condicionantes

La viabilidad del simulador propuesto no depende únicamente de su diseño técnico y pedagógico, sino también de un conjunto de factores externos de naturaleza tecnológica, económica, institucional y regulatoria que pueden condicionar significativamente su desarrollo e implementación. Estos factores configuran el contexto operativo del proyecto y determinan el grado de control que el equipo de desarrollo puede ejercer sobre su ejecución.

Disponibilidad y Evolución Tecnológica

El simulador se sustenta en tecnologías de inteligencia artificial en constante evolución, particularmente modelos de lenguaje de gran tamaño (LLM) y algoritmos de aprendizaje por refuerzo. La dependencia de APIs comerciales (OpenAI, Anthropic, Google) para el procesamiento del lenguaje natural introduce dos riesgos principales: la volatilidad de los costos operativos, ya que los proveedores pueden modificar sus tarifas o políticas de uso sin previo aviso, afectando la sostenibilidad económica del proyecto; y la obsolescencia acelerada, dado que la rápida evolución de los modelos puede requerir actualizaciones frecuentes del sistema para mantener su competitividad técnica.

La disponibilidad de hardware especializado, particularmente GPUs de alto rendimiento para entrenar modelos de IA, constituye otro factor crítico. En el contexto argentino, las restricciones cambiarias y la dependencia de importaciones generan incertidumbre sobre el acceso oportuno a estos recursos, pudiendo prolongar los tiempos de adquisición o incrementar sustancialmente los costos proyectados.

Volatilidad Económica y Fluctuación Cambiaria

El presupuesto del proyecto incluye componentes dolarizados (infraestructura cloud, licencias de software, servicios de API) cuyo costo en pesos argentinos puede variar significativamente durante el período de desarrollo. La inflación proyectada del 19% aplicada como margen de contingencia podría resultar insuficiente ante escenarios de mayor volatilidad macroeconómica. Esta situación exige flexibilidad presupuestaria y mecanismos de ajuste que permitan reorientar recursos sin comprometer la calidad técnica del producto final.

Contexto Institucional y Apoyo Organizacional

La implementación del simulador en entornos educativos formales requiere respaldo institucional explícito que trascienda la aprobación administrativa inicial. La resistencia al cambio por parte de docentes acostumbrados a metodologías tradicionales, la falta de

capacitación adecuada en tecnologías educativas emergentes, o la percepción de que la IA amenaza el rol del instructor humano pueden generar obstáculos operativos significativos.

Asimismo, la integración con sistemas existentes (plataformas LMS, bases de datos institucionales, infraestructura tecnológica) depende de políticas de TI que pueden limitar la autonomía del proyecto. Restricciones de seguridad, procedimientos de aprobación prolongados o incompatibilidades técnicas con sistemas legados son factores que pueden retrasar el despliegue o reducir la funcionalidad del simulador.

Marco Regulatorio y Protección de Datos

La naturaleza del simulador implica el procesamiento intensivo de datos educativos sensibles: historial de decisiones, patrones de razonamiento, indicadores de desempeño cognitivo. En Argentina, la Ley de Protección de Datos Personales (Ley 25.326) y su Decreto Reglamentario 1558/2001, junto con normativas internacionales como el GDPR europeo (aplicable si se procesan datos de ciudadanos europeos), establecen obligaciones estrictas de consentimiento informado, minimización de datos, seguridad técnica y derechos de acceso, rectificación y eliminación.

El cumplimiento normativo exige implementar arquitecturas de privacidad desde el diseño, medidas de resguardo de la identidad del usuario, auditorías de seguridad y políticas de retención que pueden incrementar la complejidad técnica y los costos operativos. Cambios regulatorios durante el desarrollo del proyecto podrían obligar a rediseñar componentes ya implementados.

Disponibilidad de Recursos Humanos Especializados

El mercado laboral tecnológico argentino presenta escasez de perfiles altamente especializados en inteligencia artificial, procesamiento del lenguaje natural y diseño instruccional con competencias técnicas. La competencia con empresas multinacionales que ofrecen remuneraciones dolarizadas genera rotación de personal técnico clave, lo que puede afectar la continuidad del desarrollo. Las estimaciones presupuestarias asumen disponibilidad constante del equipo, pero cambios en su composición podrían introducir retrasos significativos.

Conclusiones

El presente trabajo propuso un cambio de paradigma en la enseñanza de la gestión de proyectos, al trasladar el eje del aprendizaje desde la adquisición teórica hacia la práctica situada en la toma de decisiones bajo condiciones realistas. La propuesta del simulador inteligente constituye un diseño arquitectónico y pedagógico orientado a modelar escenarios realistas donde el estudiante pueda experimentar consecuencias, incertidumbre y presión bajo condiciones controladas.

Más que reemplazar los métodos tradicionales, el simulador se plantea como un complemento formativo que amplía las estrategias didácticas disponibles. Mientras los enfoques convencionales evalúan principalmente conocimiento conceptual, la simulación permite observar y desarrollar competencias estratégicas, emocionales y de liderazgo en entornos seguros. De este modo, se construye un puente entre teoría y práctica profesional, fortaleciendo la preparación de los futuros gestores para enfrentar la complejidad inherente a los proyectos reales.

Desde una perspectiva institucional, la propuesta introduce una arquitectura de aprendizaje basada en evidencia y analítica de desempeño, susceptible de integrarse progresivamente en programas académicos; no obstante, su implementación efectiva requiere recursos técnicos, validación empírica y un desarrollo por etapas que permita evaluar su impacto real antes de escalarlo a contextos masivos. En este sentido, el trabajo no presenta un producto terminado, sino una especificación conceptual integral que define los lineamientos para una posible materialización futura.

El valor principal de la propuesta reside en la articulación entre simulación educativa y capacidades emergentes de inteligencia artificial, planteando un marco estratégico que podría evolucionar gradualmente hacia sistemas más adaptativos y personalizados. Sin embargo, el alcance tecnológico debe evaluarse en función de su contribución pedagógica concreta y de su viabilidad institucional. La sofisticación técnica no constituye un fin en sí mismo, sino un medio condicionado a la mejora medible del aprendizaje.

En consecuencia, el simulador inteligente debe concebirse como un modelo de referencia para el diseño de experiencias formativas experimentales en gestión de proyectos, cuyo desarrollo práctico dependerá de decisiones estratégicas, recursos disponibles y validaciones sucesivas. Bajo este enfoque, el trabajo aporta una base conceptual sólida que equilibra innovación tecnológica con responsabilidad académica.

Líneas futuras de investigación

El diseño del simulador propuesto establece bases sólidas para un Sistema Tutor Inteligente en gestión de proyectos, pero también revela limitaciones inherentes que abren oportunidades significativas de investigación futura. Este capítulo identifica estas limitaciones y propone líneas de investigación específicas para superarlas, organizadas desde las más urgentes hasta aquellas de naturaleza más exploratoria. A pesar de su sofisticación técnica, el sistema presenta limitaciones fundamentales que deben reconocerse.

Limitaciones del Trabajo Actual

Limitaciones Perceptuales. El sistema no puede percibir el contexto completo del usuario, como motivaciones intrínsecas, factores emocionales externos, cansancio físico o distracciones ambientales, elementos que afectan significativamente el aprendizaje pero permanecen invisibles al sistema.

Simplificación del Modelado Cognitivo. Los modelos probabilísticos construidos constituyen aproximaciones de la inmensa complejidad de la cognición humana, capturando tendencias estadísticas pero nunca la totalidad de un individuo.

Riesgo de Optimización Superficial. Existe el peligro de que el sistema optimice métricas observables (tiempo de sesión, tasa de completitud) sin garantizar aprendizaje profundo y transferible, fenómeno conocido como "*teaching to the test*".

Restricción a Uso Individual. El diseño actual asume interacción individual, excluyendo las dinámicas colaborativas fundamentales en la práctica profesional real de gestión de proyectos.

Validación Limitada: El trabajo es primariamente arquitectónico y carece de validación empírica en contextos educativos auténticos que compruebe su efectividad pedagógica real.

Simulaciones Colaborativas y Modelado de Conocimiento Distribuido

En proyectos reales, el conocimiento está distribuido entre roles: el técnico conoce detalles de implementación que el gerente desconoce, el gerente maneja restricciones presupuestarias que el técnico ignora. Una línea crítica de investigación es extender el simulador para soportar experiencias multiusuario donde varios estudiantes asuman roles diferentes (gerente, líder técnico, representante de cliente) dentro del mismo proyecto simulado.

La arquitectura requerirá un servidor autoritativo que mantenga el estado compartido del proyecto, validando acciones según roles y propagando cambios a todos los participantes mediante *WebSockets*. El aspecto más innovador sería el modelado de conocimiento distribuido: rastrear explícitamente qué información posee cada participante y simular conflictos que emergen cuando decisiones se toman con información asimétrica, requiriendo negociación y alineamiento del equipo. Este modelado de teoría de la mente (representar las creencias de cada agente sobre las creencias de los demás) es técnicamente complejo pero pedagógicamente valioso.

Modelado Emocional y Motivacional del Estudiante

El modelo bayesiano actual infiere competencias cognitivas pero ignora dimensiones afectivas cruciales: frustración, confianza, interés intrínseco, ansiedad ante evaluación. Investigaciones futuras deberían explorar la incorporación de modelos afectivos que detecten estados emocionales mediante indicadores indirectos (patrones de navegación errática, tiempos de respuesta anormales, solicitudes repetidas de ayuda) y adapten la experiencia no solo al conocimiento sino también al estado emocional.

La detección de frustración, por ejemplo, podría desencadenar intervenciones de apoyo como simplificar temporalmente los escenarios o proporcionar retroalimentación más alentador. La detección de aburrimiento (resolución excesivamente rápida, patrones de clics aleatorios) podría incrementar la complejidad o introducir desafíos sorpresivos.

Reutilización de Componentes en Otros Dominios Educativos

Varios componentes del diseño son suficientemente genéricos para aplicarse a otros contextos:

Motor de Escenarios Narrativos

El ciclo presentar situación → capturar decisión → calcular consecuencias → actualizar estado es independiente del dominio. Podría simular diagnóstico médico (situación = síntomas del paciente, decisión = pruebas ordenadas, consecuencias = evolución del paciente) o negociación comercial (situación = oferta de contraparte, decisión = contraoferta, consecuencias = cambios en posiciones y confianza).

Modelo Bayesiano del Estudiante

La estructura de inferir variables latentes (competencias) desde observables (acciones) es universal. Requiere parametrización: definir qué competencias modelar, qué acciones observar y especificar probabilidades condicionales específicas del dominio.

Generador de retroalimentación Adaptativa

La función de analizar decisiones y generar retroalimentación constructiva es independiente del dominio si los criterios de evaluación se especifican externamente. Para facilitar reutilización, los componentes deben documentarse exhaustivamente especificando interfaces claras, y los escenarios expresarse en formatos estandarizados (JSON/YAML). Adoptar **Experience API (xAPI)** para el rastreo de interacciones permite integración automática con *Learning Management Systems* institucionales.

Transparencia, Explicabilidad y Consideraciones Éticas

El uso creciente de modelos probabilísticos y PLN introduce opacidad: las decisiones del sistema (qué retroalimentación dar, qué evento desencadenar) emergen de cálculos complejos difíciles de interpretar. Investigación futura debe abordar la explicabilidad algorítmica: permitir que usuarios comprendan por qué reciben cierto retroalimentación o por qué el sistema presenta determinados eventos.

Las técnicas de inteligencia artificial Explicable (XAI) podrían generar justificaciones comprensibles: "Presento este evento de cambio de alcance porque detecté fortaleza en gestión de tiempo pero debilidad en negociación, y este escenario entrena específicamente esa competencia". La controlabilidad debe permitir que usuarios ajusten manualmente parámetros si desean experiencias específicas, manteniendo autonomía sobre su aprendizaje.

La equidad demanda validar que el sistema no perpetúe sesgos contra grupos demográficos específicos. La privacidad obliga a proteger datos que revelan no solo conocimientos sino patrones de pensamiento y vulnerabilidades emocionales.

Fundamentalmente, el sistema debe diseñarse con humildad epistémica, reconociendo que constituye apoyo al aprendizaje, no sustituto del juicio pedagógico calificado. En contextos formales debe complementar supervisión de instructores humanos que proporcionan mentoría holística considerando desarrollo profesional integral y dilemas éticos.

Acrónimos

ABP: Aprendizaje Basado en Proyectos

API: Interfaz de Programación de Aplicaciones (*Application Programming Interface)*

AR: Aprendizaje por Refuerzo (*Reinforcement Learning,* por sus siglas en inglés*, RL)*

BART: Transformadores Bidireccionales y Autorregresivos (*Bidirectional and Auto-Regressive Transformers*).

BBN: Las Redes de Creencia Bayesianas (*Bayesian Belief Networks)*

BERT: Representaciones de Codificadores Bidireccionales Basadas en Transformadores (*Bidirectional Encoder Representations from Transformers*).

CBL: Aprendizaje Basado en Retos (*Challenge-Based Learning*)

CBM: Modelado Basado en Restricciones (*Constraint-Based Modeling)*

CPM: Método de la Ruta Crítica (*Critical Path Method*)

CPT: Tabla de probabilidad condicional (*Conditional Probability Table)*

DL: Aprendizaje Profundo (*Deep Learning*)

EDM: Minería de Datos Educacionales (*Educational Data Mining)*

EDT: Estructura de Desglose del Trabajo (*Work Breakdown Structure*, por sus siglas en inglés, WBS)

EVA: Entornos Virtuales de Aprendizaje

EVM: Gestión del Valor Ganado (*Earned Value Management*)

GPT: Transformador Generativo Preentrenado (*Generative Pretrained Transformer*).

GRU: Unidad Recurrente con Compuertas (*Gated Recurrent Unit)*

IA: inteligencia artificial (Artificial Intelligence)

KR: Representación del Conocimiento (*Knowledge Representation)*

LLM: Modelo de Lenguaje de Gran Tamaño (*Large Language Model)*

LMS: Sistemas de Gestión del Aprendizaje (*Learning Management Systems)*

LSA: Análisis Semántico Latente (*Latent Semantic Analysis)*

LSTM: Memoria a Corto y Largo Plazo (*Long Short-Term Memory*)

LTI: Interoperabilidad de Herramientas de Aprendizaje (*Learning Tools Interoperability)*

MDP: Proceso de Decisión de Markov (*Markov Decision Process)*

ML: Aprendizaje Automático (*Machine Learning)*

MLM: Modelado de Lenguaje Enmascarado (*Masked Language Modeling*).

NER: Reconocimiento de Entidades Nombradas (*Named Entity Recognition*).

NLG: Generación de Lenguaje Natural (*Natural Language Generation*).

NLU: Comprensión del Lenguaje Natural (*Natural Language Understanding*).

NSP: Predicción de la Siguiente Oración (*Next Sentence Prediction*).

PLN: Procesamiento del Lenguaje Natural (*Natural Language Processing*).

PMBOK: Guía de los Fundamentos para la Dirección de Proyectos (*Project Management Body of Knowledge*).

PMI: Instituto de Gestión de Proyectos (*Project Management Institute*).

PRNG: Generador de Números Pseudoaleatorios (*Pseudo-Random Number Generator)*

QA: Aseguramiento de la Calidad (*Quality Assurance*)

RBF: Funciones de Base Radial (*Radial Base Functions*)

RNA: Redes Neuronales Artificiales

RNN: Redes Neuronales Recurrentes

SE: Sistemas Expertos

SEA: Sistema de Evaluación Automática

SFT: Ajuste de Instruccione*s (Instruction Tuning)*

SGA: Sistema de Gestión del Aprendizaje

SL: Aprendizaje Supervisado (*Supervised Learning)*

SMA: Sistemas Multi-Agente

SRL: Etiquetado de Roles Semánticos (*Semantic Role Labeling*).

STEAM: Ciencia, Tecnología, Ingeniería, Artes y Matemáticas (Science, Technology, Engineering, Arts, and Mathematics).

STI: Sistemas Tutores Inteligentes

SVD: Descomposición de Valores Singulares (*Singular Value Decomposition*)

T5: Transformador de Transferencia Texto a Texto (*Text-to-Text Transfer Transformer*).

TAE: Teoría del Aprendizaje Experiencial (*Experiential Learning Theory*).

TIC: Tecnologías de la Información y la Comunicación

TLU: Unidad de Lógica de Umbral (*Threshold Logic Unit)*

UL: Aprendizaje no Supervisado (*Unsupervised Learning*)

VAE: Autoencoders Variacionales *(Variational Autoencoder)*

WSD: Desambiguación del Sentido de las Palabras (*Word Sense Disambiguation*).

Referencias

Abanades Sánchez, M., Primo Prieto, M., Vargas Delgado, J., y Vidal García, M. (2024*). Las tendencias de las competencias digitales infalibles en el entorno profesional.* En E. Jiménez García y P. Velasco Quintana (Eds.), *Construyendo el futuro de la educación superior en la era digital* (pp. 287–296). Dykinson, S.L. https://doi.org/10.14679/3416

Adell, J., y Girbert, M. (1997). Educació a Internet: I'aula virtual [Educación en Internet: El aula virtual]. *Temps d'Educació,18,263-279.* https://dialnet.unirioja.es/servlet/articulo?codigo=2263758

Adell, J., y Sales, A. (s.f.). *El profesor online: elementos para la definición de un nuevo rol docente*. https://dialnet.unirioja.es/servlet/articulo?codigo=6100668

Aldrich, C. (2004). *Simulations and the future of learning* [Simulaciones y el future del aprendizaje]. Pfeiffer. https://archive.org/details/simulationsfutur0000aldr

Aldrich, C. (2005*). Learning by Doing : A Comprehensive Guide to Simulations, Computer Games, and Pedagogy in E-Learning and Other Educational Experiences* [Aprender haciendo: una guía integral sobre simulaciones, juegos por computadora y pedagogía en el aprendizaje en línea y otras experiencias educativas]. Pfeiffer.

Aldrich, C. (2009). *The complete guide to simulations and serious games* [La guía complete para simulaciones y juegos serios]. Pfeiffer.

Amazon Web Services. (s.f.). AWS Pricing Calculator. Recuperado el 20 de octubre de 2025 de https://calculator.aws

Anderson, J. y Corbett, A. (1994). Knowledge tracing: Modeling the acquisition of procedural knowledge [Rastreo del conocimiento: modelado de la adquisición del conocimiento procedimental]. *User Modeling and User-Adapted Interaction*, 4. 253–278. https://doi.org/10.1007/BF01099821

Anthropic. (s.f.). *Claude API pricing*. Recuperado el 20 de octubre de 2025 de https://www.anthropic.com/api

Arana, C. (2021). *Inteligencia Artificial Aplicada a la Educación: Logros, Tendencias y Perspectivas*. *INNOVA UNTREF. Revista Argentina De Ciencia Y Tecnología*, *1*(7). Recuperado a partir de https://revistas.untref.edu.ar/index.php/innova/article/view/1107

Bahreini, K., Christyowidiasmoro, C., Dascalu, M., Dias, J., Georgiadis, K., Guimarães, M., Gutu-Robu, G., Mascarenhas, S., Nyamsuren, E., Prada, R., Ruseti, S., Santos, P. A., Westera, y W., Yumak, Z. (2020). Artificial intelligence moving serious gaming: Presenting reusable game AI components [La inteligencia artificial impulsando los juegos serios: Presentación de componentes reutilizables de IA para juegos]. *Educ Inf Technol* 25, 351–380 (2020). https://doi.org/10.1007/s10639-019-09968-2

Baidoo-Anu, D., y Owusu Ansah, L. (2023). Education in the Era of Generative Artificial Intelligence (AI): Understanding the Potential Benefits of ChatGPT in Promoting Teaching and Learning [La educación en la era de la inteligencia artificial generativa (IA): Comprendiendo los beneficios potenciales de ChatGPT en la promoción de la enseñanza y el aprendizaje]. *Journal of AI,7*(1), 52-62. https://doi.org/10.61969/jai.1337500

Banda, H. (2014). *Inteligencia Artificial: Principios y Aplicaciones*. Escuela Politécnica Nacional. https://www.researchgate.net/publication/262487459_Inteligencia_Artificial_Principios_y_Aplicaciones

Baradó, S., Ibáñez, L., y Agüero, M. (2013). Sistemas Expertos: Fundamentos, Metodologías y Aplicaciones. *Ciencia y Tecnologia,13*, 439-364. https://dialnet.unirioja.es/servlet/articulo?codigo=4843871

Barbaroux, P. (2022). Developing Leadership Skills Through Simulation-Based Training: A Research Framework and Interpretive Case Study [El desarrollo de habilidades de liderazgo a través de la formación basada en la simulación: Un marco de investigación y un estudio de caso interpretativo]. *Management international / Gestiòn Internacional*, *26*(1), 192–208. https://doi.org/10.7202/1088443ar

Barbay, J. (2021). Inteligencia artificial en la educación. Bits de Ciencia, (21), 53-55. https://doi.org/10.71904/bits.vi21.2775

Barto, A. y Sutton, R. (2018). *Reinforcement Learning* [Aprendizaje por Refuerzo]. The MIT Press.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., y Thomas, D. (2001). *Manifesto for Agile Software Development*.[Manifiesto para el desarrollo de software ágil] https://agilemanifesto.org/

Bengio, Y., Collobert, R., Louradour, J., y Weston, J. (2009). Curriculum learning [Aprendizaje por Currículo]. *Journal Of the American Podiatry Association*, *60*(1), 41-48. https://doi.org/10.1145/1553374.1553380

Bengio, Y., Courville, A. y Goodfellow, I. (2016). *Deep Learning* [Aprendizaje profundo]. MIT Press. www.deeplearningbook.org

Berné, M. (27 de Marzo de 2024). *Marco de trabajo Cynefin: dominios, dinámicas y zonas liminales*. Scrum Manager: https://www.scrummanager.com/blog/2024/03/marco-cynefin-dinamicas-y-zonas-liminales/

Bialik, M., Fadel, C. y Holmes, W. (2019). *Artificial Intelligence in Education. Promise and Implications for Teaching and Learning* [Inteligencia artificial en la educación: Promesas e implicaciones para la enseñanza y el aprendizaje].

Bishop, C. (2006). Pattern Recognition and machine learning. Springer.

Bloom, B. y Krathwohl, D. (1956) *Taxonomy of Educational Objectives: The Classification of Educational Goals, by a committee of college and university examiners. Handbook I: Cognitive Domain* [Taxonomía de los Objetivos Educativos: La Clasificación de los Objetivos Educativos, por un comité de examinadores de colegios y universidades. Manual I: Dominio Cognitivo]. David McKay Company, Inc.

Bourdeau, J., Nkambou, R. y Mizoguchi, R. (2010). *Advances in Intelligent Tutoring Systems* [Avances en sistemas inteligentes de tutoría]. Springer. https://doi.org/10.1007/978-3-642-14363-2

Bumeran Tech (s.f.). Informe de Sueldos Tecnología 2025. Bumeran Argentina. Recuperado el 20 de octubre de 2025 de https://www.bumeran.com.ar/salarios/tecnologia-sistemas-y-telecomunicaciones

Buzon-Garcia, O. (2005). La incorporación de plataformas virtuales a la enseñanza: Una experiencia de formación on-line basada en competencias. *Revista Latinoamericana de Tecnología Educativa, 4*(1). https://dialnet.unirioja.es/servlet/articulo?codigo=1303698

Campbell, D. T., y Stanley, J. C. (1963). *Experimental and quasi-experimental designs for research* [Diseños experimentales y cuasiexperimentales para la investigación]. Wadsworth Publishing.

Card, S., Mackinlay, J., y Robertson, G. G. (1991). *The information visualizer: An information workspace* [El visualizador de información: un espacio de trabajo informativo]. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 181–186. https://doi.org/10.1145/108844.108874

Castillo Herrera, M. E. (2023). Impacto de la Inteligencia Artificial en el proceso de enseñanza y aprendizaje en la educación secundaria. *LATAM Revista Latinoamericana De Ciencias Sociales Y Humanidades, 4*(6), 515 – 530. https://doi.org/10.56712/latam.v4i6.1459

Cataldi, Z., y Lage, F. (2009). *Sistemas Tutores Inteligentes: Procedimientos, métodos, técnicas y herramientas para su creación*. https://dialnet.unirioja.es/servlet/articulo?codigo=2932762

Cevallos, J. (2008). Redes Neuronales Artificiales aplicadas a la mejora de la calidad. http://dx.doi.org/10.13140/RG.2.1.2091.5929

Chevalier, M., Didier, R., Gerber, G., Magnenat, S., Mondada, F., Oudeyer, P., y Riedo, F. (mayo de 2015). *IniRobot : a pedagogical kit to initiate children to concepts of robotics and computer science* [IniRobot: un kit pedagógico para iniciar a los niños en los conceptos de robótica e informática]. https://inria.hal.science/hal-01144435

Chung. C. (2003). *Simulation Modeling Handbook: A Practical Approach* [Manual de modelado de simulación: un enfoque práctico.]. CRC Press. https://doi.org/10.1201/9780203496466

Cotton, D., Cotton, P.., y Shipway, J. (2023). *Chatting and cheating: Ensuring academic integrity in the era of ChatGPT* [Chatear y hacer trampa: Garantizar la integridad académica en la era de ChatGPT]. *Innovations in Education and Teaching International, 61(*2), 228–239. https://doi.org/10.1080/14703297.2023.2190148

Crespo, M., Guisande, M., y López-Fraile, L (2024).*El pódcast como herramienta de enseñanza-aprendizaje: La escucha activa de contenido docente.* En E. Jiménez García y P. Velasco Quintana (Eds.), *Construyendo el futuro de la educación superior en la era digital* (pp. 11–22). Dykinson, S.L. https://dialnet.unirioja.es/servlet/articulo?codigo=9797120

Cunuhay Cuchipe, W., Moreira Zamora, L., Nata Castro, D. y Peñaherrera Acurio, W. (2022). Implementación De La Inteligencia Artificial Como Recurso educativo. *Recimundo*, *6*(2) 402-413. https://doi.org/10.26820/recimundo/6.(2).abr.2022.402-413

de Antueno, E. (septiembre de 2005). *Micromundos en la escuela y simulaciones en la universidad*. I Congreso en Tecnologías de la Información y Comunicación en la Enseñanza de las Ciencias, 331-342. https://sedici.uPLN.edu.ar/handle/10915/19551

Deel (s.f.). Global Hiring Report 2025: Latin America Salaries Overview. Recuperado el 20 de octubre de 2025 de https://www.deel.com/salary-insights/

Devlin, J., Chang, M., Lee, K., y Toutanova, K. (2019). *BERT: Pre-training of deep bidirectional Transformers for language understanding* [BERT: Preentrenamiento de transformadores bidireccionales profundos para la comprensión del lenguaje.]. Conferencia del Capítulo Norteamericano de la Asociación de Lingüística Computacional, 4171–4186. https://doi.org/10.18653/v1/N19-1423

Dewey, J. (1938). *Experience & Education* [Experiencia y Educación]. Macmillan Publishing Co., Inc.

Dewey, J. (1998). *Democracia y educación* (Tercera ed.). Ediciones Morata.

Feit, F., Laufer, J., Metzger, A., y Pohl, K. (2024). *A User Study on Explainable Online Reinforcement Learning for Adaptive Systems* [Un estudio de usuarios sobre aprendizaje por refuerzo en línea explicable para sistemas adaptativos]. *ACM Transactions on Autonomous and Adaptive Systems, 19*(3), 1-44. https://doi.org/10.1145/3666005

Flouret, G. (2020). Gestión cultural de proyectos en la Argentina.

Fonseca Díaz, A., y García Cuenca, L. (2024). *La integración de la inteligencia artificial en un modelo de clasificación predictivo de la siniestralidad laboral en la Comunidad Valenciana a través de las metodologías PBL-CBS.* En E. Jiménez García y P. Velasco Quintana (Eds.), *Construyendo el futuro de la educación superior en la era digital* (pp. 381–390). Dykinson, S.L. https://dialnet.unirioja.es/servlet/articulo?codigo=9797120

Fortuny, G., y Roig, E. (2022). *La dirección de proyectos: Conceptos Básicos*. Fundació Universitat Oberta de Catalunya (FUOC).

Friedman, J., Hastie, T., y Tibshirani, R. (2009). *The Elements of Statistical Learning* [Los Elementos del Aprendizaje Estadístico]. Springer. https://doi.org/10.1007/978-0-387-84858-7

Galán Hidalgo, L., González de Ramos, C., Martínez Pascual, B., y Ramírez Adrados, A. (2024). *¿Cómo impactan las experiencias de simulación compleja en el aprendizaje de los*

estudiantes del grado en fisioterapia? En E. Jiménez García y P. Velasco Quintana (Eds.), *Construyendo el futuro de la educación superior en la era digital* (pp. 181–190). Dykinson, S.L. https://dialnet.unirioja.es/servlet/articulo?codigo=9797120

Ganascia, J. (2018). Inteligencia artificial: entre el mito y la realidad. *El Correo de la UNESCO* (3), 7-9. Obtenido de https://unesdoc.unesco.org/ark:/48223/pf0000265238_spa

García-Peña, V., Mora-Marcillo, A., y Avila-Ramirez, J. (2020). La Inteligencia Artificial en la educación. Dominio de las Ciencias, 6(3), 648-666. https://dialnet.unirioja.es/servlet/articulo?codigo=8231632

García-Ull, F. J. (2024). *ChatGPT y análisis de datos en internet*. En E. Jiménez García y P. Velasco Quintana (Eds.), *Construyendo el futuro de la educación superior en la era digital* (pp. 391–406). Dykinson, S.L. https://dialnet.unirioja.es/servlet/articulo?codigo=9797120

Géron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* [Aprendizaje automático práctico con Scikit-Learn, Keras y TensorFlow] (3 ed.). O'Reilly Media.

Goldberg, Y. (2017). *Neural Network Methods for Natural Language Processing.* [Métodos de redes neuronales para el procesamiento del lenguaje natural] Morgan & Claypool. https://doi.org/10.1007/978-3-031-02165-7

Google Cloud. (s.f.). Pricing Calculator. Recuperado el 20 de octubre de 2025 de https://cloud.google.com/products/calculator

Graesser, A., Jackson, G., Louwerse, M., Lu, S., Mitchell, H., Olney, A., y Ventura, M. (2005). *AutoTutor: A tutor with dialogue in natural language. Behavior Research Method* [AutoTutor: Un tutor con diálogos en lenguaje natural. Método de Investigación del Comportamiento], *Instruments, & Computers*, *36*(2), 180–192. https://doi.org/10.3758/BF03195563

Gross, B. (1992). La Inteligencia Artificial y su aplicación en la enseñanza. Comunicación, *Lenguaje y Educación, 13*, 73-80. https://dialnet.unirioja.es/servlet/articulo?codigo=126244

Gutiérrez, A. (1997). *Educación Multimedia y Nuevas Tecnologías*. Valladolid: Ediciones de la torre.

Guzmán, A., Núñez, G., y Sheremetov, L. (s.f.). Tecnologías de Inteligencia Artificial y de agentes computacionales en la educación: el proyecto EVA.

Han, J., Kamber, M., y Pei, J. (2012). *Data mining: Concepts and techniques* [Minería de Datos: Conceptos y Técnicas] (3rd ed.). Morgan Kaufmann. https://doi.org/10.1016/C2009-0-61819-5

Haykin, S. (2009). *neural networks and learning machines* [Redes neuronales y máquinas de aprendizaje] (Tercera ed.). Hamilton, Ontario, Canadá: Pearson.

IMS Global Learning Consortium, Inc. (16 de abril de 2019). *Learning Tools Interoperability Core Specification 1.3* [Especificación central de interoperabilidad de herramientas de aprendizaje 1.3]. Recuperado el día 09 de agosto de 2025 de https://www.imsglobal.org/spec/lti/v1p3/

Institute Project Management. (2013). Guía de los fundamentos para la dirección de proyectos (Quinta ed.). Project Management Institute, Inc.

Institute Project Management. (2021). Guía de los fundamentos para la dirección de proyectos (Séptima ed.). Project Management Institute, Inc.

Iborra Gómez, S., Kucukalic-Ibrahimovic, E., Pérez Rodríguez, V., y Quirós-Fons, A. (2024). *Aprendizaje de contenidos teóricos mediante gamificación con World Challenge Game.* En E. Jiménez García y P. Velasco Quintana (Eds.), *Construyendo el futuro de la educación superior en la era digital* (pp. 345–356). Dykinson, S.L. https://dialnet.unirioja.es/servlet/articulo?codigo=9797120

Jain, A. (2010). *Data clustering: 50 years beyond K-means* [Agrupamiento de datos: 50 años después de K-means]. *Pattern Recognition Letters*, *31*(8), 651–666. https://doi.org/10.1016/j.patrec.2009.09.011

Jara, I., y Ochoa, J. (2020). Usos y Efectos de la Inteligencia Artificial en Educación. http://dx.doi.org/10.18235/0002380

Jaynes, E. (2003). *Probability theory: The logic of science* [Teoría de la Probabilidad: La Lógica de la Ciencia]. Cambridge University Press.

Jiménez García, E., y Velasco Quintana, P. (2024). Construyendo el futuro de la educación superior en la era digital. Madrid: Dykinson. https://dialnet.unirioja.es/servlet/articulo?codigo=9797120

Jurafsky, D., y Martin, J. (2025). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models* [Procesamiento del habla y del lenguaje: una introducción al procesamiento del lenguaje natural, la lingüística computacional y el reconocimiento del habla con modelos de lenguaje]. (3ª ed.) https://web.stanford.edu/~jurafsky/slp3/

Kirkpatrick, D., y Kirkpatrick, J. (2006). *Evaluating training programs: The four levels* [Evaluación de programas de capacitación: los cuatro niveles.] (3rd ed.). Berrett-Koehler.

Kolb, D. A. (2015). *Experiential learning: Experience as the source of learning and development* [Aprendizaje experiencial: la experiencia como fuente de aprendizaje y desarrollo]. Prentice Hall.

Korth, H., Silberschatz, A., y Sudarshan, S. (2019). *Database system concepts* (7th ed.). McGraw-Hill Education.

Lama Penín, M., y Sánchez Vila, E. (2007). Técnicas de la Inteligencia Artificial aplicadas a la educación. *Revista Iberoamericana de Inteligencia Artificial, 11*(33), 7-12.

Lateef, F. (2010). *Simulation based learning: Just like the real thing* [Aprendizaje basado en simulación: Como si fuera real]. DOI: 10.4103/0974-2700.70743

Lavado-Anguera, S., Martínez-Requejo, S., Terrón-López, M., y Velasco-Quintana, P. (2024).*Implementación de IA en el aprendizaje basado en proyectos en ingeniería universitaria: Una revisión sistemática*. En E. Jiménez García y P. Velasco Quintana (Eds.), *Construyendo el futuro de la educación superior en la era digital* (pp. 445–454). Dykinson, S.L. https://dialnet.unirioja.es/servlet/articulo?codigo=9797120

León, G., y Viña Brito, S. (2017). La Inteligencia Artificial en la educación superior. Oportunidades y Amenazas. *INNOVA Research Journal, 2*(8.1), 412-422. https://doi.org/10.33890/innova.v2.n8.1.2017.399

Levi Alfaroviz, A., y Villagrasa Mejía, A. M. (2024). *Equipos multidisciplinares de alumnos de ingeniería y negocio en una experiencia de aprendizaje basado en retos (CBL).* En E. Jiménez García y P. Velasco Quintana (Eds.), *Construyendo el futuro de la educación superior en la era digital* (pp. 119–136). Dykinson, S.L. https://dialnet.unirioja.es/servlet/articulo?codigo=9797120

Macías Lara, R. (2023). La Inteligencia Artificial; análisis del presente y futuro en la educación superior. *Revista Cientifica Miltidisciplinar G-ner@ndo*. https://revista.gnerando.org/revista/index.php/RCMG/article/view/98

Maksimenko N., Romanenko Y., y Solodovnikova E. (2023). *Microlearning as a new method of teaching soft skills to university students* [El microaprendizaje como nuevo método para la enseñanza de habilidades blandas a estudiantes universitarios]. *Frontiers in Education, 8*(1177516). https://doi.org/10.3389/feduc.2023.1177516

Manning, C. y Schütze, H. (1999). *Foundations of Statistical Natural Language Processing* [Fundamentos del procesamiento estadístico del lenguaje natural]. MIT Press.

Microsoft Azure. (s.f.). Pricing – Virtual Machines and Databases. Recuperado el 20 de octubre de 2025 de https://azure.microsoft.com/pricing

Mitchell, T. (1997). *Machine Learning* [Aprendizaje de máquina]. McGraw-Hill. https://www.cs.cmu.edu/~tom/mlbook.html

Morduchowicz, R. (2023). *La Inteligencia Artificial ¿Necesitamos una nueva educación?* https://unesdoc.unesco.org/ark:/48223/pf0000386262

Moreno Padilla, R. (2019). La llegada de la Inteligencia Artificial a la educación. *RITI Journal, 7*(14). https://doi.org/10.36825/RITI.07.14.022

Murphy, K. (2022). *Probabilistic Machine Learning* [Aprendizaje de Maquina Probabilístico]. MIT Press. http://probml.github.io/book1

Myers, B. A. (1985). *The importance of percent-done progress indicators for computer-human interfaces* [La importancia de los indicadores de progreso porcentual en las interfaces humano-computadora]. *ACM SIGCHI Bulletin*, *16*(4), 11–17. https://doi.org/10.1145/1165385.317459

Nielsen, J. (1993). *Usability engineering*. AP Professional.

Norvig, P., y Russell, S. (1995). Artificial Intelligence a Modern Approach. [Inteligencia artificial: Un enfoque moderno] Prentice Hall.

NVIDIA. (s.f.). *NVIDIA A100 Tensor Core GPU*. Recuperado el 20 de octubre de 2025 de https://www.nvidia.com/en-us/data-center/a100/

OpenAI. (s.f.). *API pricing*. Recuperado el 20 de octubre de 2025 de https://openai.com/api/pricing/

Openqube (s.f.). IT Salaries Survey LATAM 2025. Fundación Sadosky. Recuperado el 20 de octubre de 2025 de http://sysar.my/sueldosgpt

PageGroup (s.f.). IT Salary Guide Argentina 2025. Michael Page Argentina. Recuperado el 20 de octubre de 2025 de https://www.michaelpage.com.ar/estudios-y-tendencias/guia-salarial-2025-actualizacion-SG25

Pan, S. J., y Yang, Q. (2009). A Survey on Transfer Learning [Una revisión sobre el aprendizaje por transferencia.]. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345-1359. https://doi.org/10.1109/tkde.2009.191

Park Woolf, B. (2009). *Building Intelligent Interactive Tutors* [Construyendo tutores inteligentes interactivos]. Elsevier.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference* [Razonamiento probabilístico en sistemas inteligentes: redes de inferencia plausible]. Morgan Kaufmann. https://dl.acm.org/doi/pdf/10.5555/534975

PostgreSQL Global Development Group. (s.f.). *PostgreSQL 16 documentation*. Recuperado el 20 de octubre de 2025 de https://www.postgresql.org/docs/16/

Quinlan, J. R. (1993). *C4.5: Programs for machine learning* [C4.5: Programas para Aprendizaje de Máquina]. Morgan Kaufmann.

Romero, C., y Ventura, S. (2020). *Educational data mining and learning analytics: An updated survey* [Minería de datos educativos y analítica del aprendizaje: una revisión actualizada]. *Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery*, *10*(3). https://doi.org/10.1002/widm.1355

Rubio, M. J. (2003). Enfoques y modelos de evaluación del e-learning. *Relieve, 9*(2), 101-120. https://revistaseug.ugr.es/index.php/RELIEVE/article/view/17177

Sánchez Rodríguez, J. (2005). Plataformas tecnológicas para el entorno Educativo. *Acción Pedagógica*(14), 18-24. https://dialnet.unirioja.es/descarga/articulo/2968980.pdf

Schwaber, K., y Sutherland, J. (2020). La Guía Scrum. https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf

Smith, N. (2011). Linguistic Structure Prediction [Predicción de estructuras lingüísticas]. Springer. https://doi.org/10.1007/978-3-031-02143-5

Tunstall, L., Von Werra, L., y Wolf, T. (2022). *Natural Language Processing with Transformers* [Procesamiento de Lenguaje Natural con Transformadores]. O'Reilly Media.

UNESCO. (2019*). International conference on Artificial intelligence and Education, Planning education in the AI Era: Lead the leap: final report* [Conferencia Internacional sobre Inteligencia Artificial y Educación, Planificar la educación en la era de la IA: liderar el salto: informe final.] Beijing, 2019. https://unesdoc.unesco.org/ark:/48223/pf0000370967

UNESCO. (2021). *Report of the Social and Human Sciences Commission (SHS)* [Informe de la Comisión de Ciencias Sociales y Humanas]. UNESCO, Paris. https://unesdoc.unesco.org/ark:/48223/pf0000379920

Velásquez Arboleda, O. (2019). El nuevo rol del docente virtual para entornos virtuales de aprendizaje, "El caso CEIPA". *Lupa Empresarial, 1*(1). https://revistas.ceipa.edu.co/index.php/lupa/article/view/401

Vilches Márquez, R. (2005). Integración de las tics a la cultura docente. *Revista Enfoques Educacionales, 7*(1), 93-102. https://dialnet.unirioja.es/servlet/articulo?codigo=9116278

Vygotsky, L. (1978). *Mind in Society* [Mente en Sociedad]. Harvard University Press.