



Buenas prácticas en el cumplimiento de plazos de entrega de desarrollo de software

Profesora : Samela, Marcela Rosalba

Alumno : Lieby, Facundo Hernan

Trabajo Final de Carrera presentado para obtener el título de Lic. en Gestión de Tecnología
informática

Diciembre, 2024

Resumen

En la actualidad, el contexto de las empresas de software se caracteriza por desafíos operativos significativos, como la asignación de recursos multifuncionales debido a la limitación tanto de personal como de herramientas disponibles. Estos desafíos afectan, entre otros aspectos, el cumplimiento de plazos de entrega, la estimación precisa del tiempo para el desarrollo de productos y la planificación efectiva de interdependencias entre equipos o proyectos. Este trabajo de investigación realiza una revisión exhaustiva de los conceptos fundamentales necesarios para comprender las diversas metodologías utilizadas en la estimación del desarrollo de software, con especial énfasis en las metodologías ágiles más comunes en el mercado, dada su estrecha relación con las estimaciones y el cumplimiento de plazos. Asimismo, el objetivo del trabajo es presentar los factores que generan el incumplimiento de los plazos de entrega en entornos donde los equipos de desarrollo de software pueden utilizar metodologías ágiles o tradicionales. La investigación realizada ofrece valiosos aportes al mundo del software, ayudando a evitar los errores más comunes en la fijación de fechas de entrega y proporcionando buenas prácticas para abordar y resolver los fracasos asociados con el cumplimiento de dichos plazos.

Palabras clave: ágil, desarrollo, eficiencia, entrega, metodologías, scrum, software

Abstract

Currently, the context of software companies is characterized by significant operational challenges, such as the allocation of multifunctional resources due to the limitations in both available personnel and tools. These challenges impact, among other things, the fulfillment of delivery deadlines, the accurate estimation of development time, and the effective planning of interdependencies between teams or projects. This research presents an exhaustive review of the fundamental concepts required to understand the various methodologies used in software development estimation, with special emphasis on the most common agile methodologies in the market, due to their close relationship with estimation and meeting deadlines. Additionally, this work aims to identify the factors that lead to missed delivery deadlines in environments where software development teams may use agile or traditional methodologies. The research offers valuable contributions to the software industry by helping to avoid the most common mistakes in setting delivery dates, and it provides best practices to address and resolve failures related to meeting these deadlines.

Keywords: agile, development, efficiency, delivery, methodology, scrum, software

Dedicatoria

Dedico este trabajo de investigación a mi amada esposa Mariana, quien me ha brindado su apoyo incondicional para atravesar estos meses de arduo trabajo en la investigación y en consecuencia, un tiempo de grandes ausencias de mi parte y muchos sacrificios.

A mis padres que me han brindado las herramientas necesarias en la vida para llegar hasta esta etapa de mi carrera y me han apoyado en las maneras más cálidas para que esto sea una realidad.

Finalmente, a todos mis colegas y amigos que de alguna forma han aportado su grano de arena y su visión para que completar esta investigación fuera posible.

Índice General

Índice de Tablas.....	8
1 - Introducción.....	9
1.1 - Identificación Del Problema.....	9
1.2 - Planteamiento Del Tema a Investigar.....	9
1.3 - Objetivos.....	10
1.4 - Justificación Del Tema.....	11
2 - Estado Del Arte.....	12
2.1 - Trabajos Relacionados.....	12
2.1.1 - Otros Trabajos Relacionados.....	14
2.2 - Carencias Halladas En Los Antecedentes.....	15
2.3 - Conclusión De Los Antecedentes.....	17
3 - Marco Teórico.....	18
3.1 - Historia De La Ingeniería De Software.....	18
3.2 - El Proceso De Desarrollo De Software.....	19
3.3 - Problemas Al Desarrollar Software.....	24
3.4 - Hitos Y Fechas Límite.....	25
3.5 - Estimación De Tiempos De Desarrollo.....	26
3.5.1 - Factores Que Dificultan La Estimación.....	26
3.5.2 - Técnicas De Estimación De Esfuerzo.....	29
3.5.2.1 - Modelos Algorítmicos.....	29
3.5.2.2 - Modelos No Algorítmicos O Técnicas Basadas En La Experiencia.....	30
3.6 - Buenas Prácticas Para La Estimación.....	32
3.7 - Metodología De Desarrollo De Software Tradicional.....	33
3.8 - Metodologías Ágiles Más Utilizadas En La Actualidad.....	36
3.8.1 - Introducción Al Agilismo.....	36
3.8.2 - Scrum.....	37
3.8.2.1 - El equipo Scrum.....	38
3.8.2.2 - Los eventos de Scrum.....	41
3.8.2.3 - Los artefactos de Scrum.....	42
3.8.3 - Kanban.....	43
3.8.3.1 - Visualizar.....	44
3.8.3.2 - Limitar el trabajo en progreso.....	46
3.8.3.3 - Manejo del flujo.....	46
3.8.3.4 - Políticas explícitas.....	48
3.8.3.5 - Implementar bucles de feedback.....	48
4 - Pregunta De La Investigación.....	50
Pregunta principal :	50

5 - Hipótesis.....	51
6 - Metodología.....	52
6.1 - Participantes.....	52
6.2 - Tipo De Muestra.....	52
7 - Resultados De La Investigación.....	55
7.1 - Recolección De Datos.....	55
7.1.1 - Puesto Del Entrevistado.....	55
7.1.2 - Tipo De Empresa.....	58
7.1.3 - Sector De La Empresa.....	59
7.1.4 - Tipo De Proyecto.....	60
7.1.5 - Duración De Los Proyectos.....	62
7.1.6 - Presupuesto Del Proyecto.....	63
7.1.7 - Metodología De Trabajo Utilizada.....	64
7.1.8 - Capacitación En La Metodología Utilizada.....	65
7.1.9 - Adaptaciones A La Metodología.....	66
7.1.10 - Herramientas.....	67
7.1.11 - Tamaño Del Equipo.....	69
7.1.12 - Cantidad De Miembros Según Su Tipo.....	70
7.1.13 - Frecuencia De Las Reuniones De Seguimiento.....	72
7.1.14 - Canales De Comunicación.....	73
7.1.15 - Método De Estimación.....	74
7.1.16 - Personal A Cargo De Las Estimaciones.....	75
7.1.17 - Ejecución De La Estimación.....	76
7.1.18 - Interacción Con Otros Equipos.....	78
7.1.19 - Evaluación De Coordinación De Tareas Con Otros Equipos.....	79
7.1.20 - Principales Causas De Incumplimiento De Fechas De Entrega.....	80
7.1.21 - Frecuencia De Incumplimiento De Plazos.....	82
7.1.22 - Impacto Del Incumplimiento De Plazos.....	83
7.1.23 - Precisión Del Proceso De Estimación.....	85
7.1.24 - Frecuencia De Retroalimentación Y Mejoras.....	86
7.2 - Análisis De Los Resultados De Las Entrevistas.....	87
8 - Conclusiones.....	114
9 - Líneas Futuras De Investigación.....	119
10 - Referencias.....	121
11 - Anexos.....	125
11.1 - Anexo 1: Modelo De Entrevista.....	125
11.1.1 - Datos Demográficos y Contextuales.....	125
11.1.2 - Características Del Proyecto.....	125
11.1.3 - Metodologías Y Prácticas De Trabajo.....	126

11.1.4 - Composición Y Dinámica Del Equipo.....	127
11.1.5 - Estimaciones.....	128
11.1.6 - Interacción Con Otros Equipos.....	129
11.1.7 - Factores De Fracaso.....	129
11.1.8 - Evaluación Y Mejoras.....	130

Índice de Figuras

Figura 1 - Capas de la ingeniería de software.....	20
Figura 2 - Etapas del desarrollo de software.....	22
Figura 3 - Cono de incertidumbre.....	28
Figura 4 - Transición de etapas de metodología en cascada.....	35
Figura 5 - Interacción de los roles de Scrum.....	39
Figura 6 - Ejemplo de tablero Kanban.....	45
Figura 7 - Los cuatro arquetipos de demora y costo.....	47
Figura 8 - Puestos desempeñados por los entrevistados.....	56
Figura 9 - Resultados de la Pregunta Nro. 2 sobre el tamaño de la organización.....	58
Figura 10 - Resultados de la Pregunta Nro. 3 sobre el sector donde se desempeña la organización.....	59
Figura 11 - Resultados de la Pregunta Nro. 4 sobre la naturaleza del proyecto.....	61
Figura 12 - Resultados de la Pregunta Nro. 5 sobre la duración del proyecto.....	62
Figura 13 - Resultados de la Pregunta Nro. 6 sobre el presupuesto del proyecto.....	63
Figura 14 - Resultados de la Pregunta Nro. 7 sobre la metodología de trabajo utilizada.....	64
Figura 15 - Resultados de la Pregunta Nro. 8 sobre la capacitación en la metodología utilizada.....	65
Figura 16 - Resultados de la Pregunta Nro. 9 sobre la modificación de las metodologías ágiles.....	67
Figura 17 - Resultados de la Pregunta Nro. 10 sobre las herramientas ágiles más utilizadas.....	68
Figura 18 - Resultados de la Pregunta Nro. 11 el tamaño promedio de los equipos de trabajo.....	69
Figura 19 - Resultados de la Pregunta Nro. 12 la cantidad de miembros del equipo según su tipo.....	70
Figura 20 - Resultados de la Pregunta Nro. 13 frecuencia de las reuniones de seguimiento.....	72
Figura 21 - Resultados de la Pregunta Nro. 14 Herramientas utilizadas para la comunicación.....	73
Figura 22 - Resultados de la Pregunta Nro. 15 Método de estimación.....	74
Figura 23 - Resultados de la Pregunta Nro. 16 Encargados de las estimaciones.....	75
Figura 24 - Resultados de la Pregunta Nro. 17 Método de ejecución de las estimaciones.....	77
Figura 25 - Resultados de la Pregunta Nro. 18 Frecuencia de interacción con otros equipos.....	78
Figura 26 - Resultados de la Pregunta Nro. 19 Evaluación de los encuestas sobre la coordinación de tareas con otros equipos.....	79
Figura 27 - Resultados de la Pregunta Nro. 20 Principales causas de incumplimiento de fechas de entrega.....	81
Figura 28 - Resultados de la Pregunta Nro. 21 Frecuencia de incumplimiento de fechas de entrega.....	82
Figura 29 - Resultados de la Pregunta Nro. 22 Impacto del incumplimiento de fechas de	

entrega.....	84
Figura 30 - Resultados de la Pregunta Nro. 23 Precisión del proceso de estimación.....	85
Figura 31 - Resultados de la Pregunta Nro. 24 Frecuencia de retroalimentación y mejoras....	86
Figura 32 - Representación gráfica del análisis de incumplimiento de fechas de entrega en función de la metodología utilizada para la estimación.....	89
Figura 33 - Representación gráfica del análisis de incumplimiento de fechas de entrega en función de la metodología utilizada para la estimación, la metodología de trabajo del equipo y su tamaño.....	92
Figura 34 - Representación gráfica del análisis de incumplimiento de fechas de entrega en función de la metodología de trabajo, el tamaño del equipo y la recepción de capacitación...	98
Figura 35 - Canales de comunicación: equipos de 2 a 5 integrantes.....	107

Índice de Tablas

Tabla 1 - Recuento de puestos normalizados por categoría.....	57
Tabla 2 - Análisis de incumplimiento de fechas de entrega en función de la metodología utilizada para la estimación.....	88
Tabla 3 - Análisis de incumplimiento de fechas de entrega en función de la metodología utilizada para la estimación, la metodología de trabajo del equipo y su tamaño.....	91
Tabla 4 - Análisis de incumplimiento de fechas de entrega en función de la metodología de trabajo utilizada, el tipo de proyecto y la adaptación de la metodología.....	94
Tabla 5 - Análisis de incumplimiento de fechas de entrega en función de la metodología de trabajo utilizada, el tamaño del equipo y la recepción de capacitación formal.....	96
Tabla 6 - Análisis de incumplimiento de fechas de entrega en función de la metodología de trabajo utilizada, la capacitación formal y la precisión de las estimaciones.....	100
Tabla 7 - Análisis de incumplimiento de fechas de entrega en función de las estimaciones incorrectas.....	102
Tabla 8 - Análisis de incumplimiento de fechas de entrega en función de los cambios en los requisitos del proyecto.....	103
Tabla 9 - Análisis de incumplimiento de fechas de entrega en función de los problemas de comunicación.....	104
Tabla 10 - Análisis de incumplimiento de fechas de entrega en función de la falta de involucramiento de los stakeholders.....	105
Tabla 11 - Análisis de incumplimiento de fechas de entrega en función de la falta de recursos..	106
Tabla 12 - Frecuencia de incumplimiento de fechas de entrega en función de la frecuencia de sincronización del equipo, la metodología de trabajo y su tamaño.....	109
Tabla 13 - Frecuencia de incumplimiento de fechas de entrega en función de la frecuencia de retroalimentación.....	111

1 - Introducción

1.1 - Identificación Del Problema

Las empresas de desarrollo enfrentan varios desafíos en la gestión de proyectos debido a la naturaleza dinámica y frecuentemente caótica de sus operaciones. Estos desafíos incluyen la falta de recursos, problemas en la gestión del tiempo y la dinámica de equipos, así como dificultades para realizar estimaciones precisas y cumplir con los plazos. Las metodologías tradicionales de gestión de proyectos a menudo no son lo suficientemente flexibles para adaptarse a los rápidos cambios necesarios en estos entornos, lo que puede llevar a ineficiencias y baja productividad.

1.2 - Planteamiento Del Tema a Investigar

En el entorno altamente competitivo y dinámico en que operan las empresas de desarrollo de Software, es crucial contar con una gestión efectiva de proyectos para garantizar la supervivencia y el crecimiento de las mismas. Sin embargo, en la mayoría de los casos, muchas de estas compañías enfrentan niveles muy elevados de desorden organizacional y significativas dificultades para la gestión eficiente de proyectos. Esto se debe en muchos casos a la falta de recursos que tienen las empresas, tanto en lo que respecta a lo tecnológico como también en lo humano. Esta escasez de recursos convierte la asignación adecuada de tareas y la planificación a largo plazo en un proceso extremadamente complicado, lo que puede dar lugar a una gestión

ineficiente del tiempo y de los recursos disponibles. Además, la falta de experiencia y cohesión en los equipos puede llevar a problemas de comunicación y coordinación, reduciendo la productividad global del equipo (Sutherland, 2014).

Muchas empresas de desarrollo de software no han adoptado una metodología adecuada para su problemática, o lo han hecho de manera incorrecta. En algunos casos, también sucede que los equipos son forzados a utilizar una metodología que no es la adecuada para la naturaleza de cada proyecto.

Es importante, entonces, entender el problema que se aborda en este trabajo: identificar los factores que dificultan el cumplimiento de los plazos de entrega y, a continuación, analizar los patrones recurrentes de incumplimiento, así como aquellos que no presentan problemas. Con base en esta información, se podrán destacar los patrones positivos que contribuyen al éxito en las entregas, es decir, las buenas prácticas a seguir.

1.3 - Objetivos

Objetivo General: Determinar los factores que dificultan el cumplimiento de los plazos de entrega en empresas de desarrollo evolutivo de software en Argentina en la actualidad.

Objetivos Particulares:

- Identificar los principales factores que afectan el cumplimiento de plazos.
- Evaluar las Prácticas Actuales en la Gestión de Proyectos de Software

-
- Analizar las metodologías y prácticas actuales empleadas en la industria del software para el cumplimiento de plazos, como Scrum, Kanban y técnicas de estimación de tiempos.
 - Desarrollar un marco de Buenas Prácticas y Recomendaciones basado en los hallazgos.

1.4 - Justificación Del Tema

Investigar como un mal uso de las metodologías ágiles, o bien, la falta de su uso puede impactar en el cumplimiento de los plazos de entrega comprometidos por los equipos de desarrollo, además de aportar información cuantitativa y estadísticas, puede ayudar a comprender qué factores no deben repetirse para aumentar la tasa de éxito en el cumplimiento de los entregables. A su vez, esta investigación ayudará a identificar, a partir de las estadísticas e información recopilada, cuáles son los posibles patrones que han llevado a estos equipos a hacer una posible mala utilización de las metodologías ágiles.

2 - Estado Del Arte

A lo largo del tiempo, han habido pocos trabajos directamente relacionados al objeto de estudio de este trabajo. En general, muchos de los estudios que se pueden encontrar están relacionados de una forma más lateral, ya sea analizando las metodologías ágiles y su impacto en los equipos. En otros casos, los pocos trabajos que pueden encontrarse son muy antiguos y no aplican como tema de actualidad, de todos modos este trabajo hará referencia en un apartado especial a uno en particular que ha sido considerado importante.

2.1 - Trabajos Relacionados

Como se mencionara previamente, en los últimos años, han habido pocos trabajos relacionados al objeto de estudio de esta investigación. Uno de ellos, es el artículo de Capuñay Uceda y Antón Perez (2021) titulado “Influencia de SCRUM en los plazos de entrega y rendimiento en los proyectos de las asignaturas de Desarrollo de Software”, que sugiere que los equipos que utilizan Scrum demuestran una mejor eficiencia que aquellos que no lo hacen. Sin embargo, estos equipos aún pueden experimentar retrasos en el cumplimiento de los plazos de entrega. Esto es interesante, porque a favor de la presente investigación, deja entrever una posible línea de investigación acerca de los patrones que han generado el incumplimiento incluso en aquellos equipos que sí optaron por utilizar Scrum.

Por otro lado, el trabajo de Achilli (2022) caratulado bajo el nombre de “*Problemas frecuentes en la aplicación de Scrum en el desarrollo de software en empresas argentinas*”, en donde si bien lo abordado son los problemas frecuentes en la aplicación de Scrum en el desarrollo

de software en empresas argentinas, también se han hecho importantes relevamientos de información que estarán estrechamente relacionados al objeto de estudio de esta investigación.

Entre las importantes afirmaciones que ha podido hacer en dicha investigación, Achilli (2022) destaca como factor común en todos los componentes de la muestra que sirvió a esta investigación el bajo grado de la utilización de la metodología Scrum de forma única. Esto, según Sutherland (2014), puede llevar a resultados subóptimos y sugiere que los equipos deben ser cautelosos al combinar o adaptar prácticas. En gran parte, esto conforma una de las principales motivaciones del presente trabajo, el cual considera en mayor medida y como hipótesis inicial que el incumplimiento de las fechas de entregas puede estar relacionado a una implementación no óptima de las metodologías.

A su vez, un estudio exploratorio de las percepciones de productividad en equipos de software ágil desarrollado por Guerrero-Calvache y Hernández (2023), ofrece una visión detallada sobre las percepciones de los profesionales del software en relación con la productividad de los equipos ágiles. Los autores identifican varios factores clave, como la velocidad de trabajo, la capacidad de adaptación al cambio y la satisfacción del cliente, que son fundamentales para evaluar la eficiencia de los equipos. Si bien el estudio no se centra directamente en el cumplimiento de plazos, estos factores están intrínsecamente relacionados con la capacidad del equipo para cumplir con los tiempos de entrega pactados. La falta de una medición clara de la productividad, como se menciona en el estudio, podría ser uno de los motivos por los cuales los equipos no pueden cumplir con las fechas límite. Es por eso que la presente investigación es una línea futura de investigación del estudio de Guerrero-Calvache y Hernández por excelencia, ya que busca relevar los patrones que llevan a los equipos de desarrollo de

software al incumplimiento de las fechas límite.

2.1.1 - Otros Trabajos Relacionados

Si bien el estudio de Kieran (2009) es más antiguo, no por ello es menos significativo para la presente investigación, ya que él mismo sugiere que hay fallos significativos en la literatura sobre métodos ágiles, incluyendo falta de claridad, cohesión teórica, parsimonia, aplicabilidad limitada y una visión ingenua sobre la evolución de la agilidad fuera del ámbito de ISD (por sus siglas en inglés *Information Systems Development*, Desarrollo de Sistemas de Información). Además, se argumenta que la teoría detrás de las prácticas ágiles es débil y que la ambigüedad y la falta de consistencia en la definición del término "ágil" han diluido su significado (p. 344). Estos detalles no son menores a la hora de intentar comprender las razones que llevan a los equipos de desarrollo a fallar en la entrega de productos en las fechas comprometidas.

Otro trabajo de gran relevancia como antecedente de la presente investigación es el de Sabogal Serrano (2017), el cual examina en profundidad los distintos elementos que influyen en la precisión de las estimaciones de esfuerzo en proyectos de software, tomando como referencia el repositorio ISBSG¹. En este estudio además, el autor identifica varios factores críticos, como la complejidad del proyecto, el tamaño del equipo, el nivel de experiencia del personal y la disponibilidad de información histórica confiable, todos ellos con un impacto directo en la precisión de las estimaciones de esfuerzo y tiempo. Este trabajo se alinea directamente con el objeto de estudio de la presente investigación, ya que uno de los principales problemas que

¹ El repositorio ISBSG es una base de datos que recopila información de proyectos de software a nivel global

enfrentan los equipos de desarrollo de software es la incapacidad de cumplir los plazos debido a estimaciones erróneas.

Por último y durante el mismo año, un estudio de Maldonado Arango (2023) llamado “Factores que afectan la productividad en equipos Scrum analizados con Pensamiento sistémico”, investiga los factores que afectan la productividad en equipos Scrum, utilizando el pensamiento sistémico como enfoque. Sus hallazgos, demuestran que la cultura organizacional, la colaboración entre los miembros del equipo y el uso adecuado de herramientas de gestión son esenciales para garantizar una alta productividad. En este sentido, el estudio resalta cómo la falta de comunicación y cohesión interna puede generar retrasos significativos, ya que las tareas no se completan dentro del tiempo estimado y, a menudo, deben ser corregidas, lo que incrementa los plazos de entrega, lo que resulta en hallazgos altamente relevantes para esta investigación.

2.2 - Carencias Halladas En Los Antecedentes

En los trabajos mencionados en la sección previa sobre el tema de competencia, pueden identificarse varias carencias. En primer lugar, el trabajo de Achilli (2022) no aborda los impactos y efectos ocurridos en los equipos de trabajo afectados por la investigación, al momento de ejecutar las tareas y generar los resultados correspondientes. Esto abre una oportunidad en sus líneas futuras de investigación para explorar el impacto en los resultados, lo que guarda una estrecha relación con la presente investigación, que podría ocupar ese lugar y llevar el estudio a un nivel más avanzado. Por otro lado, el artículo de Capuñay Uceda y Antón Perez (2021) no profundiza en los patrones causales de los equipos Scrum que incumplen con las fechas de entrega, lo que constituye un punto de partida excelente para la investigación presente.

Asimismo, el estudio de Guerrero-Calvache y Hernández (2023), aunque proporciona información valiosa sobre la productividad en equipos ágiles, no aborda directamente el cumplimiento de los plazos de entrega. La investigación se enfoca en cómo los equipos pierden productividad y cómo mejorar sus procesos internos, pero no explora en detalle la relación de estos factores con los tiempos acordados con el cliente. A pesar de su relevancia, el estudio de Maldonado Arango (2017) no ofrece un análisis exhaustivo sobre cómo los factores que destaca afectan específicamente los plazos de entrega. Aunque establece una relación entre la baja productividad y los retrasos, no profundiza en cómo estos factores se manifiestan en términos de gestión del tiempo y cumplimiento de plazos. Además, el estudio se enfoca en la implementación de Scrum, lo que podría limitar su aplicabilidad en contextos donde se utilizan otras metodologías ágiles o híbridas, que también son comunes en el desarrollo de software en Argentina. Por lo tanto, aunque este estudio ofrece información valiosa, se requiere más investigación sobre el impacto directo de estos factores en los tiempos de entrega.

Otra carencia identificada es la relacionada con el estudio de Sabogal Serrano (2017), que si bien detalla los factores que afectan la estimación del esfuerzo en proyectos de software, no aborda específicamente cómo estos factores impactan el cumplimiento de los plazos de entrega en los equipos de desarrollo. El estudio se enfoca principalmente en los aspectos técnicos de la estimación, pero no considera contextos particulares, como las dinámicas de trabajo de los equipos y sus desafíos culturales y económicos, los cuales podrían influir en el rendimiento y la planificación de estos proyectos. Finalmente, según Kieran (2009), existe una oportunidad para futuras investigaciones en torno al comportamiento y el rendimiento de las metodologías ágiles.

2.3 - Conclusión De Los Antecedentes

En base al estado del arte, es decir, los antecedentes previamente detallados en este trabajo que competen a esta investigación, se puede concluir que el tema escogido es una gran elección que no sólo servirá como la continuación o extensión de los trabajos previamente nombrados, sino que podrá proveer de importantes aportes para esta temática proporcionando pautas claras para mejorar la gestión de proyectos y evitar los errores recurrentes que llevan a los equipos de desarrollo a incumplir con los plazos de entrega.

3 - Marco Teórico

3.1 - Historia De La Ingeniería De Software

Antes de comenzar a analizar los conceptos relacionados al desarrollo de software, es importante comprender que es. Según Pressman (2010) :

El software de computadora es el producto que construyen los programadores profesionales y al que después le dan mantenimiento durante un largo tiempo. Incluye programas que se ejecutan en una computadora de cualquier tamaño y arquitectura, contenido que se presenta a medida de que se ejecutan los programas de cómputo e información descriptiva tanto en una copia dura como en formatos virtuales que engloban virtualmente a cualesquiera medios electrónicos. La ingeniería de software está formada por un proceso, un conjunto de métodos (prácticas) y un arreglo de herramientas que permite a los profesionales elaborar software de cómputo de alta calidad (p. 1).

El software ha ganado una muy elevada importancia en nuestras vidas, dado que es utilizado en comercios y aplicaciones de nuestra vida cotidiana (Pressman, 2010).

El concepto de ingeniería de software surge en el año 1968 como resultado de la famosa crisis del software² donde al comenzar a aparecer distintos tipos de hardware más complejos y con capacidad de ejecutar mejores programas y aplicaciones, estos últimos comenzaron a ser más complejos y por consiguiente, su desarrollo también se volvió más complicado. Tal así que en

² “El término crisis del software se usó desde finales de 1960 hasta mediados de 1980 para describir los frecuentes problemas que aparecían durante el proceso de desarrollo de nuevo software” (Briano, 2023, p. 10).

muchos casos el desarrollo de determinados productos no se terminaba a tiempo, provocando por supuesto un incremento en el presupuesto inicial planificado para los mismos (Briano, 2023). Debido a estos problemas, según Briano (2023), se hizo evidente la necesidad de formalizar el desarrollo de software, dada la complejidad e importancia que estos desafíos adquirieron. En respuesta a esta problemática, surgió la Ingeniería de Software como una disciplina destinada a aplicar principios y técnicas de ingeniería al desarrollo de software. Esta disciplina aborda diversas etapas del proceso, incluyendo el diseño, la implementación, la prueba y el mantenimiento.

Por otro lado, hay otras definiciones muy esclarecedoras sobre la ingeniería de software que ayudan a captar mejor su esencia y ameritan ser mencionadas. Según IEEE³ (1993), es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo operación (funcionamiento) y mantenimiento del software: es decir, la aplicación de ingeniería al software. No muy distante a ella, la definición de Sommerville (2005) sugiere que “es una disciplina de la ingeniería que comprende todos los aspectos de la producción de software” (p. 5) desde las etapas iniciales de la especificación del sistema hasta el mantenimiento de éste después de ser utilizado.

3.2 - El Proceso De Desarrollo De Software

Habiendo establecido precedente en cuanto al significado de la ingeniería de software, lo siguiente es comprender el proceso implicado en la misma. Se puede pensar en la ingeniería de software como una tecnología de varias capas (Pressman, 2010). A continuación, se expone una

³ La IEEE es el Instituto de Ingenieros Eléctricos y Electrónicos (Institute of Electrical and Electronics Engineers)

figura que representa visualmente lo expuesto:

Figura 1

Capas de la ingeniería de software



Nota. Reproducido de *Ingeniería del Software: Un enfoque práctico*, por Pressman (2010), McGraw-Hill.

Según lo expuesto en la imagen, podemos entender la ingeniería de software como una tecnología compuesta por cuatro capas. Las mismas se pueden entender según Pressman (2010) de la siguiente forma. La primera de estas capas es el compromiso con la calidad, que establece que cualquier enfoque de ingeniería debe basarse en un compromiso organizacional con la calidad. Este compromiso se alimenta de filosofías como la administración total de la calidad y Six Sigma⁴, las cuales promueven una cultura de mejora continua que, en última instancia, conduce al desarrollo de enfoques más eficaces para la ingeniería de software. La segunda capa es el proceso, que define una estructura necesaria para la obtención eficaz de tecnología. Este

⁴ Six Sigma es una metodología utilizada para minimizar errores y maximizar el valor del producto (Gygi y Williams, 2012).

proceso establece las bases para la administración de proyectos de software y determina cómo se utilizarán los métodos técnicos, tales como documentos, datos y reportes. La tercera capa son los métodos, que proporcionan la experiencia técnica necesaria para desarrollar software, e incluyen un conjunto de tareas como la comunicación, el análisis de los requerimientos, la modelación del diseño, la construcción del programa, las pruebas y el apoyo. Finalmente, la cuarta capa corresponde a las herramientas, que son el apoyo automatizado o semiautomatizado tanto para el proceso como para los métodos, facilitando la ejecución eficaz del trabajo técnico.

De este modelo en capas, es importante destacar la capa del proceso, que es en la cual nos centraremos a continuación para continuar adentrándonos en el proceso de desarrollo de software. El proceso consta de cinco actividades o etapas a través de las cuales se desarrollará la elaboración del software.

Según Pressman (2010), la primera de ellas es la comunicación y la misma comprende un trabajo muy elaborado para comprender los requerimientos del producto, los cuales generalmente son dados por un cliente. Es por ello que, dentro de esta etapa será fundamental una estrecha colaboración con el cliente para comprender todas sus necesidades y transformar las mismas en requerimientos que puedan ser interpretados por un equipo de desarrollo de software.

La segunda actividad o etapa que describe Pressman (2010) es la planeación, que consiste en tomar los requerimientos formulados en la actividad anterior y, junto a transformarlos en requerimientos accionables para el equipo de desarrollo, trazar un plan o hoja de ruta para ejecutar los mismos de la forma más óptima en pos de alcanzar los resultados deseados. Es además una etapa muy importante donde se deben denotar posibles riesgos que pudieran aparecer durante el desarrollo del producto.

En tercer lugar, nuevamente según Pressman (2010) se encuentra el modelado, el cual consiste en lograr en función de los requerimientos, plasmar los mismos en un modelo presentable y comprensible. De esta forma, se logra dar una mejor visibilidad de cómo se han interpretado las necesidades para el producto a construir, y se puede asegurar que los mismos cumplan lo esperado. Pressman (2010) sostiene que “Crea un bosquejo del objeto por hacer a fin de entender el panorama general” (p. 13).

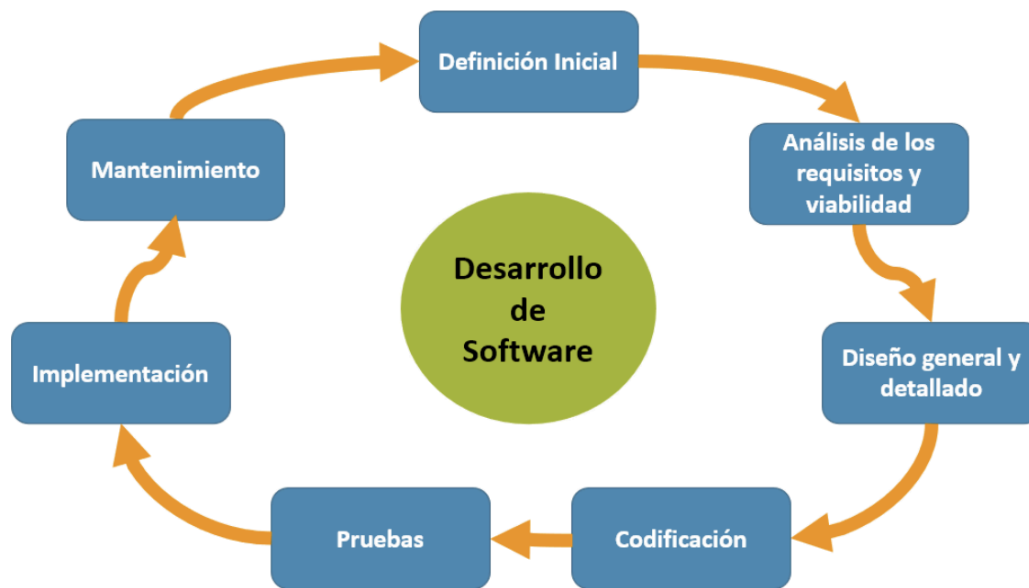
Luego del modelado, continúa la actividad de construcción. Se puede concluir gracias a Pressman (2010) que está contempla la generación del código a través del cual se plasman los requerimientos previamente mencionados y las pruebas que se requieren para encontrar errores en el mismo, y por supuesto corregirlos.

En último lugar, pero no por ello menos importante, se encuentra el Despliegue. El mismo consiste en hacer entrega del Software (en su totalidad, o bien, un incremento funcional del mismo). Además, comienza gracias a esta actividad un proceso de retroalimentación a través del cual el equipo de desarrollo puede recibir devoluciones (referidas a lo entregado) y realizar mejoras continuas sobre el producto (Pressman, 2010).

Por otro lado, resulta muy interesante realizar una comparación entre lo expuesto por Pressman (2010) y Briano (2023), donde terminaremos pudiendo denotar que, contrario a ser diferentes, sus definiciones acerca de las etapas o actividades son muy similares. A continuación, se expone una figura que nos permitirá en primer lugar, entender de forma visual lo expuesto por Briano (2023) acerca de las etapas del desarrollo de software:

Figura 2

Etapas del desarrollo de software



Nota. Reproducido de *Compilación de apuntes sobre conceptos fundamentales de la Ingeniería de Software*, por Briano (2023).

Tal como podemos ver en la figura expuesta, existe una estrecha similitud entre lo planeado por ambos autores, donde la definición inicial se empareja con la comunicación, el análisis de los requisitos y viabilidad se alinea con la planeación, y en tercer lugar el diseño general y detallado es el equivalente a el modelado. Por otro lado, en está figura tenemos el equivalente a la construcción separado en dos ítems distintos, siendo estos la codificación y pruebas. Por último, la implementación y el mantenimiento serán el equivalente a el despliegue. Resulta interesante realizar este análisis de similitudes entre estos dos autores ya que luego facilita trasladarlo a más autores y entender, que en definitiva, siempre estamos hablando de estas etapas o actividades.

3.3 - Problemas Al Desarrollar Software

Si bien los avances tecnológicos han dotado a los equipos de desarrollo de diferentes herramientas y metodologías, hay aún problemas que se acarrean desde el pasado. Entre ellos podemos encontrar desde problemas de tiempos de ejecución del desarrollo, hasta problemas de costo o de comunicación.

Entre los problemas recurrentes que se observan en el desarrollo de software, varios han sido destacados por diversos autores. Según Briano (2023), "se demora mucho tiempo en desarrollar software. Habitualmente, más de lo previsto" (p. 19), lo que genera complicaciones en la planificación y ejecución de proyectos. Además, se señala que "es difícil medir el avance del proyecto mientras se desarrolla o mantiene el software" (Briano, 2023, p. 19), y que "los costos de desarrollo son altos, frecuentemente mucho mayores a los estimados inicialmente" (p. 19). A esto se suma la extrema variabilidad en la productividad entre programadores, un factor señalado por Brooks (1975), quien también menciona la dificultad de equilibrar la eficiencia operativa y la integridad conceptual mientras se despliegan suficientes recursos humanos para gestionar la complejidad del desarrollo de sistemas a gran escala. Briano (2023) también resalta que "estimar los costos y plazos de entrega del software no es una tarea sencilla" (p. 19) y que "se gasta mucho tiempo y esfuerzo en mantenimiento" (p. 19), lo que aumenta aún más los desafíos en la gestión de proyectos. A su vez, los clientes "tienen expectativas de tiempos cada vez más cortos" (Briano, 2023, p. 19), lo que añade presión adicional sobre los equipos de desarrollo. Finalmente, DeMarco y Lister (2013) subrayan un problema relacionado con el manejo del personal, al señalar que muchas veces se trata a las personas como si fueran componentes modulares, lo cual

afecta negativamente la dinámica de los equipos y el progreso de los proyectos.

Es importante destacar, que estos problemas persisten aún en la actualidad, y terminan por causar diversos resultados no deseados como ser en el tema de interés, el incumplimiento de las fechas de entrega.

3.4 - Hitos Y Fechas Límite

Al momento de hablar de fechas límite⁵ es necesario entender cómo se establecen y qué son los hitos⁶. Los hitos son eventos concretos, específicos y medibles dentro de un proyecto que sirven como puntos de referencia claros y objetivos (Brooks, 1975). A diferencia de actividades vagamente definidas como por ejemplo "codificación" o "depuración", los hitos deben ser eventos definidos al detalle, como "código fuente completado en su totalidad e implementado" o "especificaciones firmadas por arquitectos e implementadores". Estos eventos ayudan a dividir el proyecto en fases manejables y proporcionan un camino claro hacia la finalización del proyecto. (Brooks, 1975).

Además, la ausencia de hitos, o bien, hitos mal definidos o poco claros, puede llevar al equipo de desarrollo a caer en una autodecepción y por consiguiente a una reducción de la moral (Brooks, 1975).

Podemos entonces sacar la conclusión de que debido a la existencia de los hitos, tienen lugar las fechas límite o deadlines. Generalmente los equipos de desarrollo establecen de forma conjunta con el cliente las fechas en que serán cumplidos cada uno de los hitos, por supuesto,

⁵ Comúnmente se refiere a ellas como deadlines, por su expresión en el idioma inglés.

⁶ En una gran cantidad de bibliografías es llamado milestones, por su nombre en el idioma inglés.

condicionado a la tolerancia que pueda tener la parte interesada en la entrega y la estimación que el equipo de desarrollo haya brindado para alcanzarla.

3.5 - Estimación De Tiempos De Desarrollo

La estimación de los tiempos de desarrollo de un proyecto es probablemente la tarea más difícil de llevar a cabo. Según Ruíz Constanten y Cordero Morales (2013):

Estimar consiste en determinar el valor de una variable desconocida a partir de otras conocidas, o de una pequeña cantidad de valores conocidos de esa misma variable. Es importante pensar en una predicción como en un rango más que como un simple número (p. 24).

3.5.1 - Factores Que Dificultan La Estimación

En otros ámbitos o industrias, como la construcción de una casa, estimar los tiempos no representa un desafío tan grande. Esto se debe a que es más factible contar con estadísticas que indiquen el tiempo necesario para completar ciertas tareas. Por ejemplo, siguiendo en el contexto de la construcción, se podría tener un estimado de cuánto demora por metro cuadrado pintar una pared con un determinado producto y en una determinada altura. Como se puede ver, son variables muy estáticas que hacen que hacer una proyección acerca de cuál será el tiempo para que la entrega no sea una tarea tan difícil.

En cambio, en la industria del software esto es distinto ya que hay varios factores involucrados. A continuación, se expresa cuales son estos factores según importantes autores.

Pressman (2010), sostiene que “la estimación de recursos, costo y calendario para un esfuerzo de ingeniería de software requiere experiencia, acceso a buena información histórica

(métricas) y coraje para comprometerse con las predicciones cuantitativas cuando todo lo que existe es información cualitativa” (p. 594). Además, señala que la complejidad del proyecto tiene un importante impacto sobre la planificación, pero está es relativa porque está asociada al grado de familiaridad que tenga el desarrollador con el proyecto. Volviendo al ejemplo de la construcción, es esperable que para una persona que ya tiene mucha experiencia en levantar paredes, levantar una nueva pared sea mucho más fácil que para alguien que aún no ha hecho algo similar. Por otro lado, Pressman (2010) también señala el tamaño del proyecto y la accesibilidad a información histórica (documentación) como factores importantes en la estimación.

Por otra parte, en coincidencia con lo señalado por Pressman, Briano (2023) señala de igual forma la complejidad del proyecto, el tamaño del proyecto, y la información histórica como factores que influyen en las estimaciones. Agrega además, la incertidumbre del contexto, es decir, a medida que avanza el proyecto la posibilidad de los requerimientos tengan variaciones por cuestiones internas o externas y puedan afectar lo previamente estimado.

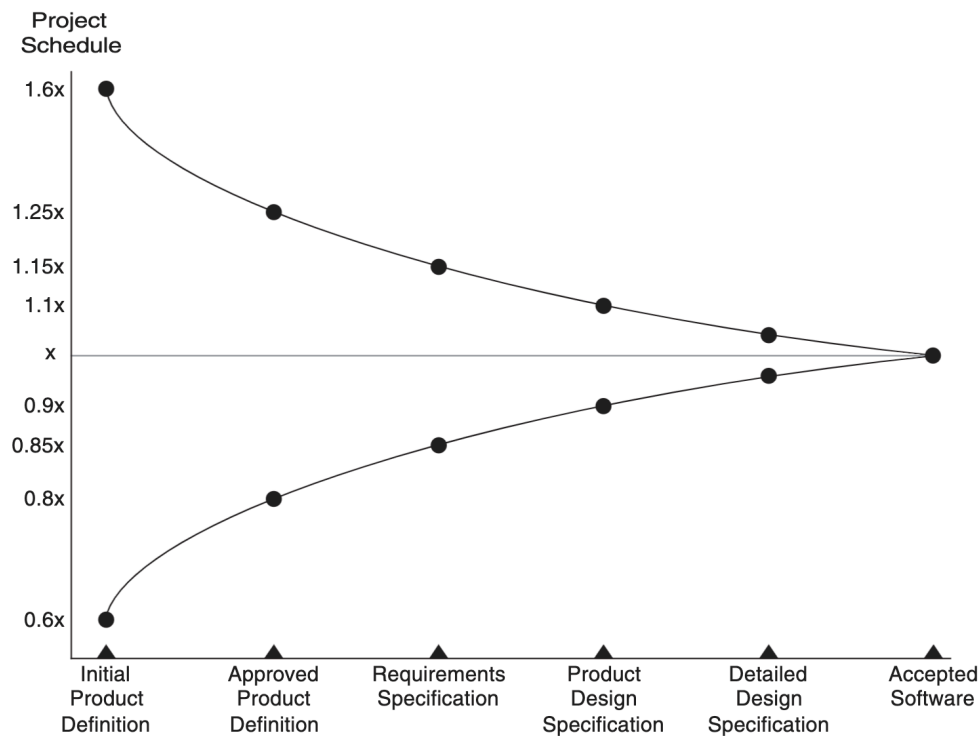
Cabe destacar que todas las mencionadas complicaciones, o bien, factores que complejizan las estimaciones para el desarrollo de software, van reduciendo a medida que avanza un proyecto y se adquiere más experiencia sobre el mismo o se reduce la incertidumbre. Este interesante concepto fue popularizado por Barry Boehm y Steve McConnell con el nombre "cono de incertidumbre⁷". Sin embargo, al principio del proyecto, las estimaciones pueden variar enormemente (de 60% a 160% en la fase de factibilidad). Esto significa que las estimaciones iniciales son altamente inciertas y pueden diferir significativamente del tiempo real que tomará

⁷ Originalmente llamado “The cone of uncertainty” en inglés.

completar un proyecto (Cohn, 2006). A continuación, se presenta a través de una figura el mencionado “cono de incertidumbre” para facilitar su comprensión de forma visual:

Figura 3

Cono de incertidumbre



Nota. Reproducido de *Agile estimating and planning*, por Cohn (2006).

Gracias a la figura exhibida previamente, es sencillo de ver cómo a medida que “Project Schedule⁸” va avanzando, la curva (es decir, la incertidumbre) comienza siendo bastante grande en “Initial Product Definition⁹”, baja considerablemente luego en “Approved Product Definition¹⁰

⁸ En español, “cronograma del proyecto”.

⁹ Traducido al español, “definición inicial del producto”.

¹⁰ En español, “definición de producto aprobada”.

” y “Requirements Specification¹¹”, y continúa por casi aplacarse al pasar por “Product Design Specification¹²”, “Detailed Design Specification¹³” y “Accepted Software¹⁴”.

3.5.2 - Técnicas De Estimación De Esfuerzo

A partir de los años 60 y hasta el presente han surgido varios modelos o técnicas de estimación de esfuerzos que pueden clasificarse en dos amplias categorías: modelos algorítmicos y modelos no algorítmicos (Mendes, 2007). A su vez, Briano (2023) hace referencia a estas dos grandes categorías pero se refiere a ellas como técnicas basadas en la experiencia y modelos algorítmicos de costos.

3.5.2.1 - Modelos Algorítmicos

En el caso de los modelos algorítmicos, se utiliza un enfoque donde se realizan fórmulas para hacer el cálculo de recursos basándose en estimaciones de determinados atributos del producto (como ser, el tamaño), así como en características del proceso, como por ejemplo la experiencia del personal (Briano, 2023). En muchos casos las empresas pueden desarrollar sus propios modelos, pero además, según Ramirez y Orantes (2011) existen algunos modelos estándares como COCOMO¹⁵ entre otros que ya pre definen las formas de proceder.

3.5.2.2 - Modelos No Algorítmicos O Técnicas Basadas En La Experiencia

En el caso de los modelos no algorítmicos, la estimación tiene sus bases en la experiencia

¹¹ En español, “especificación de requerimientos”.

¹² En español, “especificación de diseño del producto”.

¹³ En español, “especificación de diseño detallada”.

¹⁴ En español, “software aceptado”.

¹⁵ Por sus siglas en inglés Constructive Cost Model (Modelo de costo constructivo), abordado por Sommerville (2005).

del administrador con proyectos anteriores que pudieran ser similares, siendo el mismo quien emite su juicio acerca de cuáles serán los recursos necesarios. Para hacer posible y precisa esta técnica, es mandatorio que toda la información cuantitativa posible haya sido recopilada de proyectos anteriores de modo de compararlo con el proyecto en cuestión (Briano, 2023).

En comparación con los modelos algorítmicos, la desventaja de estos modelos es la subjetividad y la dependencia con las personas expertas (Salvetto De León, 2006). Estos se dividen en tres tipos: juicio de expertos, estimación por analogías y redes neuronales artificiales.

Dentro de los modelos clasificados como juicio de expertos, se procederá a mencionar algunas de las técnicas más utilizadas en la actualidad. Una de ellas es Delphi, un método que recopila una larga cantidad de opiniones de forma anónima, y busca unificar las opiniones para obtener un consenso. La gran desventaja de este método es el largo tiempo que toma para ser ejecutado (Sutherland, 2014). Por otra parte, cabe mencionar el método Planning Poker, que es una variación del método Delphi, eliminando algunos de los problemas que ese método poseía como el largo tiempo que tomaba ejecutarlo (Sutherland, 2014). Este método, es simple y según Sutherland (2014) consiste en la idea de que cada persona experta participando en la estimación, tiene un grupo de cartas¹⁶ que representan la serie de números Fibonacci¹⁷. Según el autor, cada ítem que necesita ser estimado es llevado a la mesa, donde luego de ver el ítem presentado cada miembro del equipo deberá poner una carta que elegirá en representación del peso o esfuerzo de esa tarea. Al estar todas las cartas presentadas, el equipo deberá tomar una decisión, como ser

¹⁶ Las cuales no son necesariamente físicas, en la actualidad se utilizan aplicaciones.

¹⁷ La serie numérica de Fibonacci es una sucesión en la que cada número es la suma de los dos anteriores, comenzando por 0 y 1, y se encuentra en múltiples contextos de la naturaleza y las matemáticas (Fibonacci, 2002).

promediar todas las cartas presentadas (si entre ellas no se llevan más de dos valores de cartas de diferencia), o bien, conversar al respecto si hubieran estimaciones muy diferentes y llegar a un acuerdo. Luego de estimado el ítem, se pasa a los restantes y se repite el procedimiento. Otro método muy utilizado es el Price to win, que según Kerzner (2017), se basa en determinar un precio competitivo para ganar un contrato o proyecto, adaptando el alcance, los recursos y las expectativas del proyecto a ese precio. Considera, según él, factores como el mercado, los competidores y las expectativas del cliente en vez de priorizar los costos reales. Es considerada una estrategia de alto riesgo, ya que puede comprometer la calidad del producto desarrollado. No es recomendable para productos de elevada sostenibilidad y viabilidad.

Por otro lado, dos modelos muy utilizados son el Bottom up y el Top down. El Bottom up según Kerzner (2017), implica descomponer el proyecto en componentes del tamaño más pequeño posible y detallados, identificando todas las tareas que han de ser llevadas a cabo para completarlo. El autor indica también que cada tarea se estima considerando el tiempo, costo y los recursos a utilizar, lo que resulta en una combinación muy precisa para obtener la estimación. Al igual que el método Delphi, tiene la gran desventaja de consumir mucho tiempo (Kerzner, 2017). A su vez, el Top down según Kerzner (2017) es una alternativa que empieza con una visión global del proyecto, con datos históricos y comparaciones con proyectos similares. Señala también que desde ese punto de partida, se desglosan los tiempos y costos para cada fase a ejecutar, y se realizan ajustes a medida que se vaya obteniendo más información. Esta técnica resulta útil en las primeras etapas de la planificación. Aunque es menos precisa que la técnica bottom-up, es eficiente en términos de tiempo y ayuda a definir rápidamente los límites del proyecto. Sin embargo, puede conducir a estimaciones inexactas si no se ajusta adecuadamente a

medida que el proyecto avanza y se obtienen más detalles (Kerzner, 2017).

3.6 - Buenas Prácticas Para La Estimación

En su artículo, Jørgensen (2004) sugiere una lista de doce buenas prácticas para la estimación a través de juicio de expertos que han sido validadas empíricamente, estas son:

1. Evaluar la precisión de la estimación, pero evitar la presión sobre la evaluación, es decir, la presión sobre quién estará encargado de ejecutar la estimación. Esto es, por que se quiere obtener una estimación sincera y no influenciada.
2. Evitar conflictos en las metas de la estimación. Esta práctica refiere al hecho de que muchas veces pueden haber conflictos de intereses donde, por ejemplo, el stakeholder¹⁸ es el interesado en obtener el producto lo antes posible, pero el estimador podría al dar una estimación sincera indicar tiempos que colisionan con los intereses del previamente mencionado stakeholder.
3. Solicitar a quienes realicen la estimación que justifiquen la misma y que además sean críticos acerca de ella. Este proceso genera una retroalimentación continua que mejora la precisión de las estimaciones a largo plazo.
4. Evitar información irrelevante o no confiable para la estimación.
5. Usar datos documentados de desarrollos de tareas anteriores. Esto ayuda a evitar el error humano al no necesitar acceder a toda la información desde su memoria y, en cambio, utilizar la documentación.

¹⁸ Un *stakeholder* es cualquier individuo, grupo u organización que puede afectar o ser afectado por el resultado de un proyecto o decisión. Incluye a personas o entidades que tienen un interés o "stake" en el proyecto, ya sea directa o indirectamente (Freeman, 1984).

-
6. Hallar expertos para la estimación con buen dominio del problema y con buenos antecedentes de estimación.
 7. En caso de estimar utilizando top-down y bottom-up, hacerlo de forma independientemente una de la otra, es decir, no combinarlas.
 8. Utilizar listas de estimaciones. Esto ayuda a tener una mejor organización del trabajo, no subestimar ninguna tarea y por sobre todo, no pasar nada por alto.
 9. Combinar estimaciones de diferentes expertos y diferentes estrategias.
 10. Evaluar la incertidumbre de la estimación, es decir, cuando para una determinada tarea haya algún grado determinado de incertidumbre, este debe ser incluido en forma de más tiempo o esfuerzo dentro de la estimación, pues podrían ser problemas a atravesar al desarrollar la tarea que requerirán de un tiempo extra que no es fácil visualizar al momento de la estimación.
 11. Es fundamental proporcionar retroalimentación sobre la precisión de las estimaciones, así como evaluar cómo se desarrollan las relaciones entre las tareas para ajustarlas según sea necesario.
 12. Proveer oportunidades de capacitación en estimaciones.

3.7 - Metodología De Desarrollo De Software Tradicional

Es importante entender que cuando hablamos de metodologías de desarrollo de software, estamos hablando de modelos de proceso descriptivo. Estos modelos han sido diseñados originalmente para ordenar el caos en el desarrollo del software a través de una estructura útil para el trabajo de ingeniería de software y constituyen un mapa útil para los equipos de desarrollo

(Pressman, 2010).

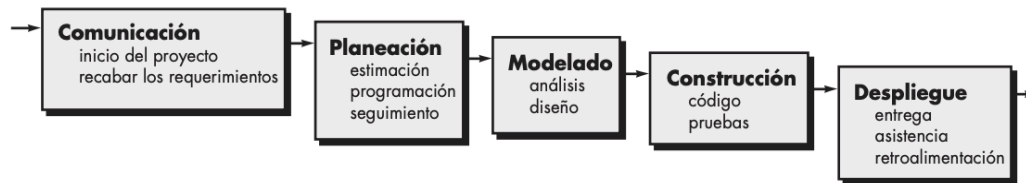
Cuando nos referimos a la metodología de desarrollo de software tradicional, generalmente lo hacemos con la definición “desarrollo o modelo en cascada”. Es muy común hoy en día, escuchar que este modelo es muy criticado. Es entonces muy importante denotar que ningún modelo es malo sino que su utilidad va a estar condicionada al tipo de proyecto (Briano, 2023). Actualmente, los proyectos de software son extremadamente dinámicos y están impulsados por el cambio. Esto hace que, en muchos casos, el enfoque tradicional no sea el más adecuado.

“Hay veces en las que los requerimientos para cierto problema se comprenden bien: cuando el trabajo desde la comunicación hasta el despliegue fluye en forma razonablemente lineal” (Pressman, 2010, p. 33). Esta afirmación de Pressman, explica con claridad cuando es adecuado el uso de una metodología en cascada.

El modelo en cascada, tiene un enfoque sistemático para el desarrollo de software, el cual comienza con la especificación de los requerimientos por parte del cliente, y luego avanza por las etapas de planeación, modelado, construcción y despliegue, como previamente se ha mencionado sobre las etapas del desarrollo del software (Pressman, 2010). Cabe destacar, que en este modelo, no es posible pasar a la siguiente etapa si la cual la precede no está concluida. Podemos tomar como ejemplo la construcción de una casa, donde en una etapa inicial se trabajará en los cimientos y durante ella, no es posible por ejemplo avanzar en la etapa de colocación del techo, puesto que son tareas dependientes. Este es un buen ejemplo de cómo además las metodologías tradicionales no existen sólo en el desarrollo del software. A continuación, se expone una figura para visualizar de forma gráfica las etapas de la metodología en cascada y su ordinalidad.

Figura 4

Transición de etapas de metodología en cascada



Nota. Reproducido de *Ingeniería del Software: Un enfoque práctico*, por Pressman (2010), McGraw-Hill.

El modelo en cascada es uno de los más antiguos entre los modelos de desarrollo de software. No obstante, según Pressman (2010), las críticas que ha recibido en las últimas décadas han llevado incluso a sus protectores más firmes a cuestionar su eficacia. Entre los principales problemas que presenta este enfoque, Pressman (2010) señala que es poco común que los proyectos reales sigan el flujo secuencial que propone el modelo. Aunque el modelo lineal permite repeticiones, las acepta de forma indirecta, lo que genera confusión cuando el equipo avanza con el proyecto y se presentan cambios. Además, Pressman (2010) destaca que, en muchos casos, es difícil para los clientes formular de manera explícita todos los requerimientos desde el inicio del proyecto, lo cual resulta problemático, ya que el modelo en cascada requiere dicha claridad desde el principio y tiene dificultades para manejar la incertidumbre inherente a los primeros pasos de un proyecto. Finalmente, otro desafío mencionado por Pressman (2010) es que los clientes deben ser pacientes, ya que no tendrán una versión funcional del software hasta que el proyecto esté en una etapa avanzada, lo que puede resultar en consecuencias graves si se

detectan errores importantes en esa fase final del proceso..

3.8 - Metodologías Ágiles Más Utilizadas En La Actualidad

3.8.1 - Introducción Al Agilismo

El enfoque Ágil en la gestión de proyectos surge como respuesta a las limitaciones de las metodologías tradicionales. El Manifiesto Ágil, elaborado en 2001 por un grupo de expertos en desarrollo de software, establece los valores y principios fundamentales del enfoque Agile (Beck et al., 2001).

Los valores constituidos en este documento, son expresados a través de los denominados “The 12 Principles of Agile” (Los doce principios de Agile) que promueven la priorización de la satisfacción del cliente mediante entregas tempranas y continuas de software funcional, la adaptabilidad frente a cambios en los requisitos, la colaboración activa entre los miembros del equipo y los stakeholders, y la capacidad de respuesta rápida al cambio (Beck et al., 2001).

Según Highsmith (2010), el enfoque Ágil se diferencia de las metodologías tradicionales, como la gestión de proyectos en cascada, por su énfasis en la flexibilidad, adaptabilidad y efectividad. Mientras que las metodologías tradicionales buscan definir de manera exhaustiva los requisitos y planificar detalladamente el proyecto desde el principio, el enfoque Agile reconoce la naturaleza dinámica y cambiante de los proyectos, abogando por ciclos de desarrollo iterativos e incrementales que permitan ajustes continuos en función de los feedbacks recibidos (Highsmith, 2010).

El surgimiento del enfoque Ágil ha revolucionado la industria del desarrollo de software y se ha extendido a otros ámbitos más allá de la tecnología, como la gestión de proyectos en

general, el marketing y la gestión de productos. Su popularidad se debe a su capacidad para fomentar la innovación, aumentar la satisfacción del cliente y mejorar la eficiencia del equipo (Highsmith, 2010).

3.8.2 - Scrum

Según Schwaber y Sutherland (2017), creadores del reconocido documento “Scrum Guide” (Guía de Scrum), Scrum es un marco de trabajo ágil utilizado para gestionar proyectos complejos y adaptativos que se basa en los principios establecidos en el Manifiesto Ágil y se centra en la entrega continua de valor al cliente a través de iteraciones cortas y regulares, conocidas como Sprints.

En pocas palabras, Scrum consiste en tener un product owner (dueño de producto) que ordene el trabajo en forma de tareas en el product backlog (cartera de productos), un equipo que transforme esas tareas en incrementos del producto y junto a los stakeholders (las partes interesadas) hagan una revisión y lleven a cabo los ajustes necesarios en ciclos posteriores (Schwaber y Sutherland, 2017).

El uso de Scrum tiene como objetivo promover varios valores clave. Entre ellos se encuentran el compromiso del equipo con el proyecto, la concentración en los objetivos y un alto nivel de franqueza, respeto y coraje (Schwaber y Sutherland, 2017).

3.8.2.1 - El equipo Scrum

Según Schwaber y Sutherland (2017), creadores de la guía de Scrum, un equipo trabajando bajo este marco de trabajo debe contar con un Product Owner, un Scrum Master y Desarrolladores. Estos equipos deben sumar en total diez integrantes o menos, ya que se ha

observado que este tamaño mejora la comunicación y la productividad (Schwaber y Sutherland, 2017).

En una interesante ampliación acerca de los roles del equipo Scrum según Briano (2023), podemos encontrar que él autor clasifica los mismos en dos categorías: roles comprometidos directamente con el proyecto y otros roles involucrados en el proyecto. Se puede concluir con su bibliografía, que los roles mencionados en la Guía de Scrum entran en la primera categoría mencionada.

A continuación, se detallan los distintos roles de Scrum comprometidos directamente con el proyecto para su mejor comprensión.

Desarrolladores

Son los integrantes del equipo Scrum que deben llevar a cabo el incremento utilizable del producto (Schwaber y Sutherland, 2017). Para ello, deben crear un plan para el Sprint (que será llamado el Sprint Backlog), y llevar el mismo a cabo inculcando calidad hasta que el mismo alcance la definición de hecho¹⁹ adaptando sus planes todos los días en función del objetivo del Sprint (Schwaber y Sutherland, 2017).

Product Owner

Según Briano (2023) “es la persona responsable de lograr el máximo valor empresarial para el proyecto. También es responsable de la articulación de requisitos del cliente. El Product

¹⁹ La definición de hecho, en inglés “Definition of done”, es según Schwaber y Sutherland (2017) una descripción formal del estado del incremento cuando alcanza los niveles de calidad requeridos para el producto.

Owner representa la voz del cliente dentro del proyecto” (p. 61). Dentro de sus responsabilidades encontramos además la efectiva comunicación del objetivo del producto al equipo de scrum, ordenar los ítems del backlog según su prioridad y asegurar que los mismos son transparentes y comprensibles (Schwaber y Sutherland, 2017).

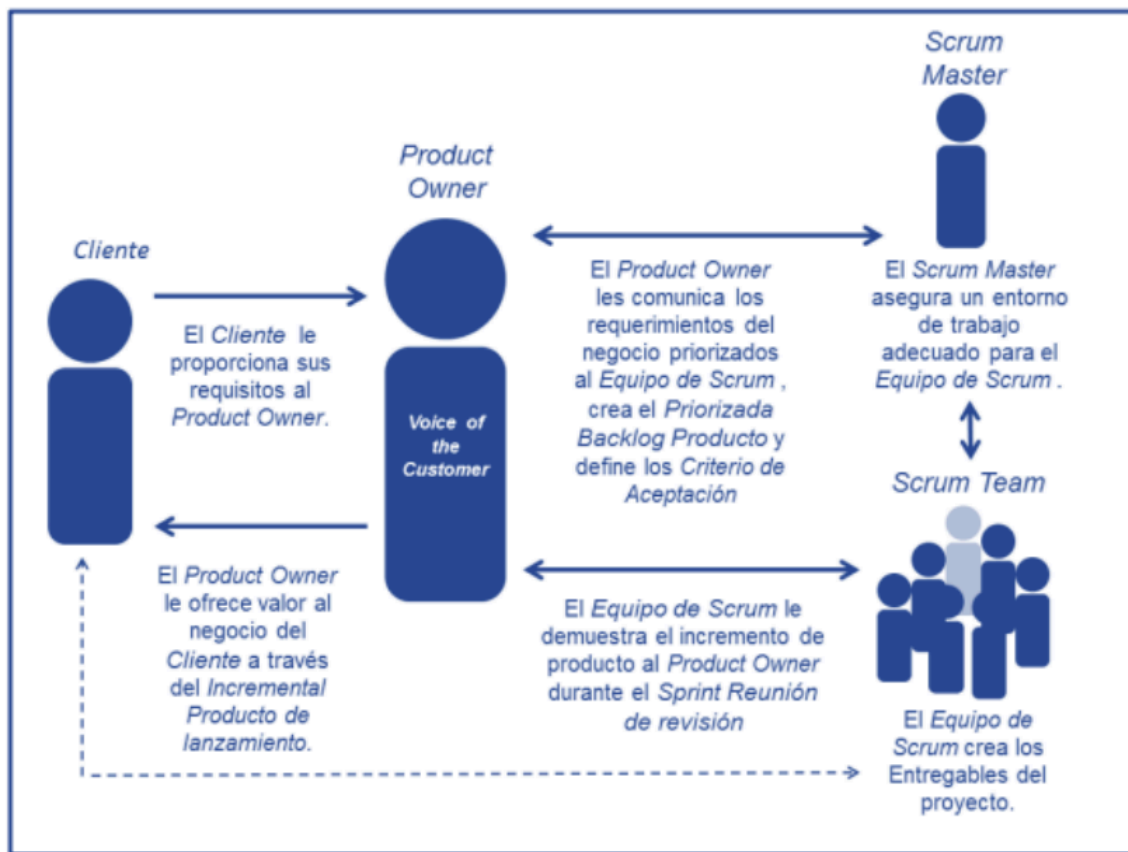
Scrum Master

Es el responsable de que Scrum sea ejecutado tal cual lo establece la guía de scrum (Schwaber y Sutherland, 2017). A su vez, tiene la responsabilidad de ayudar al equipo a mantenerse concentrado en generar los incrementos, remover los impedimentos que el equipo pueda tener para hacer progreso y asegurar que los eventos de Scrum sucedan de forma efectiva (Schwaber y Sutherland, 2017).

A continuación, se expone una imagen para facilitar la comprensión de la interacción de estos roles dentro del equipo Scrum:

Figura 5

Interacción de los roles de Scrum



Nota. Reproducido de *Compilación de apuntes sobre conceptos fundamentales de la Ingeniería de Software*, por Briano (2023).

Por otro lado, según Briano (2023) los otros roles involucrados con el proyecto se definen como : “Aquellos roles que, si bien son importantes, no participan directamente del proceso Scrum. No siempre están presentes y no son obligatorios, aunque en muchos proyectos desempeñan un papel muy importante y deben ser tenidos en especialmente en cuenta”. (p. 62). Podemos concluir como ejemplo que los roles a los que hace referencia Briano con su definición son los stakeholders, clientes, usuarios, etcétera.

3.8.2.2 - Los eventos de Scrum

Los eventos de Scrum según Schwaber y Sutherland (2017), se encuentran contenidos dentro del Sprint, el cual representa los latidos de Scrum, donde las ideas son llevadas a cabo. A continuación, se detallan los eventos de Scrum.

Sprints

Los sprints deben tener una duración fija de un mes o menos para ser consistentes, y cada Sprint comienza con la finalización del anterior (Schwaber y Sutherland, 2017). Podemos concluir gracias a la bibliografía previamente referenciada que el Sprint es un evento contenedor de eventos.

Sprint Planning

Según Briano (2023) “es donde el Product Owner presenta la versión actualizada y priorizada del Backlog, para que el equipo pueda estimar que tareas podrán ser incluidas en el próximo sprint. Se define entonces que se va a hacer y cómo se realizará” (p. 62).

Daily Scrum

Es un evento diario de 15 minutos para los desarrolladores del equipo scrum que tiene lugar todos los días a la misma hora y lugar para reducir complejidad, y su objetivo es monitorear el progreso sobre el incremento en función del objetivo del sprint, para poder ejecutar medidas correctivas de ser necesario (Schwaber y Sutherland, 2017).

Sprint Review

Su propósito es analizar el resultado del Sprint que finaliza y poder determinar futuras mejoras o ajustes (Schwaber y Sutherland, 2017). Según Briano (2023) “No se trata de una demo, sino que se presenta el software funcionando en producción. De esta reunión pueden surgir nuevas necesidades que pasan a actualizar el Product Backlog” (p. 63).

Sprint Retrospective

Este evento según Schwaber y Sutherland (2017) es un evento que busca aumentar la calidad y la efectividad del equipo. El equipo Scrum inspecciona cómo funcionó el Sprint anterior en función de los individuos, las interacciones y procesos, las herramientas y la definición de hecho, e intenta identificar accionables directos para ofrecer mejoras a las cuestiones que pudieran ser relevadas en dicha inspección (Schwaber y Sutherland, 2017).

3.8.2.3 - Los artefactos de Scrum

Los artefactos de Scrum representan trabajo o valor y están diseñados para maximizar la transparencia de la información más importante (Schwaber y Sutherland, 2017). Estos están compuestos por el Product Backlog, el Sprint Backlog y el Incremento.

El Product Backlog es una lista ordenada por prioridad de las tareas que son necesarias para mejorar el producto y es la única fuente de trabajo para el equipo Scrum (Schwaber y Sutherland, 2017). Esto lleva consigo la tarea de refinar el backlog durante todo el Sprint, lo cual consiste en descomponer los ítems en tareas más pequeñas y dar definiciones más profundas y precisas para los mismos (Schwaber y Sutherland, 2017). A su vez, el Sprint Backlog está compuesto por el objetivo del sprint (en inglés, “Spring Goal”), el conjunto de tareas del backlog tomadas para el Sprint, y un plan accionable de cómo llevarlas a cabo (Schwaber y Sutherland,

2017). El incremento por otro lado, es según Schwaber y Sutherland (2017) un avance tangible hacia el objetivo del producto que acumula los anteriores incrementos y se verifica minuciosamente para garantizar que todos funcionen en conjunto. Además Schwaber y Sutherland (2017) sostienen que durante un Sprint pueden crearse varios incrementos, y la suma de estos se presenta en la Revisión del Sprint, apoyando así el empirismo. No obstante, un incremento puede entregarse a los interesados antes de que finalice el Sprint, ya que la Revisión del Sprint no debe considerarse como una barrera para liberar valor pero el trabajo solo se considera parte de un incremento si cumple con la Definición de Terminado (Schwaber y Sutherland, 2017).

3.8.3 - Kanban

Kan-ban es una palabra en japonés que literalmente significa “tarjeta de señal” en español (Anderson, 2010). En un ambiente de fabricación, está tarjeta es utilizada como una señal para indicar a un paso anterior de un proceso que debe producir más, no pudiendo los trabajadores de cada paso del proceso realizar trabajos a menos que se les indique con una kanban desde un paso posterior (Anderson, 2010). Este sistema viene de una familia de enfoques conocidos como pull systems (sistemas de tracción) y es el mecanismo que sustenta el sistema de producción de Toyota y su enfoque Kaizen²⁰ para la mejora continua (Anderson, 2010).

Según Anderson (2010), en el sistema Kanban se coloca en circulación una cantidad de tarjetas equivalente a la capacidad acordada de un sistema, donde cada tarjeta se adjunta a una pieza de trabajo y actúa como mecanismo de señalización, pudiendo comenzar un nuevo trabajo

²⁰ Kaizen es una cultura laboral en la que toda la fuerza de trabajo se centra en mejorar continuamente la calidad, la productividad y la satisfacción del cliente (Anderson, 2010).

sólo cuando hay una tarjeta disponible. Además, sostiene que esta tarjeta libre asociada a una pieza de trabajo sigue el progreso del mismo a medida que fluye a través del sistema. Cuando no hay más tarjetas libres, no es posible comenzar ningún trabajo adicional, en consecuencia, cualquier nuevo trabajo debe esperar en una cola hasta que haya una tarjeta disponible (Anderson, 2010).

Las prácticas generales de Kanban definen un total de seis actividades esenciales para quienes administran este tipo de sistemas, según Anderson y Carmichael (2016) estas son :

- Visualizar
- Limitar el trabajo en progreso (comúnmente llamado por su acrónimo en inglés WIP, que representa “Work in progress”). Esto según Anderson (2010) no permite sobrecargar la capacidad de la fuerza de trabajo, que está determinada por la cantidad de tarjetas en circulación que se ha establecido adecuadamente.
- Manejar el flujo
- Hacer las políticas explícitas
- Implementar bucles de feedback
- Mejorar colaborativamente, evolucionar experimentalmente.

A continuación, se aborda una descripción más amplia de las primeras cinco ya que sus nombres por si solos, no permiten llegar a comprender por completo su valor.

3.8.3.1 - Visualizar

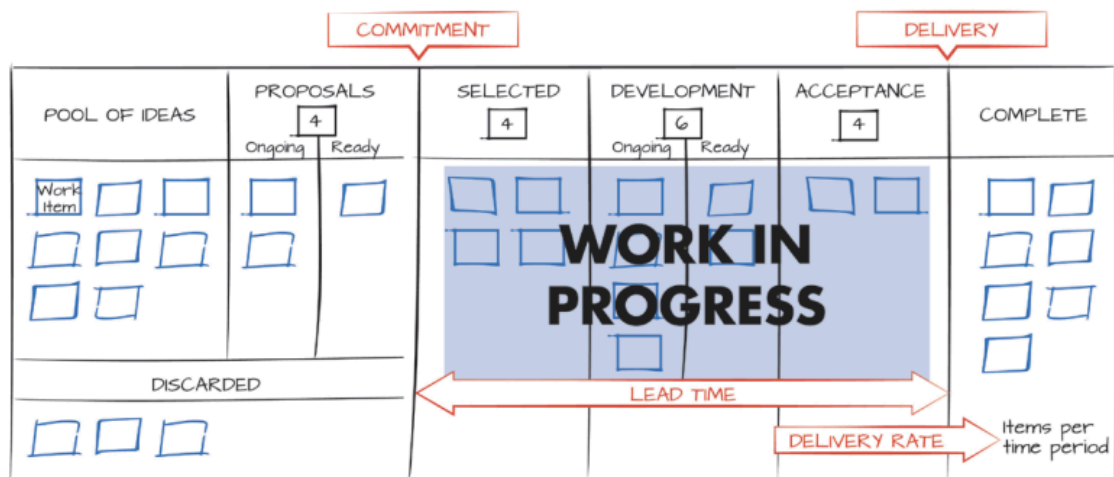
Un tablero Kanban es una de las maneras de visualizar el trabajo y el proceso a través del

cual este fluye, pero para que este tablero no pase a ser un simple sistema de flujos es importante que el trabajo y las entregas comprometidas están definidas, así como también los límites del WIP deben ser mostrados para reflejar el límite de cada estado (Anderson y Carmichael, 2016). A su vez, es sumamente importante visualizar las normas o políticas establecidas, como por ejemplo agregando pequeños resúmenes entre columnas de lo que debe ser hecho antes de que un ítem se mueva a la siguiente (Anderson y Carmichael, 2016).

Según Anderson y Carmichael (2016), los diseños de los tableros pueden variar mucho a través de las distintas implementaciones de Kanban dependiendo de cómo se usan. Además, indican que el método no tiene restricciones en como diseñar el tablero. A continuación, se presenta una figura con un ejemplo de un tablero Kanban donde puede observarse las distintas columnas que reflejan los estados del flujo y el trabajo en progreso de las mismas:

Figura 6

Ejemplo de tablero Kanban



Nota. Reproducido de *Essential Kanban Condensed*, por Anderson y Carmichael (2016).

3.8.3.2 - Limitar el trabajo en progreso

Acorde con lo afirmado por Anderson (2010), no es aconsejable que la gente trabaje en un entorno donde se realiza un número improductivo de tareas en paralelo. Esto viene emparejado con los riesgos que genera el hecho de trabajar en múltiples tareas al mismo tiempo. Según Rock (2009) una persona puede concentrarse en sólo una tarea a la vez porque cambiar entre tareas utiliza mucha energía, en consecuencia uno puede cometer muchos más errores, o bien, bajar mucho su rendimiento o precisión. Además, sostiene que la única forma de hacer dos tareas que requieren esfuerzo mental de manera rápida y precisa, es hacerlas de a una a la vez.

Es por lo anteriormente expuesto que resulta importante delimitar los límites del WIP (trabajo en progreso) de manera tal que los nuevos ítems no son comenzados hasta que el trabajo ha sido completado, evitando así tener mucho trabajo parcialmente completado, lo cual resulta caro, enlentece el proceso y no es más que un desperdicio de recursos (Anderson y Carmichael, 2016).

3.8.3.3 - Manejo del flujo

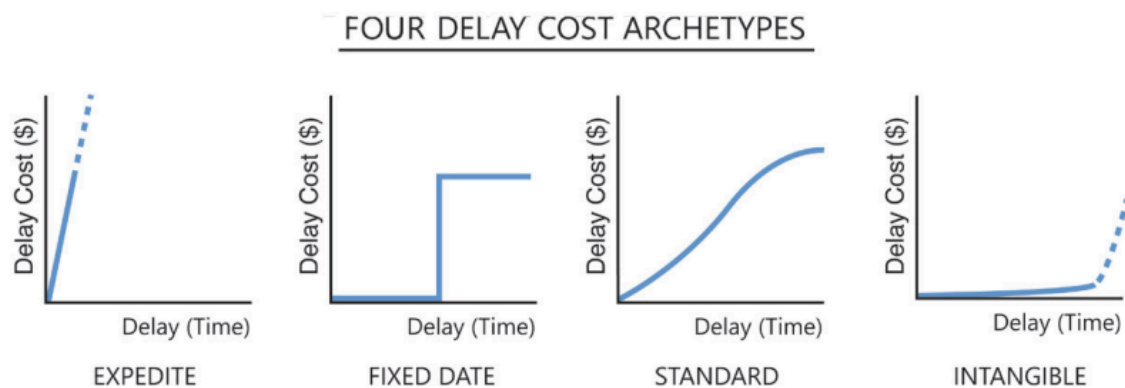
En un sistema Kanban, el objetivo es optimizar el flujo de trabajo para maximizar la entrega de valor, minimizar los tiempos de entrega y hacer que el proceso sea lo más predecible posible (Anderson y Carmichael, 2016). No obstante, estos objetivos pueden ser conflictivos, y debido a la complejidad de los entregables, es necesario un control empírico mediante la transparencia, inspección y adaptación continua. Es esencial prestar atención a cuellos de botella y bloqueos que restrinjan el flujo del trabajo (Anderson y Carmichael, 2016).

Un aspecto clave para mejorar este flujo es el costo del retraso (CoD), el cual se refiere al

valor que se pierde cuando una tarea se posterga por un tiempo determinado. La urgencia o el costo del retraso por unidad de tiempo es otro factor importante y puede variar según la longitud del retraso. Para gestionar este costo, Kanban clasifica los elementos de trabajo en cuatro arquetipos: urgente, fecha fija, estándar e intangible (Anderson y Carmichael, 2016). Estos arquetipos no sólo permiten priorizar tareas, sino también definir diferentes políticas de servicio. Además, la relación con los clientes del sistema es crítica, ya que valoran métricas como el tiempo de entrega, la tasa de defectos y la previsibilidad del servicio. Para mantener este equilibrio, los sistemas kanban suelen definir niveles de servicio que guían las expectativas y capacidades del sistema (Anderson y Carmichael, 2016). A continuación, se exponen cuatro gráficos a través de una figura donde se puede observar el comportamiento de estos cuatro arquetipos:

Figura 7

Los cuatro arquetipos de demora y costo



Nota. Reproducido de *Essential Kanban Condensed*, por Anderson y Carmichael (2016).

3.8.3.4 - Políticas explícitas

Anderson y Carmichael (2016) destacan la importancia de hacer explícitas las políticas dentro de un proceso Kanban. Estas políticas van más allá de la definición del flujo de trabajo, generando restricciones que, aunque limitan las acciones, permiten la optimización continua mediante la experimentación. Es fundamental que las políticas sean simples, visibles, siempre aplicadas y fácilmente modificables por quienes gestionan el servicio. Establecer límites de trabajo en progreso (WiP) es un ejemplo clave, pero la eficacia de estos límites depende de la capacidad para revisarlos y adaptarlos según las circunstancias (Anderson & Carmichael, 2016).

En paralelo, Anderson (2010) subraya que las políticas, aunque en su mayoría establecidas por la dirección, deben ser revisadas y ajustadas periódicamente, ya que pueden haber sido relevantes en su implementación inicial, pero perder validez con el tiempo si no se adaptan a nuevas realidades. Este enfoque garantiza que las políticas no se conviertan en barreras estáticas, sino en herramientas flexibles para mejorar el rendimiento del equipo (Anderson, 2010).

3.8.3.5 - Implementar bucles de feedback

En cualquier proceso controlado, los bucles de retroalimentación son esenciales, y su relevancia aumenta en escenarios de cambio evolutivo. Mejorar la retroalimentación en áreas clave, como la alineación estratégica, la coordinación operativa y la gestión de riesgos, es fundamental para mejorar los servicios (Anderson y Carmichael, 2016).

El sistema kanban establece siete tipos de cadencias, o ciclos de retroalimentación, que se refieren tanto a reuniones periódicas como al intervalo entre ellas. Según Anderson y Carmichael

(2016), las 7 cadencias o reuniones que establece Kanban son:

- Revisión estratégica: para la selección de los servicios a proveer.
- Revisión de operaciones: para entender el balance entre servicios, desplegando recursos para maximizar el valor de las entregas.
- Revisión de riesgos: se revisan los riesgos que pueden afectar la entrega efectiva de servicios.
- Revisión de la prestación de servicio: se evalúa la efectividad de un servicio.
- Reunión de reposición: en la misma se mueven ítems hacia la zona de los compromisos.
- La reunión Kanban: es una reunión auto organizada de coordinación diaria.
- Reunión de planificación de entrega: en donde se planean las entregas a los clientes.

4 - Pregunta De La Investigación

El objetivo de esta investigación es profundizar en los factores que impiden que los equipos de desarrollo de software en Argentina cumplan con los plazos de entrega establecidos en sus proyectos. El incumplimiento de los tiempos de entrega es una problemática recurrente en el desarrollo de software y afecta tanto a los resultados financieros como a la relación con los clientes, además de impactar negativamente en la productividad de los equipos. En este contexto, la pregunta principal que guía esta investigación es la siguiente:

Pregunta principal :

¿Cuáles son los factores más relevantes que influyen en el incumplimiento de los plazos de entrega en los equipos de desarrollo de software en Argentina?

Esta pregunta busca identificar, de manera precisa y contextualizada, las causas que generan retrasos en la finalización de los proyectos de software. Además, se intenta explorar cómo diferentes aspectos, tanto internos como externos a los equipos de desarrollo, afectan la capacidad para cumplir con las fechas de entrega.

5 - Hipótesis

En función del problema planteado y de la pregunta de investigación que guía este estudio, se formula la siguiente hipótesis, que intenta prever las posibles causas del incumplimiento de los plazos de entrega en los equipos de desarrollo de software en Argentina.

Hipótesis: La principal causa del fracaso en el cumplimiento de los plazos de entrega de software se debe a una estimación de trabajo realizada de forma errónea debido a la utilización de una metodología de trabajo no ágil y no apta para la naturaleza de su proyecto, sumado a la falta de conocimiento de los miembros acerca de la metodología apropiada.

6 - Metodología

La metodología empleada en esta investigación será de carácter cuantitativo, ya que el principal objetivo es generar datos numéricos interpretados en distintas variables, que puedan ser transformados en estadísticas confiables y utilizables. Esto permitirá la formulación de hechos y el descubrimiento de patrones relacionados al objeto de estudio.

Además, la metodología cuantitativa facilita la recolección de información de un número significativo de participantes, proporcionando una base sólida y representativa para el análisis de los resultados (Hernández Sampieri, 2014).

El uso de esta técnica es llevado a cabo para esta investigación de forma virtual, a través de “Formularios de Google” y a modo de encuesta con una serie de preguntas muy específicas y aplicadas a la investigación. Para garantizar la seguridad y confidencialidad de los datos se ha optado por mantener un carácter anónimo sobre la misma.

6.1 - Participantes

La muestra, según Hernández Sampieri (2014) “un subgrupo de la población o universo” (p. 171), está compuesta por un total de 167 participantes, los cuales han sido cuidadosamente seleccionados de modo tal que todos apliquen al universo en cuestión. Todos los encuestados desarrollan su actividad en torno al desarrollo de software en Argentina, como bien fuera objeto de la presente investigación.

6.2 - Tipo De Muestra

El tipo de muestra seleccionado para esta investigación es no probabilística, dado que no

todos los integrantes de la población tienen las mismas probabilidades de ser seleccionados (Hernández Sampieri, 1983), sino que el investigador debió seleccionar a los participantes en torno a determinados criterios para entrevistarlos mediante encuestas, con el criterio de que trabajen en equipos de desarrollo de software evolutivo y en Argentina.

Para llevar a cabo esta investigación, y en este caso, las encuestas que componen su fuente de información, se han definido una gran cantidad de variables, que serán de gran ayuda para poder encontrar patrones de comportamiento en torno a los factores que afectan el cumplimiento de las fechas de entrega. Las mismas se encuentran distribuidas en la encuesta a lo largo de ocho secciones (Anexo 1) que han sido definidas con el fin de agruparlas lógicamente según su contenido e intención de la siguiente manera:

1. Datos demográficos y contextuales: En esta sección se busca comprender el contexto en el cual desempeña sus tareas laborales el entrevistado. Si bien la misma es de carácter introductorio, es fundamental a los fines de la investigación para poder establecer patrones a la hora de sacar conclusiones.

2. Características del proyecto: Aquí de un modo un poco más específico, se recopila información acerca de las características del proyecto para el cual el entrevistado desempeña sus tareas, ya sea la naturaleza del mismo, la duración o el presupuesto. Estos factores pueden ser determinantes a la hora de encontrar patrones.

3. Metodologías y Prácticas de Trabajo: Este bloque intenta principalmente identificar si el entrevistado realiza sus tareas bajo una metodología tradicional o ágil, y en tal caso cuál de ellas, para poder realizar un análisis más profundo del comportamiento de su equipo.

4. Composición y Dinámica del Equipo: Este apartado pretende comprender mejor la

estructura del equipo donde lleva a cabo las tareas el entrevistado, de forma tal que puedan hallarse defectos de diseño o incompatibilidades con la metodología que ejecuta.

5. Estimaciones: De forma similar a la sección anterior, este bloque será determinante para comprender el modo en que el equipo del entrevistado realiza sus estimaciones, para encontrar la relación entre las mismas y el cumplimiento de los plazos de entrega, así como también hacer hallazgos de incompatibilidades entre lo ejecutado y lo que la metodología utilizada determina.

6. Interacción con Otros Equipos: Aquí, de forma no tan específica o profunda quizás como las anteriores secciones, se busca comprender cómo es la relación e interacción del entrevistado con otros equipos, ya que esto puede ser un factor retardante para los equipos a la hora de cumplir plazos si fuera mal ejecutado.

7. Factores de Fracaso: Este bloque busca de lleno comprender y recopilar cuáles son los factores que generan el fracaso en el cumplimiento de plazos de entrega para el equipo del entrevistado.

8. Evaluación y Mejoras: En este último espacio, se busca relevar información acerca de qué estrategias son ejecutadas para la revisión y mejoras de los procesos del equipo del entrevistado.

7 - Resultados De La Investigación

7.1 - Recolección De Datos

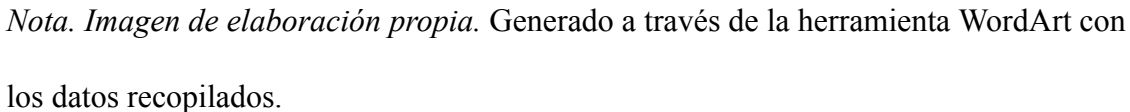
Las respuestas generadas sobre el formulario de la entrevista se han ido almacenando de forma automática a medida que los entrevistados efectuaban el completado de la misma, lo cual es el funcionamiento habitual de la plataforma utilizada. Esta herramienta permite descargar esta información a una hoja de cálculo para efectuar el análisis de los datos, ordenarlos, realizar gráficos, e incluso utilizarlos como base de datos. Además, todas las respuestas han sido recopiladas de forma anónima para garantizar al entrevistado la protección de sus datos.. En esta sección, se procederá a un análisis inicial de los resultados recolectados a partir de las métricas ofrecidas por los Formularios de Google.

7.1.1 - Puesto Del Entrevistado

Debido a que esta pregunta esperaba una respuesta de texto libre donde el entrevistado pudiera ingresar el nombre del cargo que desempeña, la variedad de resultados ha resultado muy grande. Es por ello que, en primera instancia, se ha hecho uso de una herramienta en línea de software gratuito para la creación de nubes semánticas, llamada WordArt. Gracias a este software, se puede obtener para este caso una visualización gráfica de todos los puestos donde los tamaños de letra expuestos en la imagen son proporcionales al peso, o bien la frecuencia, de cada puesto ingresado como respuesta.

Se expone a continuación una figura con la imagen generada gracias a WordArt:

Puestos desempeñados por los entrevistados



Como puede verse gracias a la ayuda de la herramienta, los principales puestos que rápidamente destacan entre las respuestas son a simple vista Project Managers, Programadores y QAs. Ahora bien, este método es interesante para un primer abordaje pero no permite extraer métricas concretas sobre los resultados. Por este motivo, se ha procedido a efectuar un proceso de normalización de los datos sobre los resultados de esta pregunta, asignando una categoría genérica a cada una de las respuestas. El proceso fue efectuado directamente sobre la hoja de cálculo exportada de formularios de Google y la información se ha analizado gracias a una tabla dinámica, también herramienta de las hojas de cálculo de Google.

Se expone a continuación una tabla confeccionada luego de haber normalizado la información :

Tabla 1

Recuento de puestos normalizados por categoría

Puesto desempeñado	Recuento
Administrador de Sistema	1
Agile Coach	2
Analista Funcional	9
Asesor	1
Business Analyst	1
Consultor SAP	2
Coordinador	2
Devops	3
Director	5
Diseñador	3
Docente	1
Gerente	10
Ingeniero en Analytics	1
Lider de equipo	21
Otro	8
Producto	4
Programador	53
Project Manager	14
QA	15
Scrum Master	1
Seguridad	2
Soporte	7

Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas

realizadas.

A través de la tabla expuesta, se logra determinar que del total de los entrevistados un 32% de ellos se desempeña como programadores, siendo este el puesto más repetido entre las respuestas normalizadas. En segundo lugar, podemos encontrar a los líderes de equipo, abarcando un total del 13% de las respuestas normalizadas. Luego con porcentajes similares, se encuentran los Project Managers y QAs con 8% y 9% respectivamente. Puestos como Analista Funcional, Gerente y soporte oscilan entre el 4 y 6 por ciento de los resultados. El resto de los puestos que pueden observarse en la tabla, no ocupan porcentajes tan grandes como los previamente mencionados. Este análisis ayuda mucho a contextualizar y comprender el tipo de personas que han colaborado con estas entrevistas, las cuales se ajustan según lo esperado a los perfiles esperados para esta investigación.

7.1.2 - Tipo De Empresa

De la mano de esta pregunta, se intenta comprender y contextualizar cuál es el tamaño de las empresas donde trabajan las personas encuestadas. Esto resultará de gran importancia a la hora de hacer un análisis más profundo de los resultados e intentar ahondar en patrones.

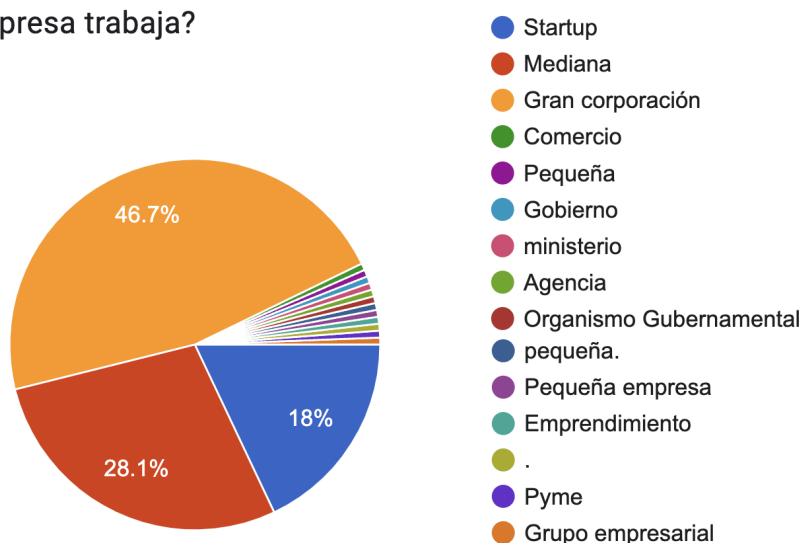
A continuación, se puede visualizar a través de la Figura 9 la distribución de estas respuestas.

Figura 9

Resultados de la Pregunta Nro. 2 sobre el tamaño de la organización.

¿En qué tipo de empresa trabaja?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Como puede apreciarse en la imagen un 46,7% de los entrevistados trabajan en empresas u organizaciones de gran tamaño, mientras que un 28,1% de ellos trabaja en empresas medianas y otro 18% de ellos en empresas pequeñas.

7.1.3 - Sector De La Empresa

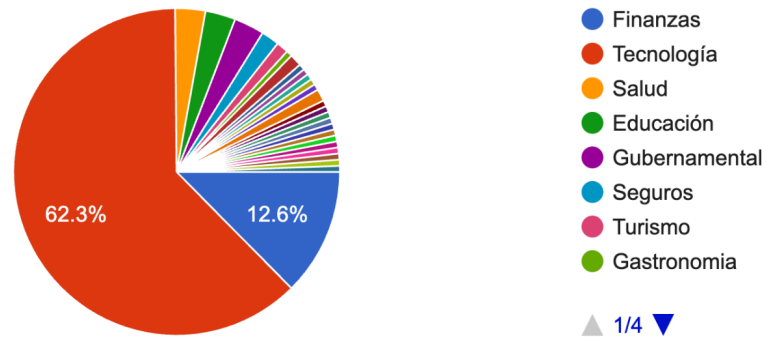
En esta sección, se busca comprender en qué tipo de organizaciones desempeñan sus actividades los participantes de la encuesta. Es decir, más allá del tamaño de estas organizaciones que fuera previamente evaluado, su naturaleza y la naturaleza de sus actividades. En este caso ha resultado muy útil y sencillo de ver a través de las métricas de formularios de Google donde se enfoca el mayor porcentaje de los encuestados, lo cual puede verse en la figura 10 a continuación.

Figura 10

Resultados de la Pregunta Nro. 3 sobre el sector donde se desempeña la organización.

¿En qué sector se encuentra su empresa?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

En este caso resulta muy sencillo apreciar que más de la mitad de los encuestados, en particular el 62,3% trabajan en empresas del sector de tecnología. Otro 12.6% de ellos desempeña sus actividades en empresas u organizaciones del rubro financiero. En este caso, no se resalta el porcentaje abarcado por el resto de los encuestados ya que estos arrojan datos muy dispersos mientras que esta investigación intenta abordar patrones, motivo por el cual se toma como importantes los porcentajes que ya han sido resaltados.

7.1.4 - Tipo De Proyecto

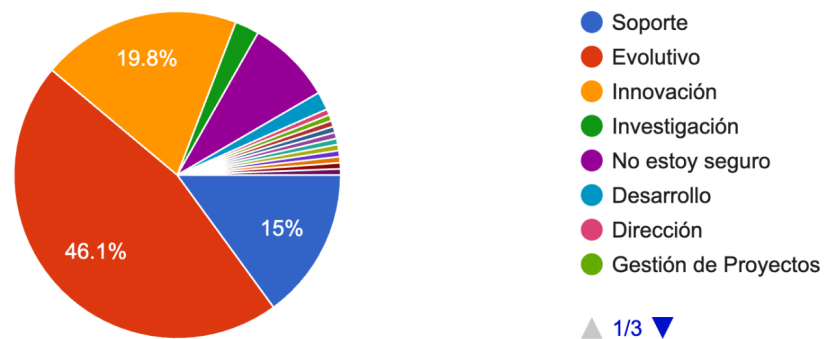
Esta pregunta ha no sólo recopilado respuestas acerca del tipo de proyecto donde se desempeñan los encuestados, sino que también ha ayudado a comprender qué grado de noción tienen las personas acerca de la naturaleza de los proyectos donde se desempeñan. Obsérvese en primer lugar los resultados expuestos a continuación a través de la figura 11:

Figura 11

Resultados de la Pregunta Nro. 4 sobre la naturaleza del proyecto.

¿En qué tipo de proyecto trabaja principalmente?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Como puede notarse en la figura, un 46,1% de las personas encuestadas trabajan en proyectos del tipo desarrollo de software evolutivo, pasando a ser esta la naturaleza más repetida entre los resultados. Por otro lado, un 19,8% de la muestra trabaja en proyectos de innovación, la cual era una de las categorías enlistadas para elegir como respuesta. En tercer lugar, con un porcentaje sutilmente menor, un 15% de los encuestados trabajan en proyectos de soporte.

Resulta muy importante destacar, que un 8,4% de los encuestados ha escogido “No estoy seguro” como respuesta, lo que podría indicar que, o bien, no han sido provistos con la capacitación correspondiente para comprender la clasificación de su proyecto, o bien, podrían estar trabajando en un ambiente caótico donde conviven diferentes naturalezas de proyectos en simultáneo.

7.1.5 - Duración De Los Proyectos

Esta pregunta intenta obtener respuesta al tiempo promedio que duran los proyectos donde trabajan las personas que han sido entrevistadas. Esto ayudará a contextualizar que tan cambiante son los entornos donde se desempeñan estas personas. Se expone a continuación una figura con los resultados recopilados para esta pregunta:

Figura 12

Resultados de la Pregunta Nro. 5 sobre la duración del proyecto.

¿Cuál es la duración promedio de los proyectos en los que trabaja?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Obsérvese que del total de los 167 entrevistados, un 38,9% de los mismos ha respondido que el o los proyectos en donde trabaja, duran más de 12 meses. Por otro lado, un 17,4% ha contestado que los proyectos donde trabaja duran entre 6 y 12 meses. Esto supone algo bueno, porque si tenemos en cuenta ambos resultados, más del 50% de los encuestados tiene proyectos estables, es decir están menos expuestos al cambio. Por otro lado, un 16,8% señalan que sus

proyectos tienen entre 3 y 6 meses de duración, y un 19,8% de los entrevistados participan en proyectos de menos de 3 meses de duración.

7.1.6 - Presupuesto Del Proyecto

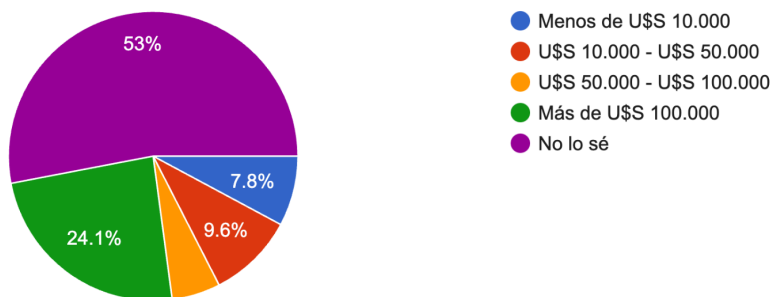
Esta pregunta fue diseñada para poder contextualizar los recursos económicos con que cuentan los proyectos donde se desempeñan los encuestados. Resulta de gran interés destacar que más de la mitad de las personas, no conocían el presupuesto de su proyecto. Si bien esto es algo que podría ser normal, permite también entrever el poco contexto que tienen a veces las personas sobre el proyecto. A continuación se expone una figura para visualizar las métricas para esta pregunta.

Figura 13

Resultados de la Pregunta Nro. 6 sobre el presupuesto del proyecto.

¿Cuál es el presupuesto promedio de los proyectos en los que trabaja? (Expresado en dólares estadounidenses)

166 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Puede observarse como fuera mencionado, que un 53% de los encuestados desconocen

está información. Por otro lado, de los restantes el mayor porcentaje con un total de 24,1% de ellos asegura que su proyecto tiene un presupuesto de más de 100.000 dólares estadounidenses, lo cual supone un presupuesto muy grande. El aproximadamente 23% restante, trabaja en proyectos con presupuestos menores a 100.000 dólares estadounidenses, siendo el 5,4% proyectos de entre 50.000 y 100.000 dólares, el 9,6% oscilando entre los 10.000 y 50.000 dólares, y en último lugar los de más bajo presupuesto ocupando un 7,8% sobre el total.

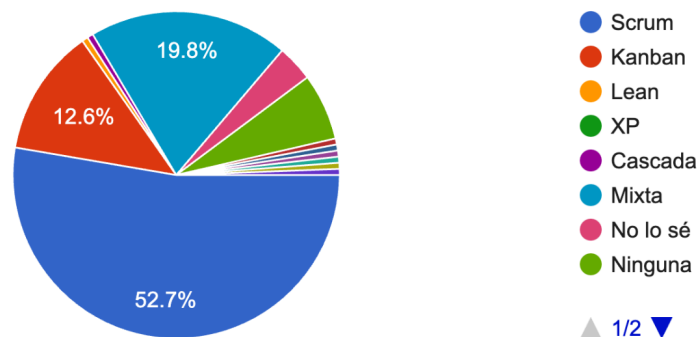
7.1.7 - Metodología De Trabajo Utilizada

Si bien es sencilla, esta pregunta está orientada a poder identificar principalmente si las personas trabajan utilizando metodologías ágiles o tradicionales. Se expone en primer lugar, la figura 14 donde se puede visualizar con claridad las metodologías predominantes:

Figura 14

Resultados de la Pregunta Nro. 7 sobre la metodología de trabajo utilizada.

¿Qué metodología de trabajo utiliza principalmente?
167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Resulta de esta pregunta una de las primeras grandes afirmaciones que esta investigación podrá hacer, que es que sólo el 0,6% de las personas trabaja bajo una metodología en cascada. Este 0,6%, representa a sólo una persona de las 167 encuestadas. Esto coincide con lo expuesto en el marco teórico de esta investigación, donde podemos comprender que las metodologías ágiles son las que comprenden la mayor participación en la ingeniería de software en la actualidad. Por otro lado, del 99,4%, sólo un 3,6% de ellos ha respondido no saber bajo qué metodología trabaja. Esto nos deja finalmente con la capacidad de afirmar que un 95,8% de los encuestados trabajan con metodologías ágiles. De ese porcentaje, un 52,7% trabajan utilizando el marco de trabajo Scrum y un 12,6% utilizando Kanban, la dos metodologías fuertemente destacadas por el marco teórico de esta investigación.

7.1.8 - Capacitación En La Metodología Utilizada

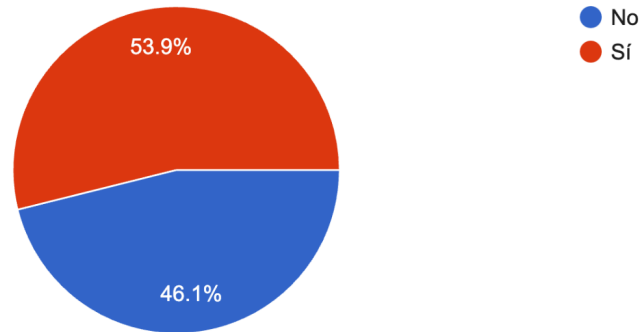
Una de las cuestiones de importancia a la hora de trabajar con estas metodologías ágiles, como bien lo destacan muchas de las bibliografías anteriormente citadas, es el grado de capacitación que tengan los miembros de los equipos sobre esta. Eso es en efecto, lo que busca abordar esta pregunta. Obsérvese la siguiente figura, donde se exponen los resultados a esta pregunta.

Figura 15

Resultados de la Pregunta Nro. 8 sobre la capacitación en la metodología utilizada.

¿Ha recibido capacitación formal sobre la metodología que utiliza?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Sin mucho que agregar por el momento, se puede observar que el 46,1% de los participantes no ha recibido capacitación formal para las metodologías en las cuales le está siendo requerido trabajar. En cambio, un 53,9% de los encuestados si ha recibido capacitación al respecto.

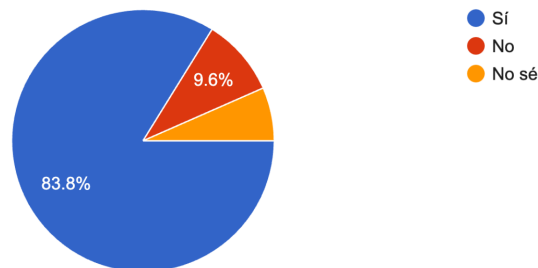
7.1.9 - Adaptaciones A La Metodología

Esta pregunta busca comprender en qué porcentaje los equipos de trabajo tienden a adaptar las metodologías para utilizarlas en su dinámica. Esto permitirá en un posterior análisis más exhaustivo comprender cómo esto pudiera afectar el funcionamiento del equipo, y en consecuencia, el cumplimiento de los plazos de entrega. Se observa a continuación una figura con el resultado de las respuestas:

Figura 16

Resultados de la Pregunta Nro. 9 sobre la modificación de las metodologías ágiles.

¿Ha adaptado o personalizado la metodología utilizada para ajustarse mejor a las necesidades del proyecto?
167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

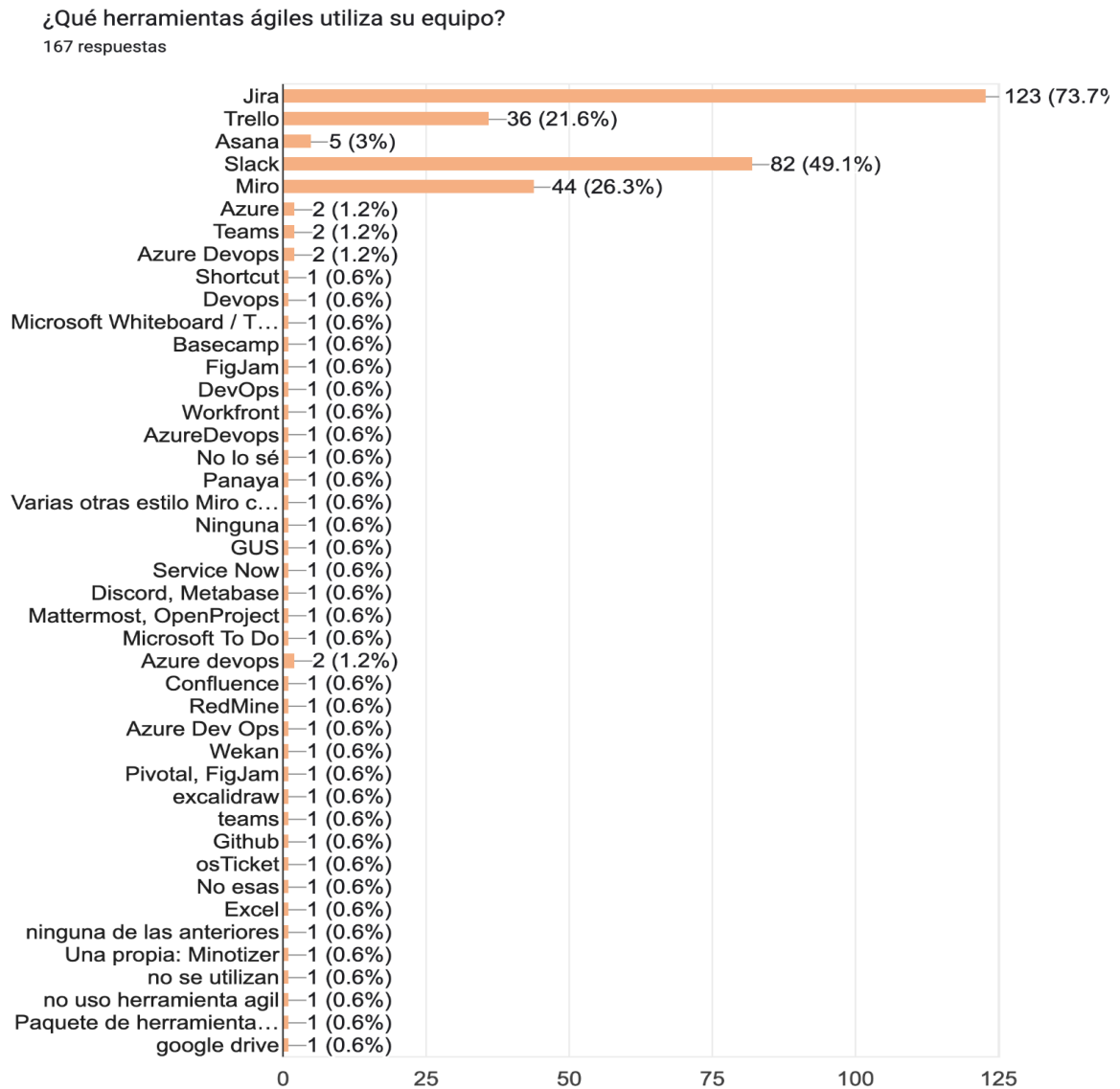
Puede notarse que un muy elevado número de encuestados afirma que la metodología ágil ha sido adaptada para utilizarse en su equipo. Esto supone un 83,8% de los encuestados. Sólo un 9,6% de ellos afirma no haber realizado modificaciones a la metodología. Esto será examinado en detalle a posterior, para el hallazgo de patrones donde esto pudiera de algún modo estar perjudicando a estos equipos y a su cumplimiento de plazos de entrega.

7.1.10 - Herramientas

Aquí podremos encontrar una recopilación de respuestas acerca de cuáles son las herramientas ágiles más utilizadas por los equipos correspondientes a las personas encuestadas. Resultará a posterior, de suma utilidad a la hora de hallar patrones de comportamiento y sus resultados. Se expone a continuación una figura con los resultados:

Figura 17

Resultados de la Pregunta Nro. 10 sobre las herramientas ágiles más utilizadas.



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Puede observarse como herramienta número uno a Jira, mencionada por un total de 123 encuestados, es decir, el 73,7% de ellos. Por otro lado Slack, Miro y Trello también registran grandes porcentajes de utilización siendo estos 49,1%, 26,3% y 21,6% respectivamente. Pero es innegable que Jira es la herramienta más utilizada por los equipos correspondientes a las personas encuestadas.

7.1.11 - Tamaño Del Equipo

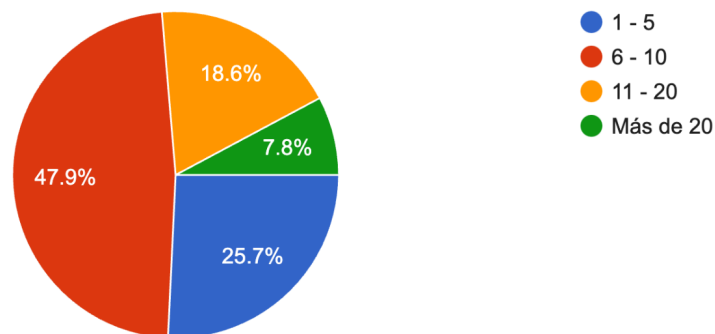
Esta pregunta está orientada a conocer el tamaño de los equipos correspondientes a la muestra, es decir, los encuestados. Obsérvese a continuación, una figura con los resultados:

Figura 18

Resultados de la Pregunta Nro. 11 el tamaño promedio de los equipos de trabajo

¿Cuál es el tamaño promedio de su equipo de trabajo?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Como se puede observar en la imagen el tamaño predominante de los equipos con un

resultado del 47,9% es de entre 6 y 10 personas, lo cual según Sutherland (2014) podríamos concluir que está bien, o incluso podría estar al límite, ya que lo que él indica es un equipo de siete miembros pudiendo agregar o reducir hasta 2 de ellos. En este caso y tomando eso como parámetro válido, si sumamos el 47,9% que corresponde a los equipos de entre 6 y 10 personas y a eso le adicionamos el 25,7% que corresponde a equipos entre 1 y 5 personas, podemos concluir que de los encuestados, un total de 71,6% trabaja en equipos con tamaños dentro de los estándares para equipos de Scrum. Por otro lado podemos observar que una cantidad mucho menor pero notable, trabaja en equipos de mucho mayor tamaño donde el 18,6% tiene entre 11 y 20 miembros y un 7,8% de ellos más de 20 miembros. Este grupo será de gran interés para las conclusiones de esta investigación, por lo cual se continuará en una sección posterior con su análisis.

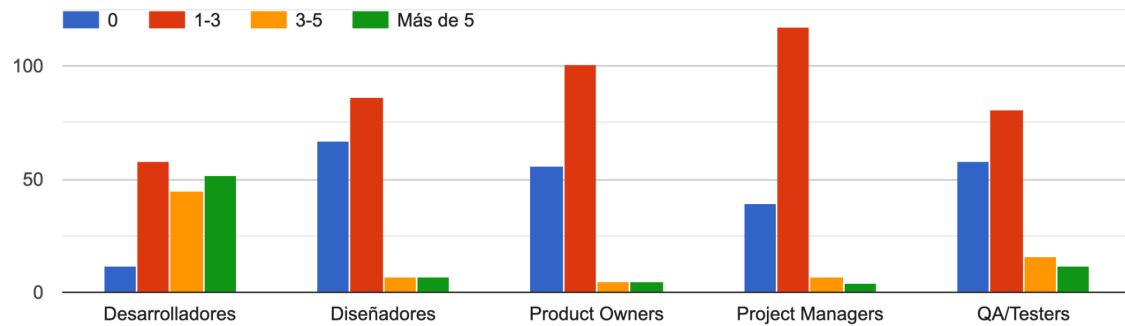
7.1.12 - Cantidad De Miembros Según Su Tipo

Aquí la intención ha sido desglosar los resultados obtenidos en la pregunta previa, de modo tal de comprender la composición de esos equipos. Se presentan a continuación los resultados de la misma:

Figura 19

Resultados de la Pregunta Nro. 12 la cantidad de miembros del equipo según su tipo

¿Cuál es la cantidad que posee su equipo de cada uno de estos puestos? Responda con números



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Puede observarse que, para todos los casos a excepción de los desarrolladores, la respuesta más común ha sido entre 1 y 3 personas para el puesto. En el caso de los desarrolladores, puede concluirse que su cantidad es muy variable entre 1 y más de 5 de ellos, pero lo que con seguridad no es común de encontrar entre las respuestas, es un equipo con ausencia de desarrolladores.

Por otro lado, en lo que respecta a los puestos de diseñadores, product owners, project managers y testers, puede observarse que en la mayor cantidad de respuestas se informa que hay entre 1 y 3 de ellos por equipo, y a diferencia de los desarrolladores hay una elevada cantidad de respuestas donde su cantidad es cero, pudiendo así darse a entender que en muchos casos es común también que estos puestos no estén presente.

7.1.13 - Frecuencia De Las Reuniones De Seguimiento

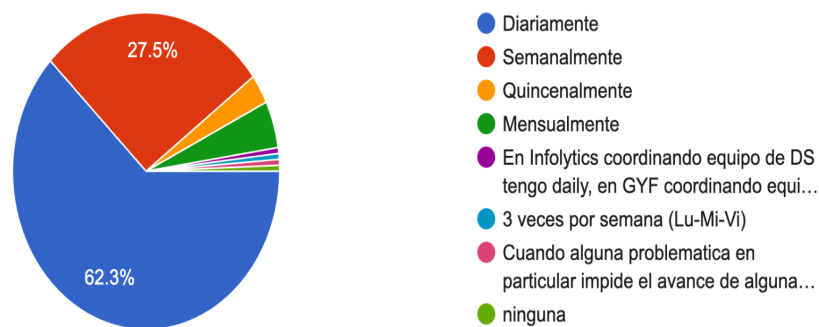
Esta pregunta tiene como objetivo, comprender que tan a menudo los equipos correspondientes a las personas encuestadas llevan a cabo reuniones de seguimiento. Véase a continuación su resultado:

Figura 20

Resultados de la Pregunta Nro. 13 frecuencia de las reuniones de seguimiento

¿Con qué frecuencia realiza reuniones de seguimiento?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Como puede observarse en la imagen expuesta anteriormente, es muy sencillo notar que la mayoría de los encuestados realizan reuniones de seguimiento de forma diaria, para ser precisos un 62,3% de ellos lo llevan a cabo de este modo. A su vez, un 27,5% de los encuestados realizan su sincronización de forma semanal, mientras que todo el pequeño porcentaje restante tiene una frecuencia aún mayor, que será analizada también como parte de los patrones que

cumplen los equipos que no logran cumplir las fechas de entrega.

7.1.14 - Canales De Comunicación

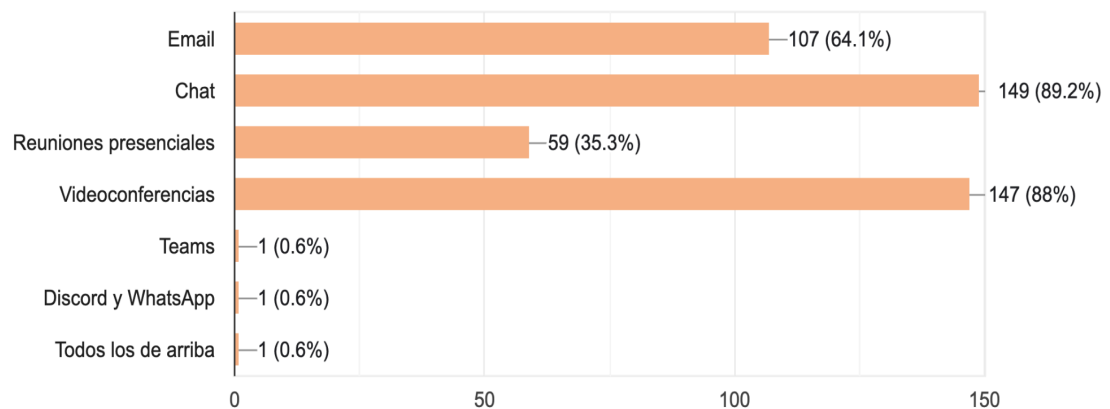
A modo de ampliar más el contenido de la pregunta anterior, esta sección busca comprender por medio de qué herramientas son llevadas a cabo las reuniones de seguimiento que realizan los equipos de los encuestados. Podemos ver su resultado en la siguiente figura:

Figura 21

Resultados de la Pregunta Nro. 14 Herramientas utilizadas para la comunicación

¿Qué canales de comunicación utiliza su equipo?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Gracias al resultado obtenido en esta pregunta, puede notarse que en su gran mayoría, los equipos de los encuestados utilizan el chat y las videoconferencias como canal de comunicación. Para ser más precisos, un 88% de los encuestados ha afirmado utilizar videoconferencias y un

89,2% sostiene utilizar el chat como medio de comunicación. Si bien en menor porcentaje, pero no por ello menos importante, se puede observar también un 64,1% de respuestas en donde las personas han afirmado utilizar email como canal de comunicación. Por último, en un porcentaje mucho menor se encuentran las reuniones presenciales con un 35,3%, que si bien es un número mucho más bajo que el obtenido para las videoconferencias, puede interpretarse esto como el resultado de la aparición de tanto teletrabajo.

7.1.15 - Método De Estimación

Ahondando de forma más precisa en la forma en que los equipos de los encuestados realizan sus estimaciones, esta pregunta permitirá comprender en qué medida cada uno de ellos adopta una técnica distinta de estimación. A continuación sus resultados son expuestos a través de una figura:

Figura 22

Resultados de la Pregunta Nro. 15 Método de estimación

El equipo para el que trabaja, realiza las estimaciones :
167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Resulta interesante analizar estos resultados en el contexto de esta investigación donde más de un 50% de los encuestados trabajan con metodología ágil y en particular Scrum. Puede observarse que un 53,9% de las respuestas indican realizar las estimaciones de forma ágil, lo cual coincide en gran medida con el más de 50% obtenido de las respuestas acerca de la metodología de trabajo utilizada en donde este fue Scrum. Por otro lado, un 34,1% indica realizar sus estimaciones asignando una unidad de tiempo a las mismas, mientras que un 10,2% afirma directamente no realizar estimaciones. Ambos dos grupos serán de importante interés al momento de hallar patrones de comportamiento en futuras secciones.

7.1.16 - Personal A Cargo De Las Estimaciones

Continuando por la línea de las estimaciones, esta pregunta busca comprender con un mayor grado de detalle, quienes son los encargados de llevar a cabo las estimaciones en cada uno de los equipos correspondientes a las personas encuestadas. Para ello, se adjunta una figura con los resultados de la pregunta :

Figura 23

Resultados de la Pregunta Nro. 16 Encargados de las estimaciones

Las estimaciones son realizadas por...

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

En base a los resultados obtenidos de esta pregunta, puede observarse que sólo el 46,1% de los encuestados ha respondido que las estimaciones son realizadas por todo el equipo, sea cual sea el cargo de las personas. Puede verse como el porcentaje empieza a ser menor que el de las personas que indicaron utilizar Scrum, lo cual podría significar que el diferencia correspondería a encuestados que podrían no llevar a cabo una implementación de Scrum donde todo el equipo participa de la estimación. Por otro lado, puede observarse también que dos grandes porcentajes de encuestados han respondido que las estimaciones son realizadas sólo por los desarrolladores o por los líderes técnicos, con un 16,8% y un 19,2% respectivamente.

7.1.17 - Ejecución De La Estimación

Al igual que en la pregunta anterior, a través de esta sección se busca ahondar aún más en el método utilizado para ejecutar las estimaciones. En este caso en particular, se busca comprender de qué modo los equipos de los encuestados llevan a cabo las estimaciones de

trabajo, en particular, como asignan cada una de sus tareas a alguna unidad de medida. Obsérvese a continuación una figura con las respuestas obtenidas:

Figura 24

Resultados de la Pregunta Nro. 17 Método de ejecución de las estimaciones

El trabajo es estimado en:
167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Entre las respuestas, puede destacarse que un 44,3% de los encuestados trabajan en equipos cuyas estimaciones son realizadas en Story Points entendiendo el peso o esfuerzo asociado para llevar a cabo cada una de las tareas, lo cual coincide con los conceptos previamente abordados acerca de las estimaciones a través de Poker Planning. Por otro lado, es importante destacar que un 12% de las personas indicó que en sus equipos se estima utilizando Story Points como unidad de medida, pero asocian estos directamente a una unidad de tiempo, lo cual según Cohn (2006) no es lo más óptimo ya que como él explica cuando habla acerca de los días ideales, el tiempo necesitado por una persona para llevar a cabo una tarea o acción, no es el mismo que el

que pueda requerir otra persona. Es por eso que ese 12% de los encuestados representarán un grupo de interés para esta investigación en una posterior sección, en función de encontrar qué patrones cumplen a la hora de alcanzar una fecha de entrega. Asimismo, un 38,3% de los participantes de la encuesta asegura que sus equipos trabajan estimando directamente con unidades de tiempo que son asignadas a cada tarea.

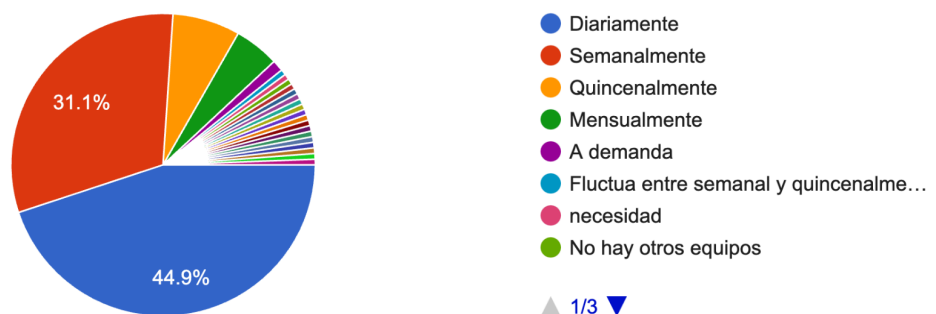
7.1.18 - Interacción Con Otros Equipos

Esta pregunta intenta comprender con que frecuencia los equipos de los entrevistados interactúan con otros equipos con los cuales pueden tener dependencia. Sus resultados serán importantes a la hora de comprender cómo afecta su mayor o menor frecuencia en términos de cumplimientos de fechas de entrega. A continuación se adjunta la figura correspondiente a los resultados de esta pregunta:

Figura 25

Resultados de la Pregunta Nro. 18 Frecuencia de interacción con otros equipos

¿Con qué frecuencia interactúa con otros equipos?
167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Puede concluirse que al igual que en la pregunta donde se intentó comprender la frecuencia con que los equipos realizaban reuniones de seguimiento, en este caso los resultados también indican que los equipos de los encuestados realizan en su mayoría sincronizaciones diarias o semanales con otros equipos. En particular, el mayor porcentaje de ellos lo hace de forma diaria, siendo representado por un 44,9%. En cuanto a los equipos que interactúan con otros grupos de forma semanal, estos representan un 31,1% de los resultados. Entre los porcentajes apreciables, puede destacarse también un 7,2% realizándose de forma quincenal, y un 4,8% de forma mensual.

7.1.19 - Evaluación De Coordinación De Tareas Con Otros Equipos

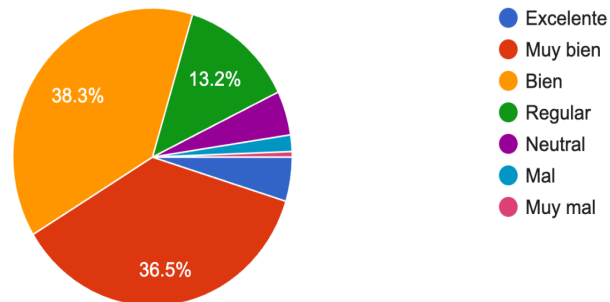
En función de comprender cómo la interacción con otros equipos impacta positiva o negativamente en los resultados de cumplimiento de fechas de entrega, esta pregunta está orientada a entender que tan bien están coordinadas las tareas entre equipos. Pueden verse a continuación los resultados de la pregunta a través de una figura:

Figura 26

Resultados de la Pregunta Nro. 19 Evaluación de los encuestas sobre la coordinación de tareas con otros equipos

¿Qué tan bien coordinadas están las tareas entre su equipo y otros equipos?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

En este caso, puede observarse que ningún porcentaje significativo de los encuestados en primera medida ha escogido “Excelente” como su opción de respuesta. En cambio, sólo un 36,5% de ellos ha respondido tener tareas muy bien coordinadas con otros equipos, mientras que un 38,3% sostiene que las tareas solamente están bien coordinadas. Por otro lado, todo el porcentaje restante de participantes indica una coordinación de tareas regular o incluso inferiores, ocupando las regulares un 13,2% de ellas. Estos grupos de encuestados serán analizados en mayor detalle para la examinación de patrones y su impacto en las fechas de entrega.

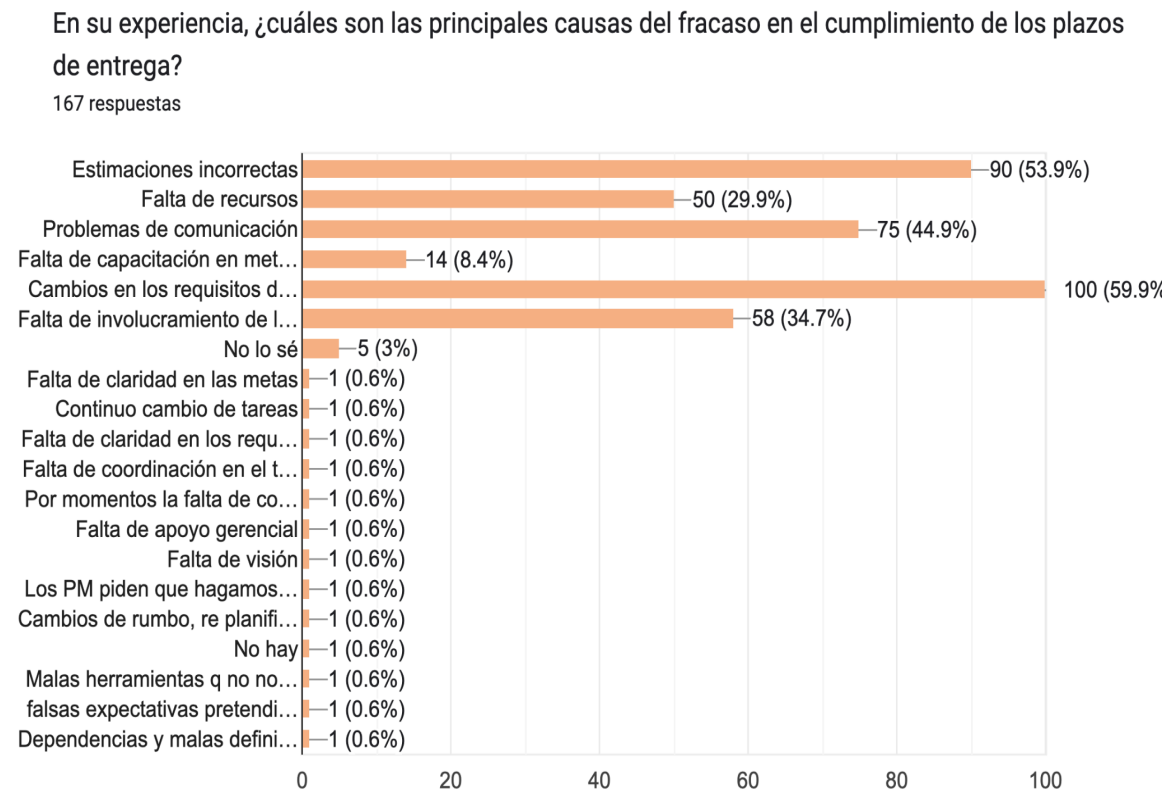
7.1.20 - Principales Causas De Incumplimiento De Fechas De Entrega

Esta pregunta representa uno de los puntos más importantes para esta investigación y sus resultados. En ella, se intenta obtener de forma directa de parte de los encuestados cuáles son en su experiencia los principales factores que causan que los equipos de los cuales forman parte no puedan cumplir con determinadas fechas de entrega. A continuación, se presentan por medio de

una figura los resultados obtenidos.

Figura 27

Resultados de la Pregunta Nro. 20 Principales causas de incumplimiento de fechas de entrega



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Tal como puede observarse en la figura, un muy importante número de los encuestados han indicado como principales factores los “cambios en los requisitos del proyecto”, las “estimaciones incorrectas” y “problemas de comunicación”, con un 59,9%, un 53,9% y un 44,9%

respectivamente. En los tres casos estamos hablando de un número muy alto de los encuestados, lo que permite con seguridad concluir que estos son factores influyentes para la mayor cantidad de los equipos a la hora de incumplir con un plazo de entrega. A su vez, aunque no en cantidades tan altas, un número considerable de los encuestados ha señalado la “falta de involucramiento de los stakeholders” y la “falta de recursos” como factores influyentes, con un 34,7% y un 29,9% respectivamente.

7.1.21 - Frecuencia De Incumplimiento De Plazos

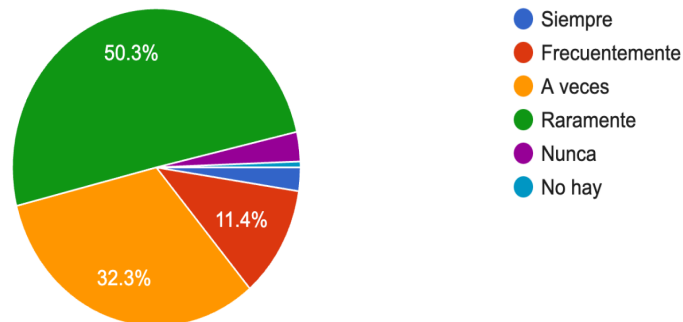
A partir de la interrogante planteada en esta pregunta, se busca no sólo comprender con qué frecuencia los equipos fallan en el cumplimiento de los plazos de entrega, sino también el mero hecho de que fallen. Es decir, aún si fallaran con poca frecuencia, esos fallos son también objeto de estudio de esta investigación. Obsérvese a continuación el resultado obtenido para esta pregunta:

Figura 28

Resultados de la Pregunta Nro. 21 Frecuencia de incumplimiento de fechas de entrega

¿Con qué frecuencia su equipo no cumple con los plazos de entrega?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

A diferencia de otras preguntas, donde las mayores conclusiones eran obtenidas por los resultados más altos, resultará de gran interés aquí enfocarse en más profundidad en las respuestas pertenecientes a los equipos que están más lejos de cumplir las fechas de entrega siempre. Es decir, más allá de que un 50,3% de los encuestados afirma que raramente sus equipos no cumplen los plazos de entrega, y sólo un 3,6% indica nunca fallar en ello, resta aún un 46,1% de ellos que son objeto de esta investigación para determinar los patrones que cumplen, o bien, no cumplen, para llegar a estos resultados no deseados.

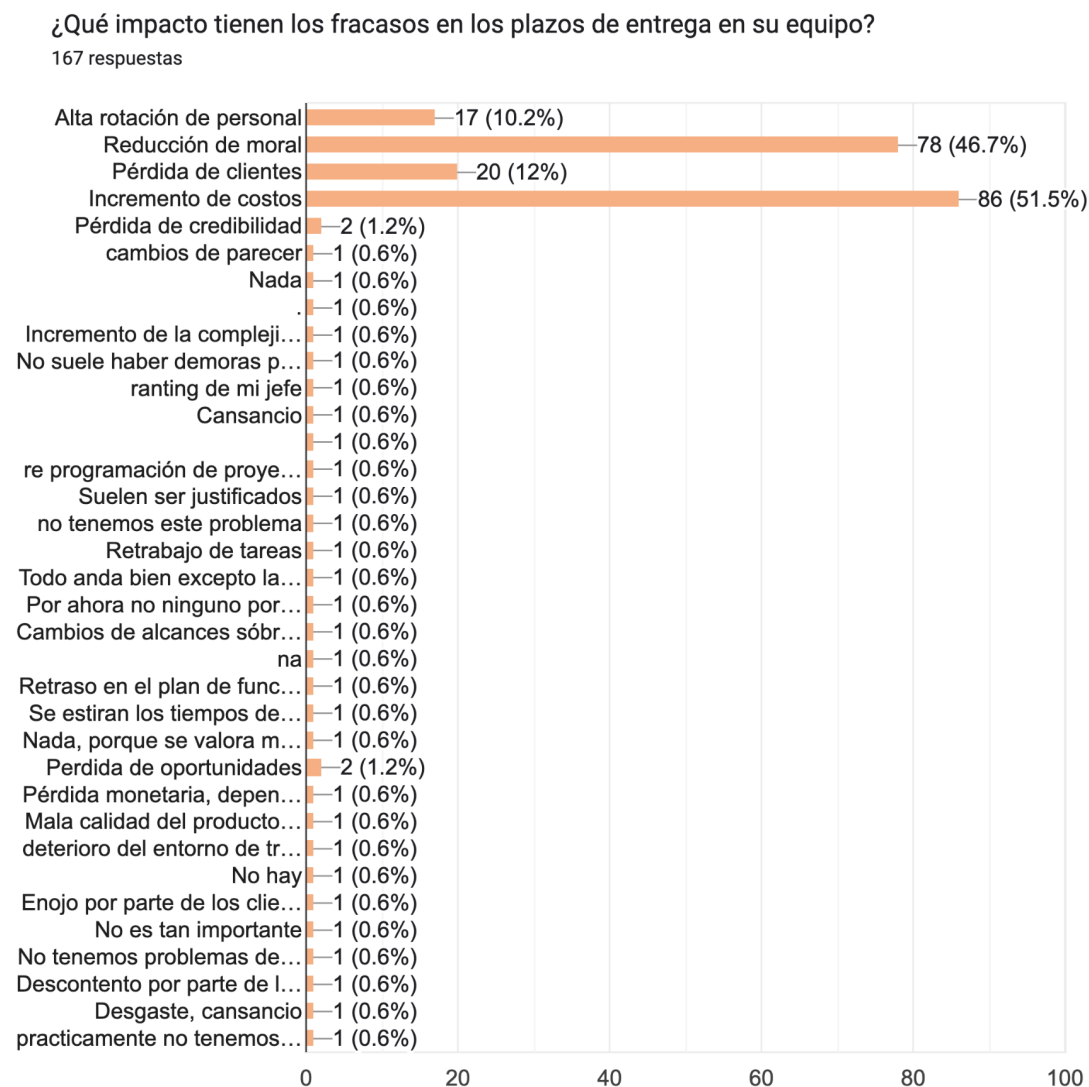
7.1.22 - Impacto Del Incumplimiento De Plazos

Además de comprender la frecuencia con que los respectivos equipos de los encuestados no logran cumplir con las fechas límite de entrega, y de comprender sus razones, resulta interesante también aprovechar esta ocasión para abordar los resultados que generan estos problemas para los grupos de trabajo. A través de la siguiente figura, se pueden visualizar los

resultados obtenidos:

Figura 29

Resultados de la Pregunta Nro. 22 Impacto del incumplimiento de fechas de entrega



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Puede notarse con facilidad, que la mayoría de las respuestas se han concentrado en dos grandes grupos, donde la gente encuestada ha indicado que el impacto del incumplimiento de los plazos de entrega resulta en un incremento de costos para la compañía, así como también en una alta rotación del personal. En este caso un 51,5% de las respuestas ha indicado la primera opción, mientras que un 46,7% se ha inclinado por la segunda de ellas. Sin lugar a dudas, estos efectos pueden ser ejemplo para las empresas de reacciones a evitar en sus equipos, lo cual volverá para estas de gran utilidad a esta investigación.

7.1.23 - Precisión Del Proceso De Estimación

En cuanto a esta pregunta, la intención detrás de ella radica en poder cotejar la información de que tan preciso los encuestados definen su proceso de estimación contra como han definido ejecutarlo previamente. A continuación, se exponen sus resultados mediante una figura:

Figura 30

Resultados de la Pregunta Nro. 23 Precisión del proceso de estimación

¿Qué tan preciso considera que es su proceso de estimación actual?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Puede notarse que un 47,3% de los encuestados afirman tener un proceso de estimación preciso, mientras que sólo un 1,2% de ellos ha definido a su proceso como muy preciso. Asimismo, un 35,9% de ellos se ha mostrado de forma neutral en cuanto a esta pregunta y un 11,4 % ha sostenido tener procesos poco precisos.

7.1.24 - Frecuencia De Retroalimentación Y Mejoras

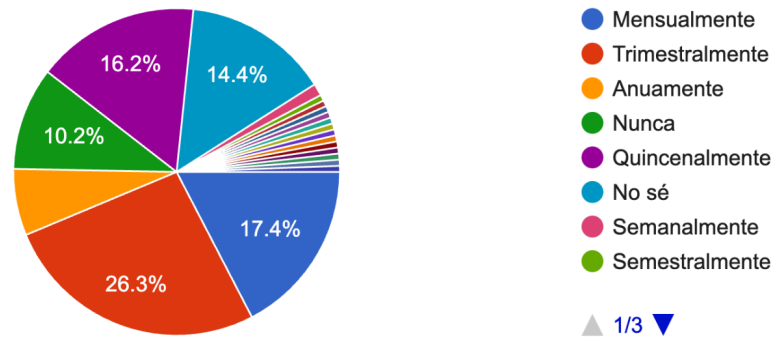
En último lugar, esta pregunta está orientada a resaltar la frecuencia con que los equipos revisan y ajustan su proceso de trabajo, es decir, su proceso de retroalimentación. A continuación pueden observarse los resultados obtenidos:

Figura 31

Resultados de la Pregunta Nro. 24 Frecuencia de retroalimentación y mejoras

¿Con qué frecuencia se revisa y ajusta el proceso de trabajo en su equipo?

167 respuestas



Nota. Reproducido de Google Formularios. Generado a través de las encuestas realizadas.

Como puede verse, esta respuesta ha sido la que más diversidad de resultados ha presentado. Predominan con un 26,3% los equipos que realizan procesos de retroalimentación de forma trimestral. En segundo lugar, pero muy cercano al tercero, puede verse que los equipos realizan sus revisiones de forma mensual (17,4%) y quincenal (16,2%). Es importante destacar que un 24,6% de los encuestados afirman que no realizan procesos de revisión y ajuste, o bien, no saben de su existencia, lo cual también puede resultar en objeto de análisis de esta investigación en función de los patrones que influyen en el incumplimiento de las fechas de entrega.

7.2 - Análisis De Los Resultados De Las Entrevistas

Habiendo expuesto apropiadamente todas las preguntas realizadas a los participantes de la encuesta y sus correspondientes resultados, se procede en esta sección a un análisis más profundo de los mismos a través de la utilización de distintas técnicas según sea conveniente para la

naturaleza de cada pregunta y sus tipos de resultados.

En primer lugar, se procede a la cruce de los datos obtenidos para la metodología utilizada para la estimación contra los resultados donde las frecuencias de incumplimiento de fechas de entrega son “A veces”, “Frecuentemente” y “Siempre”. Para este análisis han sido filtradas las respuestas no representativas de metodología de estimación, es decir, las cuales tienen una sola respuesta y su porcentaje es irrelevante. La tarea se ha llevado a cabo gracias a la utilización de una hoja de cálculo de Microsoft Excel y su herramienta de tablas dinámicas. A continuación, se expone una tabla con los resultados:

Tabla 2

Análisis de incumplimiento de fechas de entrega en función de la metodología utilizada para la estimación

Metodología ejecutada para la estimación	Frecuencia de incumplimiento de fechas de entrega			
	A veces	Frecuentemente	Siempre	Total
En tiempo, asociando cada tarea a una cantidad de tiempo.	23	8	3	34
Story Points, asociando esos puntos a una determinada cantidad de tiempo.	8	4		12
Story Points, entendiendo el peso/esfuerzo involucrado para llevar a cabo la tarea.	21	7	1	29

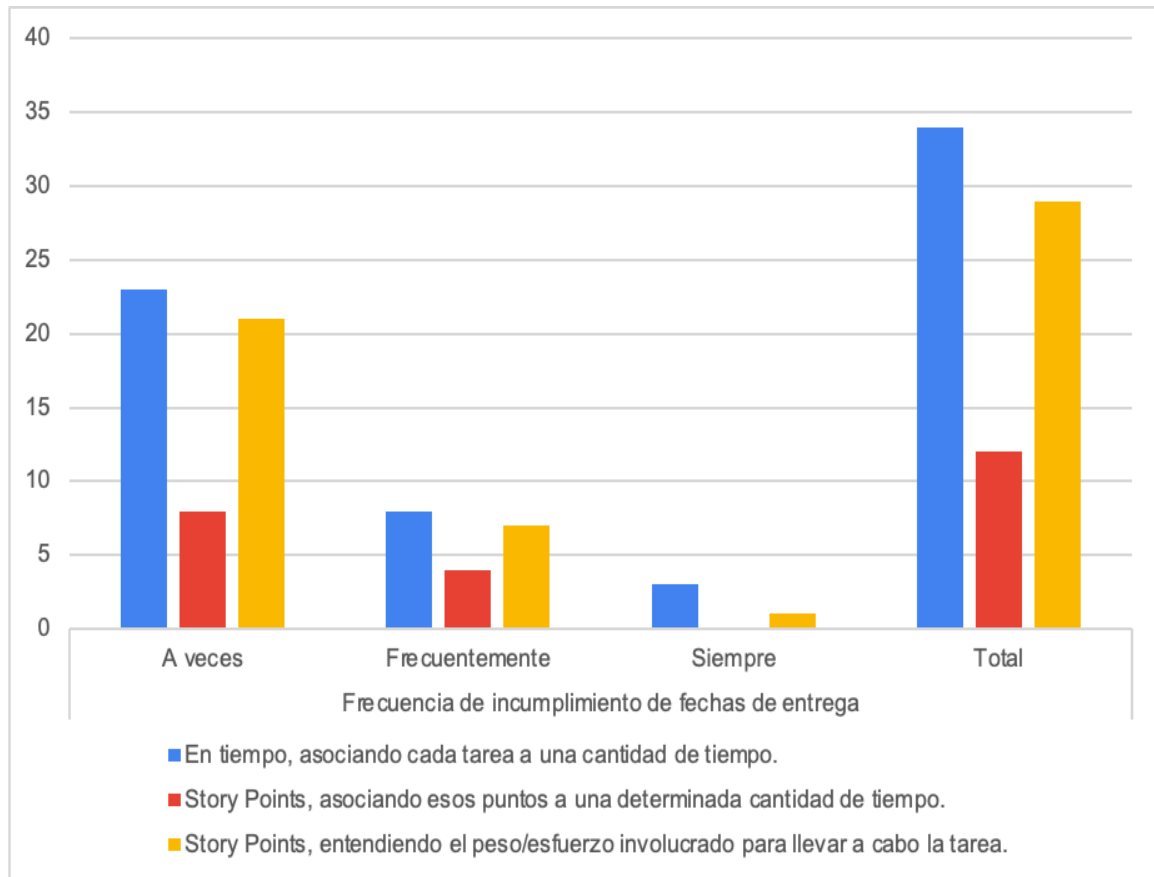
Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

Tal como puede apreciarse en la tabla 2, en todas las frecuencias de incumplimiento de fechas de entrega los niveles más altos se han registrado cuando la metodología de estimación

utilizada es “en tiempo, asociando cada tarea a una cantidad de tiempo”. Es así que, este grupo ejecutando dicha metodología, representa un total de 45,33% sobre los resultados analizados. Asimismo, es importante no ignorar que un 38,67% de los resultados corresponden a encuestados que han respondido incumplir a veces, frecuentemente o siempre con los plazos de entrega utilizando la metodología de estimación “Story Points, entendiendo el peso/esfuerzo involucrado para llevar a cabo la tarea”. Este hecho podría contradecir lo inicialmente planteado por esta investigación, que fuera que los equipos incumplen en los plazos de entrega por utilizar una metodología de estimación incorrecta. En este caso, y con este análisis, no se puede afirmar aún que esto sea cierto, por lo cual se procederá posteriormente a ir más profundo en estos resultados, cruzando estos datos con otras variables para comprender si más allá de estimar correctamente, los equipos no están cumpliendo con algún parámetro esperado. Para concluir con este análisis, se presenta a continuación una figura con una representación gráfica de los resultados:

Figura 32

Representación gráfica del análisis de incumplimiento de fechas de entrega en función de la metodología utilizada para la estimación



Nota. Elaboración propia con la utilización de Microsoft Excel, utilizando los datos provenientes de las encuestas realizadas.

Como resultado del anterior análisis donde se han cotejado las variables de metodología empleada para la estimación de trabajo y la frecuencia de incumplimiento de fechas de entrega, se ha podido ver la necesidad de explotar estos datos en niveles más complejos para poder arribar a mejores conclusiones. En este caso, se ha procedido a realizar una tabla dinámica en Microsoft Excel y se ha utilizado como fila la metodología de estimación empleada. En cuanto a las columnas, las cuales explotan esta información en más subniveles, se ha utilizado la metodología ágil utilizada por el equipo como primer nivel de agrupamiento, y el tamaño del equipo como

segundo nivel. A su vez, el dato final utilizado para calcular los valores que se expondrán a continuación ha sido la frecuencia de incumplimiento de plazos. Véase a continuación una tabla para la apropiada exposición de los datos calculados:

Tabla 3

Análisis de incumplimiento de fechas de entrega en función de la metodología utilizada para la estimación, la metodología de trabajo del equipo y su tamaño.

Metodología de Estimación	Metodología Ágil																	
	Cascada	Kanban				Lean	Mixta				Ninguna		No lo sé		Scrum			
	Tamaño del equipo																	
	1 - 5	1 - 5	11 - 20	6 - 10	6 - 10	1 - 5	11 - 20	6 - 10	Más de 20	1 - 5	6 - 10	1 - 5	6 - 10	1 - 5	11 - 20	6 - 10	Más de 20	
En tiempo, asociando cada tarea a una cantidad de tiempo.	1	2		3	1	4	1	1	1	2	1	1	2	4	1	7		
Story Points, asociando esos puntos a una determinada cantidad de tiempo.		1	1			3	2								1	3	1	
Story Points, entendiendo el peso/esfuerzo involucrado para llevar a cabo la tarea.						1		2		2				3	7	11	2	

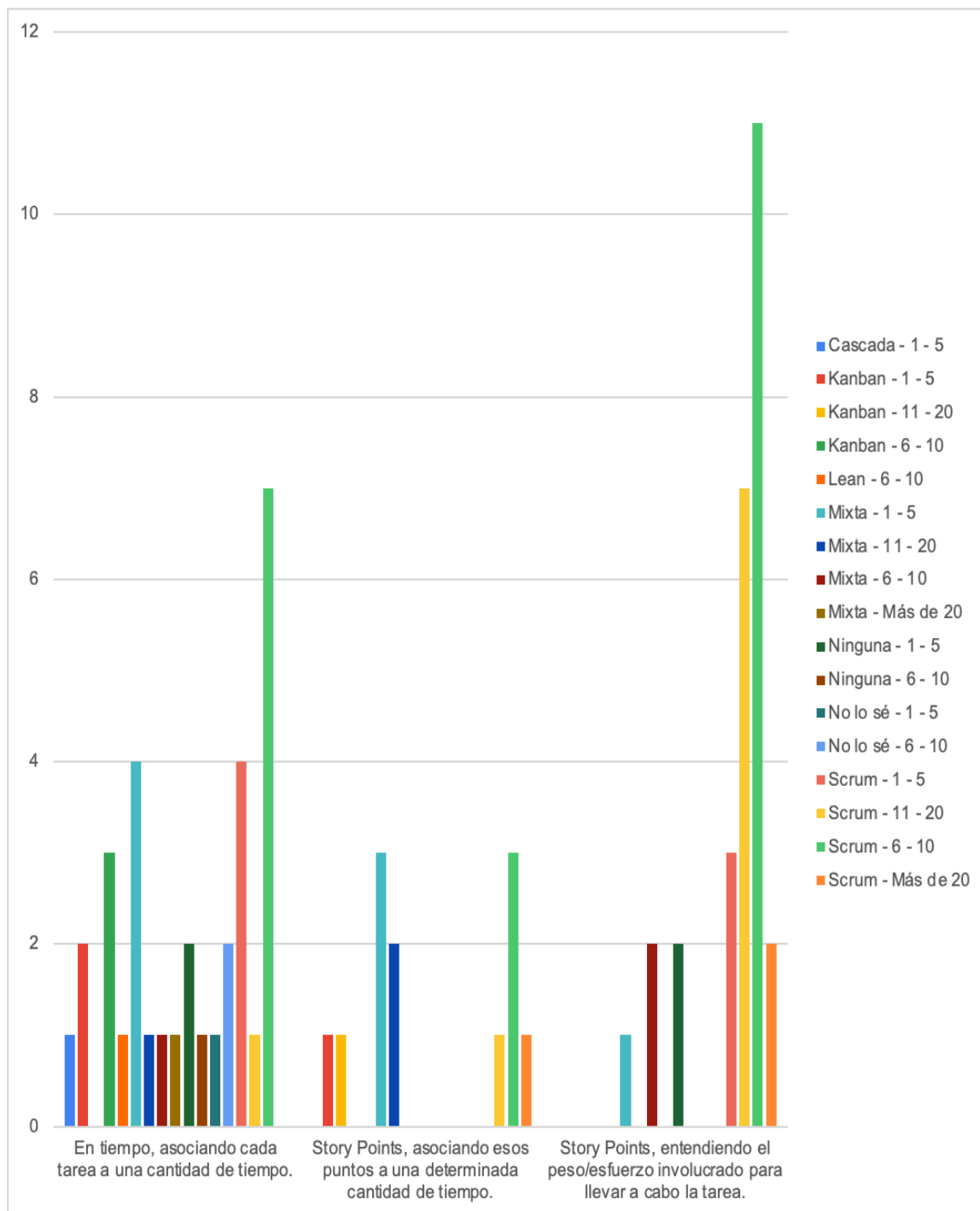
Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

Tal como puede observarse en los resultados obtenidos gracias a la tabla, los mayores problemas radican en los equipos de Scrum que tienen de entre 6 a 10 integrantes, o bien, entre 11 y 20 integrantes. Esto, contrario a lo analizado en la tabla 2 donde podía presumirse un problema de la metodología, proporciona un mayor entendimiento del problema. En este caso, puede observarse que los equipos correspondientes a los participantes de la encuesta que más incumplen con los plazos de entrega son equipos Scrum, realizando sus estimaciones en Story Points y comprendiendo el peso o esfuerzo involucrado para llevar a cabo la tarea que tienen una

cantidad de integrantes de entre 6 y 10. Esto podría suponer un problema ya que para su óptimo funcionamiento un equipo Scrum debería de tener idealmente siete miembros, pudiendo adicionar o restar dos de ellos (Sutherland, 2014). Asimismo, los equipos de Scrum correspondientes a los encuestados, que también trabajan estimando en Story Points y comprendiendo el peso o esfuerzo para llevar a cabo la tarea pero tienen entre 11 y 20 miembros, también han demostrado según los resultados de la tabla obtenidos gracias a la encuesta que tienen un alto índice de incumplimiento de entregas, lo que coincide una vez más con lo expuesto por Sutherland (2014) acerca de el tamaño ideal de un equipo Scrum. Se adiciona a continuación una figura para facilitar la comprensión gráfica de lo expuesto previamente:

Figura 33

Representación gráfica del análisis de incumplimiento de fechas de entrega en función de la metodología utilizada para la estimación, la metodología de trabajo del equipo y su tamaño.



Nota. Elaboración propia con la utilización de Microsoft Excel, utilizando los datos provenientes de las encuestas realizadas.

Con el fin de abordar esta problemática desde distintos puntos, se ha procedido también a cruzar los resultados obtenidos para las variables metodología de trabajo utilizada, tipo de proyecto y si el equipo ha adaptado la metodología, es decir, le ha aplicado modificaciones. Para ello, se ha confeccionado una nueva tabla dinámica con la ayuda de Microsoft Excel, donde pueden verse las filas agrupadas en un primer nivel según la metodología de trabajo utilizada y el tipo de proyecto, en un segundo nivel. Además, se ha utilizado la adaptación de la metodología como columna para diferenciar entre los equipos que hubieran acondicionado la metodología y los que no. A continuación se expone la tabla con los resultados obtenidos:

Tabla 4

Análisis de incumplimiento de fechas de entrega en función de la metodología de trabajo utilizada, el tipo de proyecto y la adaptación de la metodología

Metodología / Tipo de Proyecto	Adapto metodología			Grand Total
	No	No sé	Sí	
Cascada			1	1
Evolutivo			1	1
Kanban		1	6	7
Evolutivo		1	5	6
Innovación			1	1
Lean			1	1
Innovación			1	1
Mixta		1	14	15
Cumplimiento			1	1
Evolutivo			6	6
Innovación			5	5
Soporte		1	2	3
Ninguna	3	1	1	5
Desarrollo			1	1
Evolutivo	2			2
No estoy seguro		1		1
Soporte	1			1
No lo sé		1	2	3
Evolutivo			1	1
Múltiple		1		1
Soporte			1	1
Scrum	3		38	41
Desarrollo			1	1
Evolutivo	1		18	19
Gestión de Proyectos			1	1
Innovación	1		8	9
Investigación	1		2	3
No estoy seguro			4	4
Soporte			4	4
Grand Total	6	4	63	73

Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

De los resultados obtenidos gracias a esta tabla y el cruce de información que la misma realiza, es posible notar que el 56,16% de estos valores corresponden a equipos utilizando la metodología Scrum. En este caso puede observarse con claridad que el gran foco, donde 38 participantes en total han brindado esta información, está en los equipos que han aplicado modificaciones a la metodología en cuestión. A su vez, los más afectados son los equipos Scrum en proyectos evolutivos y de innovación. Por otro lado, se puede destacar que los equipos utilizando metodologías de desarrollo mixtas y Kanban, también representan sobre el total

valores dignos de estudio. En cuanto a la metodología mixta, se puede apreciar que casi todos los casos donde se registran problemas con incumplimientos de entregas es cuando han modificado la metodología, lo cual en este caso no supone más que una redundancia porque la metodología mixta no es más que la combinación de metodologías y por ello, la modificación de las mismas. Por último, si bien en mucho menor escala con un 9,59% se encuentran los resultados obtenidos para la metodología Kanban. Puede apreciarse en ellos un cierto grado de incumplimiento habiendo modificado la metodología y utilizando Kanban para desarrollos evolutivos y de innovación.

Siguiendo en línea con el análisis en función de las variables relacionadas a la metodología, es oportuno también cotejar los datos obtenidos para la metodología de trabajo utilizada, el tamaño del equipo y si han recibido capacitación formal en cuanto a la metodología utilizada. Vale aclarar que estas variables serán analizadas una vez más en función de la frecuencia de incumplimiento de fechas de entrega por parte de los equipos de los entrevistados. A continuación, se expone una tabla dinámica realizada con Microsoft Excel donde puede observarse el resultado de cruzar esta información:

Tabla 5

Análisis de incumplimiento de fechas de entrega en función de la metodología de trabajo utilizada, el tamaño del equipo y la recepción de capacitación formal

Metodología/Capacitación/Tamaño de equipo	Frecuencia de incumplimiento
Cascada	1
Sí	1
1 - 5	1
Kanban	7
No	5
1 - 5	2
6 - 10	3
Sí	2
1 - 5	1
11 - 20	1
Lean	1
Sí	1
6 - 10	1
Mixta	15
No	7
1 - 5	5
6 - 10	1
Más de 20	1
Sí	8
1 - 5	3
11 - 20	3
6 - 10	2
Ninguna	5
No	4
1 - 5	3
6 - 10	1
Sí	1
1 - 5	1
No lo sé	3
No	2
1 - 5	1
6 - 10	1
Sí	1
6 - 10	1
Scrum	41
No	11
1 - 5	2
11 - 20	2
6 - 10	7
Sí	30
1 - 5	5
11 - 20	7
6 - 10	15
Más de 20	3
Grand Total	73

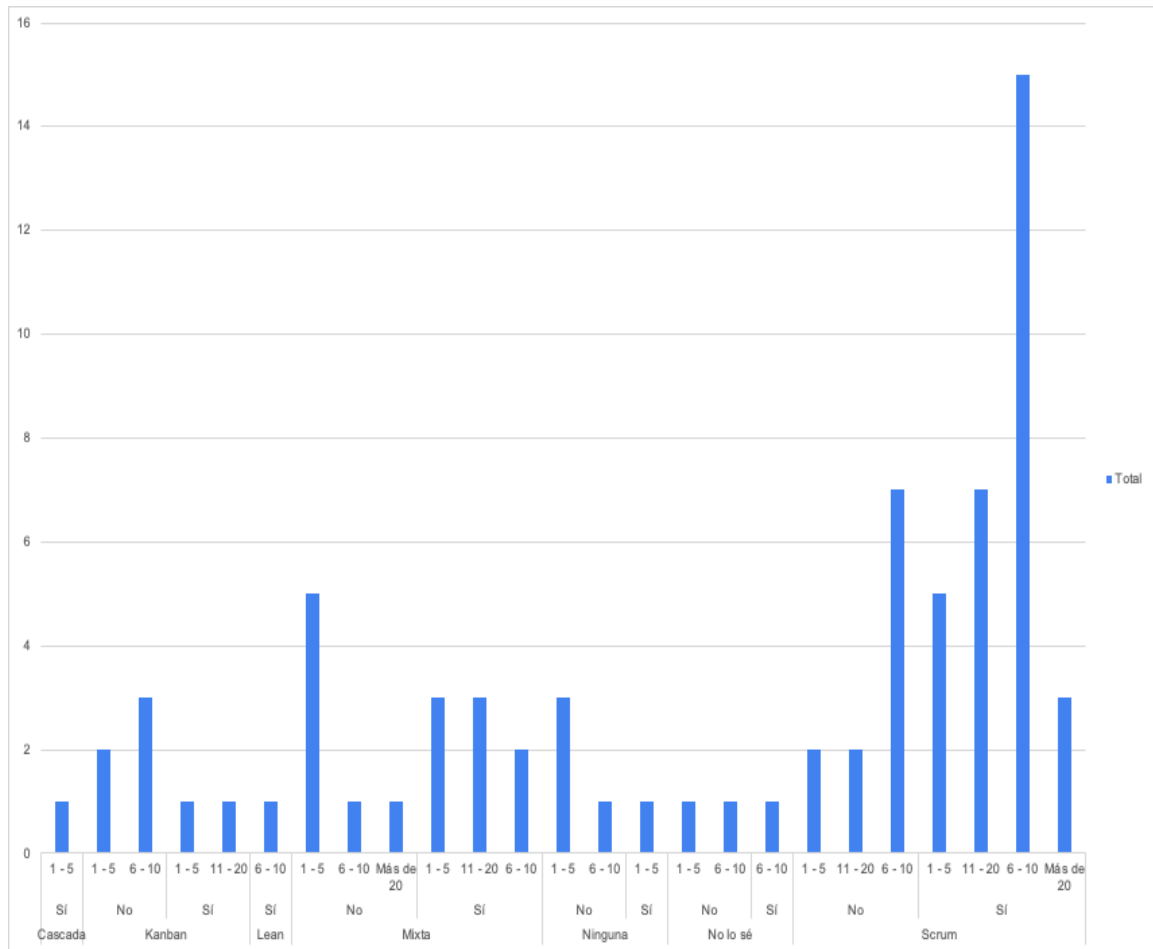
Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas

realizadas.

Se desprende del análisis de esta tabla, que una vez más los equipos de Scrum de entre 6 y 10 personas, incluso habiendo recibido capacitación formal, son los equipos con más resultados en incumplimientos de plazos de entrega. Sin mucho más que contrastar contra el análisis obtenido en la tabla 3, estos resultados refuerzan la teoría de que los equipos de Scrum que exceden el tamaño recomendado, según Sutherland (2014) idealmente de siete personas pudiendo reducir o incrementar dos de ellas, no obtienen óptimos resultados a la hora de realizar entregas. En este caso, este grupo representa un 30% de los entrevistados. a continuación, se expone una figura para facilitar la comprensión gráfica de estos resultados:

Figura 34

Representación gráfica del análisis de incumplimiento de fechas de entrega en función de la metodología de trabajo, el tamaño del equipo y la recepción de capacitación.



Nota. Elaboración propia con la utilización de Microsoft Excel, utilizando los datos provenientes de las encuestas realizadas.

Otra forma valiosa de analizar los resultados obtenidos es observar la frecuencia de incumplimiento de fechas límite en función de la precisión de las estimaciones en función de haber recibido capacitación formal en la metodología y, por supuesto, la metodología utilizada. De esta forma, se intenta comprender si la efectiva capacitación en la metodología vuelve más precisos a los equipos a la hora de estimar y en consecuencia, les permite cumplir con las fechas establecidas para las entregas. A continuación, se puede observar una tabla dinámica realizada

con Microsoft Excel nuevamente en donde se ha cruzado esta información:

Tabla 6

Análisis de incumplimiento de fechas de entrega en función de la metodología de trabajo utilizada, la capacitación formal y la precisión de las estimaciones.

Precisión/Capacitación/Metodología 	Incumplimiento Frecuente
☐ Nada preciso	2
☐ No	1
Mixta	1
☐ Sí	1
Scrum	1
☐ Neutral	37
☐ No	15
Kanban	3
Mixta	2
Ninguna	3
No lo sé	2
Scrum	5
☐ Sí	22
Cascada	1
Kanban	1
Lean	1
Mixta	3
No lo sé	1
Scrum	15
☐ Poco preciso	15
☐ No	4
Kanban	1
Mixta	2
Scrum	1
☐ Sí	11
Kanban	1
Mixta	3
Scrum	7
Grand Total	54

Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

Al analizar la información obtenida en la tabla previamente expuesta, los resultados indican que casi un 28% de los encuestados (el mayor porcentaje de un grupo registrado en el análisis de esta tabla) que trabajan en equipos Scrum, han recibido capacitación formal en la metodología y además califican sus estimaciones de forma neutral incumplen con cierta frecuencia los plazos de entrega. A su vez, es Scrum quien también lidera los resultados en la categoría de precisión de la estimación calificada como “poco precisa” con un 12,96%. Si bien esta información por sí sola, no permite realizar ninguna afirmación fehaciente, permite al menos validar una vez más que no es suficiente recibir capacitación y tener un proceso de estimación preciso para que Scrum funcione bien, sino que hay factores como el tamaño del equipo que son muy influyentes en este resultado.

Luego de haber analizado la frecuencia de incumplimiento de los plazos de entrega en función de las variables metodológicas, o bien, las variables correspondientes a las características del equipo y haber podido obtener determinadas conclusiones al respecto, es momento de analizar otro grupo de variables que también afectan el apropiado cumplimiento de las fechas de entrega. En este caso, se toman los factores más relevantes en el incumplimiento mencionados por los encuestados filtrando los resultados menos relevantes y se analiza esta información en función de la frecuencia de incumplimiento. En particular, la frecuencia de incumplimiento de plazos se analizará en función de los siguientes factores influyentes en el incumplimiento según los encuestados: estimaciones incorrectas, cambios en los requisitos del proyecto, problemas de comunicación, falta de involucramiento de los stakeholders y falta de recursos.

En primera instancia, se ha evaluado la frecuencia de incumplimiento en función del

factor estimaciones incorrectas. Puede verse a continuación la tabla dinámica correspondiente donde se ha realizado el análisis de estos datos:

Tabla 7

Análisis de incumplimiento de fechas de entrega en función de las estimaciones incorrectas.

Frecuencia de incumplimiento	Estimaciones incorrectas	Total
A veces	35	54
Frecuentemente	12	19
No hay		1
Nunca	1	5
Raramente	41	84
Siempre	1	4
Total	77	167

Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

A partir de esta tabla, puede notarse que las frecuencias de interés, es decir “a veces”, “frecuentemente” y “siempre” contienen una gran cantidad de respuestas en función del total de la frecuencia. En el caso de los equipos que reportaron no poder cumplir con los plazos de entrega siempre, el factor de estimaciones incorrectas representa un 25% de ellas. Asimismo, de entre los encuestados que reportaron no cumplir con las fechas límite de forma frecuente, un 63,16% de ellos ha indicado las estimaciones incorrectas como uno de los factores. En último lugar, de las personas que señalan que sus grupos de trabajo incumplen a veces con los plazos de entrega acordados, un 64,81% de ellos sostienen que uno de los motivos son las estimaciones

incorrectas.

En segundo lugar, se ha procedido a evaluar la frecuencia de incumplimiento de los deadlines en función del factor reportado por los entrevistados y calificado como cambios en los requisitos del proyecto. Como en el resto de los casos, puede verse a continuación una tabla dinámica donde se han obtenido estos resultados:

Tabla 8

Análisis de incumplimiento de fechas de entrega en función de los cambios en los requisitos del proyecto.

Frecuencia de incumplimiento ▼	Cambios en los requisitos del proyecto		Total
A veces	27		54
Frecuentemente	13		19
No hay			1
Nunca	3		5
Raramente	53		84
Siempre	4		4
Total	67	100	167

Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

Nótese que en este caso, el 100% de los entrevistados que reportaron que sus equipos incumplen siempre con las fechas de entrega han afirmado que un factor son los cambios en los requisitos del proyecto. A su vez, el 68,42% de los encuestados que han sostenido incumplir con los plazos de forma frecuente indican que los cambios en los requisitos son un factor influyente. Por último, el 50% de los participantes de la encuesta que indicaron incumplir a veces con las

fechas límite también han afirmado que los cambios en los requisitos del proyecto es un factor influyente.

De igual modo en que se ha evaluado la frecuencia de incumplimiento en función de las estimaciones incorrectas y de los cambios en los requisitos del proyecto, se procede a evaluar la influencia de los problemas de comunicación en el fracaso al cumplir con los plazos de entrega. Véase a continuación, los resultados de haber trabajado con dichas variables en una tabla dinámica:

Tabla 9

Análisis de incumplimiento de fechas de entrega en función de los problemas de comunicación.

Frecuencia de incumplimiento		
Frecuencia de incumplimiento	Problemas de comunicación	Total
A veces	27	54
Frecuentemente	10	19
No hay		1
Nunca	2	5
Raramente	33	84
Siempre	3	4
Total	75	167

Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

A partir de los resultados obtenidos, puede notarse que de los participantes que indican incumplir siempre con las fechas de entrega, un 75% de ellos coincide en que los problemas de comunicación son un factor influyente en esta cuestión. Los encuestados que afirman incumplir frecuentemente o a veces, en ambos casos han rondado el 50% de ellos indicando la mala comunicación como un problema influyente, en particular con un 52,63% y 50% respectivamente.

Habiendo analizado los resultados en función de los tres primeros factores influyentes en el incumplimiento de entregas mencionados, se continua ahora con la evaluación de estos resultados en función del factor influyente “falta de involucramiento de los stakeholders”. Antes de pasar a analizar sus resultados, se expone la tabla dinámica correspondiente al tratamiento de sus datos:

Tabla 10

Análisis de incumplimiento de fechas de entrega en función de la falta de involucramiento de los stakeholders.

Frecuencia de incumplimiento ▼	Falta de involucramiento de los stakeholders	Total
A veces	18	54
Frecuentemente	9	19
No hay		1
Nunca	1	5
Raramente	28	84
Siempre	2	4
Total	58	167

Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

Gracias a este análisis, se puede observar que los porcentajes de entrevistados que afirman la falta de involucramiento de los stakeholders no es tan alto como el porcentaje para los factores previos. En este caso, de los equipos que incumplen siempre con las fechas de entrega, un 50% de ellos sostienen la falta de involucramiento de los stakeholders como un factor influyente. A su vez, el 47,37% de los que han indicado incumplir con las entregas de forma frecuente también mencionan este factor como de influencia. En menor escala, con un 33,33%, los equipos que incumplen algunas veces con las fechas de entrega sostienen el mencionado factor como influyente en esta problemática.

En cuanto al último de los factores influyentes en el incumplimiento de entregas mencionados, resta abordar la falta de recursos. Para ello, se ha realizado una vez más una tabla dinámica para poder analizar sus valores, la cual se expone a continuación:

Tabla 11

Análisis de incumplimiento de fechas de entrega en función de la falta de recursos.

Frecuencia de incumplimiento ▼	Falta de recursos	Total
A veces	19	54
Frecuentemente	7	19
No hay		1
Nunca	1	5
Raramente	22	84
Siempre	1	4
Total	50	167

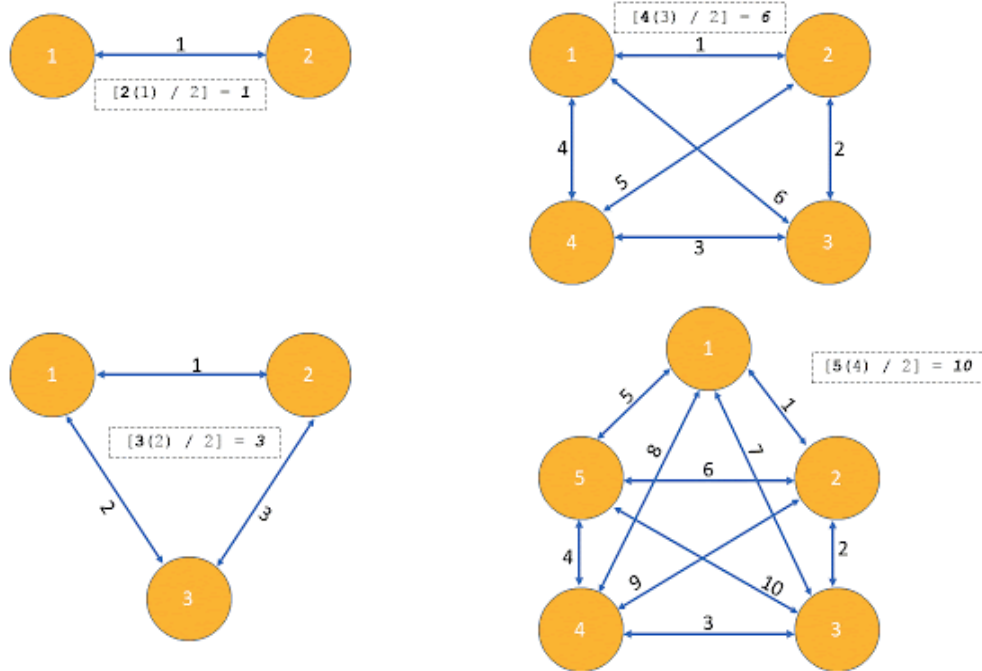
Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

De igual forma en que se han analizado los factores previos, puede observarse en este caso un descenso del porcentaje de afirmación de este factor por parte de los entrevistados, en comparación con los anteriores. Sólo el 25% de los entrevistados que han afirmado incumplir siempre con las fechas de entrega sostiene la falta de recursos como un factor influyente. En cuanto a los que han indicado incumplir de manera frecuente o algunas veces, estos demuestran porcentajes de un 36,84% y 35,19% respectivamente.

A partir del análisis de los resultados para estos factores, pueden realizarse algunas afirmaciones de valor. Los factores más influyentes en los equipos que han indicado fallar siempre en sus fechas de entrega son los cambios en los requisitos del proyecto y problemas de comunicación. Es interesante ver entre ellos a los problemas de comunicación, reforzando la teoría de que el número de canales de comunicación incrementa dramáticamente con el número de miembros de los equipos, siendo algo que no puede ser manejado de forma óptima (Sutherland, 2014). Como se indica en el blog *Tales of Agile Adoption*, los equipos pequeños pueden gestionar mejor la cantidad de canales de comunicación cuando se limitan a un número razonable (*¿Por qué los equipos de Scrum tienen un tamaño limitado?*, 2017). Esto también se ilustra en la imagen expuesta a continuación (ver Figura 35).

Figura 35

Canales de comunicación: equipos de 2 a 5 integrantes.



Canales de comunicación: equipos de 2 a 5 integrantes

Nota. Reproducido de *Tales of Agile Adoption*,

<https://tales-of-agile-adoption.blogspot.com/2017/08/por-que-los-equipos-de-scrum-tiene-n-un.html>

Por otro lado, en cuanto a las personas encuestadas que indicaron que sus equipos incumplen en las fechas de entrega de manera frecuente, estas han destacado en mayor medida los cambios en los requisitos del proyecto con un 68,42% y estimaciones incorrectas con un 63,16%. Cabe destacar que, si bien en menor porcentaje, también han indicado con un 52,63% los problemas de comunicación, reforzando así lo concluido gracias a los resultados de las respuestas de las personas cuyos equipos fallan siempre en los plazos establecidos. Asimismo, el último grupo de interés que es el que indicó pertenecer a equipos de trabajo que fracasan algunas veces al momento de cumplir con las fechas de entrega ha reportado a través de sus respuestas a

las estimaciones como el factor más influyente en esta problemática.

Habiendo efectuado hasta el momento un análisis de los resultados a través de las variables correspondientes a la metodología de trabajo, características del proyecto y equipo, y factores influyentes en el incumplimiento de las fechas de entrega, se puede avanzar sobre un análisis en función de cómo las variables referidas a la comunicación impactan en el funcionamiento de los equipos. En primera instancia, se realiza una evaluación de la relación entre la frecuencia de las reuniones de seguimiento y la frecuencia de incumplimiento de fechas a la hora de realizar entregas. Se expone a continuación una tabla dinámica con los resultados generados para el análisis de estas variables:

Tabla 12

Frecuencia de incumplimiento de fechas de entrega en función de la frecuencia de sincronización del equipo, la metodología de trabajo y su tamaño.

Tamaño del equipo/Frecuencia Sincronización	Incumplimiento de entregas
Cascada	1
1 - 5	1
Semanalmente	1
Kanban	7
1 - 5	3
Diariamente	2
Mensualmente	1
11 - 20	1
Diariamente	1
6 - 10	3
Diariamente	2
Semanalmente	1
Lean	1
6 - 10	1
Semanalmente	1
Mixta	15
1 - 5	8
Diariamente	2
Mensualmente	1
Quincenalmente	1
Semanalmente	4
11 - 20	3
Diariamente	2
Semanalmente	1
6 - 10	3
Diariamente	2
Semanalmente	1
Más de 20	1
Semanalmente	1
Ninguna	5
1 - 5	4
Diariamente	3
Semanalmente	1
6 - 10	1
Quincenalmente	1
No lo sé	3
1 - 5	1
Semanalmente	1
6 - 10	2
Semanalmente	2
Scrum	40
1 - 5	7
Diariamente	4
Mensualmente	1
Semanalmente	2
11 - 20	8
Diariamente	7
Semanalmente	1
6 - 10	22
Diariamente	17
Quincenalmente	1
Semanalmente	4
Más de 20	3
Diariamente	1
Semanalmente	2
Grand Total	72

Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

Una vez más, al igual que en el análisis de variables previas, puede notarse que un 23,61% de los resultados corresponden a equipos Scrum de entre 6 y 10 miembros realizando reuniones de seguimiento de forma diaria, lo cual es lo indicado para dicha metodología (Sutherland, 2014), corresponden a resultados donde se fracasa en el cumplimiento de fechas de entrega desde algunas veces hasta siempre. De igual modo, un 9,72% de los resultados también corresponden a equipos de Scrum de entre 11 y 20 miembros. Esto refuerza una vez más el hallazgo realizado al analizar las tablas previas acerca de cómo el tamaño de los equipos afecta negativamente a los equipos Scrum en cuanto al cumplimiento de las fechas de entrega.

Finalmente, a fines de concluir el análisis de los resultados, se analiza la última variable pendiente con elevado valor al ser comparada contra la frecuencia de incumplimiento de fechas límite. En este caso se trata de que tan a menudo el equipo realiza revisiones de su proceso actual y ejecuta ajustes para mejorarlo. De este modo, se podrá comprender cómo la retroalimentación de los equipos tiene o no un rol importante en cuanto a lograr cumplir con los plazos de entrega. Se expone a continuación la correspondiente tabla dinámica a través de la cual se han analizado los resultados:

Tabla 13

Frecuencia de incumplimiento de fechas de entrega en función de la frecuencia de retroalimentación.

Tamaño equipo/Frecuencia Retroalimentación		Incumplimiento de fechas
1 - 5		25
al finalizar el proyecto		1
Anualmente		2
En la empresa se viven cuestionando los metodos de trabajo pero todo queda en catarsis nomas		1
No sé		6
Nunca		4
Quincenalmente		3
Todos los días, es una migración mi proyecto y está caótico.		1
Trimestralmente		7
11 - 20		13
Anualmente		1
Cada 6 meses		1
Mensualmente		3
No sé		2
Quincenalmente		4
Trimestralmente		2
6 - 10		34
A demanda, segun las necesidades de los proyectos		1
Anualmente		2
Mensualmente		8
No sé		2
Nunca		5
Quincenalmente		5
Semanalmente		1
Siempre que sea necesario		1
Trimestralmente		9
Más de 20		5
Mensualmente		1
Quincenalmente		1
Trimestralmente		3
Grand Total		77

Nota. Elaboración propia, utilizando los datos normalizados provenientes de las encuestas realizadas.

Como puede observarse en la tabla 13, el mayor grado de incumplimiento de fechas de entrega se registra en equipos de entre 6 y 10 personas nuevamente con un 11,69% para los que realizan la revisión de su proceso trimestralmente y un 10,39% para los que la realizan de forma mensual. A su vez, los equipos de entre 1 y 5 personas que realizan su revisión de forma trimestral, también conforman un porcentaje significativo sobre este total, con un total del 9,09%. Esto podría sugerir que la realización tan espaciada de los procesos de revisión y ajuste de los equipos este relacionada con el incumplimiento de las fechas de entrega, aunque será necesaria

una investigación más profunda para poder sustentar dicha afirmación.

8 - Conclusiones

El presente trabajo ha revelado que el incumplimiento de los plazos de entrega en los equipos de desarrollo de software en Argentina se debe a una combinación de factores críticos, destacándose especialmente el tamaño de los equipos, la frecuencia de las revisiones de procesos, los cambios en los requisitos del proyecto y los problemas de comunicación. Los equipos de entre 6 y 10 miembros han mostrado un alto índice de fallas, superando el límite de 9 miembros propuesto por Sutherland, lo que confirma que el tamaño excesivo de los equipos impacta negativamente en su eficiencia. Este hallazgo es particularmente importante, ya que los equipos de Scrum, que siguieron el proceso de reuniones diarias tal como lo establece la metodología, también registraron un porcentaje significativo de incumplimiento. Los equipos de 11 a 20 miembros no estuvieron exentos de estos problemas, lo que refuerza que el aumento en el número de integrantes dificulta la coordinación y la comunicación, afectando directamente el cumplimiento de los plazos.

Otro factor clave es la frecuencia de las revisiones de procesos. Los equipos que realizan revisiones trimestrales, tanto de 6 a 10 miembros como aquellos de 1 a 5, registraron los mayores índices de fallas, con un 11,69% y un 9,09%, respectivamente. Esto sugiere que los ciclos de revisión espaciados podrían estar relacionados con el incumplimiento de los plazos, aunque será necesario investigar más a fondo para confirmar esta hipótesis. La observación de que los equipos más pequeños también enfrentan dificultades en este sentido subraya que no solo el tamaño, sino la estructura y la regularidad de los procesos de ajuste son factores críticos para el éxito en la gestión de proyectos de software.

El análisis de las metodologías actuales también reveló que los equipos Scrum que realizan modificaciones a la metodología, especialmente en proyectos evolutivos o de innovación, presentan elevados índices de fallas. Este resultado es coherente con los equipos que aplicaron metodologías mixtas, las cuales, por su propia naturaleza, introducen una complejidad adicional que tiende a generar mayores niveles de incumplimiento. Asimismo, aunque Kanban mostró una menor frecuencia de fallas (9,59%), los equipos que utilizaron esta metodología para proyectos evolutivos también experimentaron problemas, sobre todo cuando la metodología fue modificada para adaptarse a las particularidades del proyecto.

Un aspecto relevante de la investigación es que, a pesar de haber recibido capacitación adecuada, muchos equipos continúan fallando en el cumplimiento de los plazos. Scrum, a pesar de ser una metodología ampliamente utilizada y de contar con procesos de estimación precisos como el uso de Story Points, sigue liderando en términos de frecuencia de fallas. Esto indica que la capacitación, aunque necesaria, no garantiza por sí sola el éxito de los equipos si no se acompaña de una implementación adecuada y una adaptación efectiva de la metodología al contexto específico del proyecto. Es evidente que los problemas subyacentes, como los cambios constantes en los requisitos y las dificultades en la comunicación, continúan siendo determinantes.

En este sentido, los problemas de comunicación emergen como uno de los factores más críticos. El análisis reveló que el 68,42% de los equipos que fallan con frecuencia señalan los cambios en los requisitos del proyecto como la principal causa de retrasos, mientras que el 63,16% destacó las estimaciones incorrectas. Además, un 52,63% de estos equipos identificó los problemas de comunicación como un factor clave, lo que confirma que la teoría de Sutherland

sobre la multiplicación de los canales de comunicación en equipos grandes se sostiene. La cantidad creciente de canales dificulta una gestión eficiente de la información y afecta negativamente la capacidad de los equipos para cumplir con los plazos establecidos. Este problema es especialmente relevante en equipos grandes, donde las barreras comunicativas se intensifican y generan mayores dificultades en la coordinación y la alineación de objetivos.

En equipos que fallan de manera menos frecuente, el factor más influyente señalado fue la estimación incorrecta del tiempo y el esfuerzo requerido. Esto sugiere que, aunque no siempre sea el principal causante de los incumplimientos, los errores en la estimación continúan representando un desafío significativo para los equipos de desarrollo de software. La falta de precisión en la estimación afecta directamente la planificación y ejecución de los proyectos, lo que a su vez compromete los plazos acordados.

Sin embargo, los equipos que han demostrado un desempeño exitoso en cuanto al cumplimiento de plazos siguen varias buenas prácticas que pueden servir como guía para mejorar los procesos. Estos equipos involucran a todo su personal en el proceso de estimación, lo que permite una visión más integral y precisa de los tiempos y recursos necesarios para cumplir con los objetivos. La inclusión de todos los miembros en la estimación mejora no sólo la precisión, sino también el compromiso con los plazos acordados. Además, estos equipos realizan revisiones periódicas de sus procesos de trabajo, ya sea quincenal o mensualmente, lo que les permite realizar ajustes rápidos y efectivos en el flujo de trabajo para mantenerse alineados con los objetivos del proyecto. La regularidad en las revisiones y ajustes es un factor clave para el éxito de los equipos, ya que les permite detectar problemas antes de que se conviertan en impedimentos mayores.

Estos equipos también hacen un uso eficiente de herramientas avanzadas de gestión, que permiten un seguimiento detallado del progreso y la gestión de recursos, además de facilitar la comunicación y la coordinación dentro del equipo. El uso adecuado de estas herramientas mejora la capacidad de los equipos para supervisar las tareas y ajustar las estimaciones cuando es necesario. También se destaca la importancia de las reuniones de seguimiento regulares, que permiten abordar problemas a medida que surgen, manteniendo el proyecto dentro de los plazos establecidos. Estas reuniones, cuando se llevan a cabo de forma diaria o quincenal, son una práctica eficaz para mantener el equipo enfocado en las metas y responder rápidamente a cualquier cambio en las condiciones del proyecto.

Otro aspecto que diferencia a los equipos que cumplen con los plazos es su capacidad de coordinación con otros equipos. Los equipos que logran cumplir consistentemente con los plazos tienden a estar bien coordinados con otras áreas, asegurando que no haya demoras causadas por dependencias entre equipos. La buena comunicación y la colaboración fluida con otros equipos o departamentos han demostrado ser factores esenciales para el éxito en proyectos complejos donde varias unidades organizativas están involucradas.

Por tanto, la hipótesis de que el incumplimiento de los plazos de entrega se debe principalmente a una estimación incorrecta del trabajo, sumada al uso de metodologías inapropiadas o una falta de conocimiento sobre las metodologías adecuadas, se valida parcialmente. Si bien los errores en las estimaciones y el uso incorrecto de las metodologías son factores determinantes, la investigación ha revelado que otros aspectos, como los cambios en los requisitos, los problemas de comunicación y el tamaño excesivo de los equipos, también juegan un papel crucial en el incumplimiento de plazos. Estos factores adicionales complejizan aún más

el escenario, evidenciando que, para mejorar el rendimiento de los equipos de desarrollo de software, es necesario un enfoque más integral que aborde no solo las metodologías y las estimaciones, sino también los procesos de comunicación y la gestión del tamaño del equipo.

En conclusión, la investigación ha cumplido con los objetivos planteados, tanto generales como específicos. Ha permitido identificar los factores más influyentes en el incumplimiento de plazos, evaluar la efectividad de las metodologías ágiles y desarrollar recomendaciones basadas en los hallazgos. Sin embargo, los resultados también indican que algunos aspectos, como el tamaño de los equipos, la frecuencia de las revisiones y la calidad de la comunicación, requieren un análisis más profundo. Las metodologías ágiles, cuando no se implementan correctamente o se modifican en exceso, pierden efectividad, y los problemas en la comunicación y las estimaciones imprecisas siguen siendo grandes obstáculos para el cumplimiento de los plazos de entrega en los equipos de desarrollo de software.

9 - Líneas Futuras De Investigación

A partir de los hallazgos obtenidos, surgen varias áreas que requieren un análisis más profundo para entender los factores que contribuyen al incumplimiento de plazos en los equipos de desarrollo de software en Argentina. Es fundamental explorar en detalle los equipos de entre 6 y 10 miembros que fallan en el cumplimiento de plazos, centrándose en cuántos de estos superan el límite de 9 miembros propuesto por Sutherland (2014). Su investigación sugiere que los equipos ágiles más efectivos suelen tener un máximo de 9 integrantes, lo que podría indicar que aquellos que exceden este número enfrentan dificultades adicionales en coordinación y comunicación. Estudiar esta franja específica y su relación con el tamaño óptimo puede proporcionar insights para mejorar la composición de los equipos y su eficiencia en el cumplimiento de plazos.

El análisis de los datos presentados en la Tabla 13 muestra que los equipos de entre 6 y 10 personas tienen el mayor grado de incumplimiento de fechas, con un 11,69% en aquellos que revisan sus procesos trimestralmente, y un 10,39% en los que lo hacen mensualmente. Asimismo, los equipos más pequeños (1-5 personas) que realizan sus revisiones de forma trimestral también presentan un porcentaje significativo de incumplimiento (9,09%). Esto sugiere que la frecuencia de las revisiones y ajustes en los procesos podría estar vinculada al incumplimiento de plazos, por lo que sería necesaria una investigación más profunda para sustentar esta afirmación y comprender mejor cómo la periodicidad de las revisiones afecta la capacidad de los equipos para adaptarse a los desafíos del proyecto.

Otra línea de investigación interesante sería realizar un estudio comparativo sobre cómo

la frecuencia de las revisiones impacta no solo en el cumplimiento de plazos, sino también en la productividad general y satisfacción del equipo. Examinar si una mayor frecuencia de revisiones mejora el flujo de trabajo y permite identificar problemas antes de que afecten los tiempos de entrega podría ofrecer estrategias útiles para optimizar el rendimiento de los equipos.

En cuanto a la capacitación continua, sería relevante investigar cómo los programas de formación estructurada en metodologías ágiles y técnicas de estimación pueden reducir los márgenes de error y mejorar la capacidad de los equipos para cumplir con los plazos. Esto sería particularmente beneficioso para los equipos más grandes, que suelen tener dificultades adicionales en coordinación y comunicación. Evaluar la efectividad de estos programas en equipos de diferentes tamaños y con distintas frecuencias de revisión ayudaría a identificar las mejores estrategias de capacitación.

Finalmente, profundizar en la relación entre la cultura organizacional y la adopción exitosa de metodologías ágiles podría ofrecer recomendaciones valiosas. Investigar cómo la cultura organizacional influye en la capacidad de los equipos para implementar adecuadamente estas metodologías, y cómo se puede fomentar una cultura de adaptabilidad y mejora continua, es fundamental para asegurar el éxito en la adopción de enfoques ágiles y el cumplimiento de plazos.

En resumen, estas líneas de investigación proponen un análisis más detallado de la relación entre el tamaño de los equipos, la frecuencia de las revisiones de procesos, y el impacto de la cultura organizacional y la capacitación continua en la efectividad de los equipos. Estas áreas de estudio pueden ofrecer soluciones específicas para mejorar la eficiencia y el cumplimiento de plazos en la industria del software.

10 - Referencias

- Achilli, M. (2022). *Problemas frecuentes en la aplicación de Scrum en el desarrollo de software en empresas argentinas* [Trabajo Final de Carrera, Universidad Abierta Interamericana]. Repositorio Institucional - Universidad Abierta Interamericana.
- Anderson, D.J. (2010). *Kanban: Successful evolutionary change for your technology business*. Blue Hole Press.
- Anderson, D. J., y Carmichael, A. (2016). *Essential Kanban Condensed*. Lean Kanban University Press.
- Beck, K., Beedle, M., Bennekum, A. V., Cockburn, A., Cunningham, W., Fowler, M., Martin, R. C., Mellor, S., Thomas, D., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Schwaber, K. y Sutherland, J.(2001). *Manifiesto for Agile Software Development*. Agile Alliance. <https://www.agilealliance.org/agile101/the-agile-manifesto/>
- Briano, C. A. (2023). *Compilación de apuntes sobre conceptos fundamentales de la Ingeniería de Software*. <https://briano.com.ar/libro>
- Brooks, F. P. (1975). *The Mythical Man-Month: Essays on Software Engineering*. Addison-Wesley.
- Capuñay Uceda, O.E. y Antón Perez, J.M. (2021). Influencia de SCRUM en los plazos de entrega y rendimiento en los proyectos de las asignaturas de Desarrollo de Software. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, (29), 36-42. <https://doi.org/10.24215/18509959.29.e4>

-
- Cohn, M. (2006). *Agile estimating and planning*. Pearson Education, Inc.
- DeMarco, T. y Lister, T. (2013). *Peopleware: Productive Projects and Teams*. Addison-Wesley.
- Fibonacci, L. (2002). *Liber Abaci*. Springer.
- Freeman, R. E. (1984). *Strategic Management: A Stakeholder Approach*. Cambridge University Press.
- Guerrero-Calvache, M., y Hernández, G. (2023). Un estudio exploratorio de las percepciones de productividad en equipos de software ágil. *TecnoLógicas*, 26(56), e2625.
<https://doi.org/10.22430/22565337.2625>
- Gygi, C. y Williams, B. (2012). *Six Sigma For Dummies*. John Wiley & Sons, Inc.
- Hernández Sampieri, R. (2014). *Metodología de la investigación*. McGraw-Hill Interamericana Editores, S.A. de C.V.
- Highsmith, J. (2010). *Agile Project Management: Creating Innovative Products (2 ed.)*. Pearson Education.
- Hoy, F., Longenecker, J. G., Palich, L. E. y Petty, J. W. (2018). *Administración de pequeñas empresas*. Cengage Learning.
- IEEE. (1990). *IEEE Standard Glossary of Software Engineering Terminology*. IEEE.
<https://ieeexplore.ieee.org/document/159342>
- Jørgensen, M. (2004). A review of studies on expert estimation of software development effort. *Journal of Systems and Software*, 70(1–2), 37–60.
- Kerzner, H. (2017). *Project Management: A Systems Approach to Planning, Scheduling, and Controlling* (12th ed.). Wiley.
- Kieran, C. (2009) *Agility from First Principles: Reconstructing the Concept of Agility in*

Information Systems Development. *Information Systems Research* 20(3), 329 - 354.

<https://doi.org/10.1287/isre.1090.0236>

Larman, C. y Vodde, B. (2015). *Large-Scale Scrum: More with LeSS*. Addison-Wesley Professional.

Maldonado Arango, M. A. (2017). *Factores que afectan la productividad en equipos Scrum analizados con Pensamiento sistémico* [Tesis de Magíster en Ingeniería, Universidad Nacional de Colombia]. Repositorio Institucional - Universidad Nacional de Colombia.

Mendes, E. (2007). *Cost Estimation Techniques for Web Projects*. Igi Publishing

¿Por qué los equipos de Scrum tienen un tamaño limitado? (2017, agosto 21). *Tales of Agile Adoption*.

<https://tales-of-agile-adoption.blogspot.com/2017/08/por-que-los-equipos-de-scrum-tiene-n-un.html>

Pressman, R. S. (2010). *Ingeniería del software*. McGraw-Hill Interamericana de España.

Ramírez, A., y Orantes, S. (2011). Prototipo de una herramienta de apoyo para la estimación de costos, en la etapa de desarrollo de un proyecto de Software. *Revista Digital Universitaria. Universidad Nacional Autónoma de México*, 12(6), 1–15.

Rock, D. (2009). *Your Brain at Work: Strategies for Overcoming Distraction, Regaining Focus, and Working Smarter All Day Long*. HarperCollins.

Ruíz Constanten, Y., y Cordero Morales, D. (2013). Estimación en proyectos de software integrando los métodos de Boehm y Humphreyc. *Revista Cubana de Ciencias Informáticas*, 7(3), 23–36.

Salvetto De León, P. F. (2006). *Modelos automatizables de estimación muy temprana del tiempo*

y esfuerzo de desarrollo de sistemas de información. Universidad Politécnica de Madrid.

Sabogal Serrano, C. M. (2017). *Un caso de estudio sobre los factores que afectan la estimación de proyectos de software con base en el repositorio ISBSG* [Tesis de Magíster en Ingeniería, Universidad Nacional de Colombia]. Repositorio Institucional - Universidad Nacional de Colombia.

Schwaber, K. y Sutherland, J. (2017). *The Scrum Guide*. Scrum.org. <https://scrumguides.org/>

Sommerville, I. (2005) *Ingeniería del Software (Séptima Edición)*. Pearson Educación S.A. Madrid.

Sutherland, J. (2014). *Scrum: The Art of Doing Twice the Work in Half the Time*. Random House Business.

11 - Anexos

11.1 - Anexo 1: Modelo De Entrevista

11.1.1 - Datos Demográficos y Contextuales

1. ¿Cuál es su cargo actual?
2. ¿En qué tipo de empresa trabaja?
 - a. Startup
 - b. Mediana
 - c. Gran corporación
 - d. Otra
3. ¿En qué sector se encuentra su empresa?
 - a. Finanzas
 - b. Tecnología
 - c. Salud
 - d. Educación
 - e. Gubernamental
 - f. Otra

11.1.2 - Características Del Proyecto

1. ¿En qué tipo de proyecto trabaja principalmente?
 - a. Soporte
 - b. Evolutivo
 - c. Innovación
 - d. Investigación
 - e. No estoy seguro
 - f. Otra
2. ¿Cuál es la duración promedio de los proyectos en los que trabaja?
 - a. Menos de 3 meses
 - b. 3 - 6 meses

-
- c. 6 - 12 meses
 - d. Más de 12 meses
 - e. Otra
3. ¿Cuál es el presupuesto promedio de los proyectos en los que trabaja? (Expresado en dólares estadounidenses)
- a. Menos de U\$S 10.000
 - b. U\$S 10.000 - U\$S 50.000
 - c. U\$S 50.000 - U\$S 100.000
 - d. Más de U\$S 100.000
 - e. No lo sé

11.1.3 - Metodologías Y Prácticas De Trabajo

- 1. ¿Qué metodología de trabajo utiliza principalmente?
 - a. Scrum
 - b. Kanban
 - c. Lean
 - d. XP
 - e. Cascada
 - f. Mixta
 - g. No lo sé
 - h. Ninguna
 - i. Otra
2. ¿Ha recibido capacitación formal sobre la metodología que utiliza?
- a. No
 - b. Sí
3. ¿Ha adaptado o personalizado la metodología utilizada para ajustarse mejor a las necesidades del proyecto?
- a. Sí
 - b. No
 - c. No sé
4. ¿Qué herramientas ágiles utiliza su equipo? (Múltiples respuestas)
- a. Jira
 - b. Trello

-
- c. Asana
 - d. Slack
 - e. Miro
 - f. Otra

11.1.4 - Composición Y Dinámica Del Equipo

1. ¿Cuál es el tamaño promedio de su equipo de trabajo?
 - a. 1 - 5
 - b. 6 - 10
 - c. 11 - 20
 - d. Más de 20

2. ¿Cuál es la cantidad que posee su equipo de cada uno de estos puestos? Responda con números
 - a. Desarrolladores
 - i. 0
 - ii. 1-3
 - iii. 3-5
 - iv. Más de 5
 - b. Diseñadores
 - i. 0
 - ii. 1-3
 - iii. 3-5
 - iv. Más de 5
 - c. Product Owners
 - i. 0
 - ii. 1-3
 - iii. 3-5
 - iv. Más de 5
 - d. Project Managers
 - i. 0
 - ii. 1-3
 - iii. 3-5
 - iv. Más de 5
 - e. QA/Testers
 - i. 0
 - ii. 1-3

iii. 3-5

iv. Más de 5

3. ¿Con qué frecuencia realiza reuniones de seguimiento?

- a. Diariamente
- b. Semanalmente
- c. Quincenalmente
- d. Mensualmente
- e. Otra

4. ¿Qué canales de comunicación utiliza su equipo?

- a. Email
- b. Chat
- c. Reuniones presenciales
- d. Videoconferencias
- e. Otra

11.1.5 - Estimaciones

1. El equipo para el que trabaja, realiza las estimaciones :

- a. De forma ágil
- b. En tiempo
- c. No realizamos estimaciones
- d. Otra

2. Las estimaciones son realizadas por...

- a. Por todo el equipo sea cual sea su cargo.
- b. Por los líderes técnicos
- c. Por los Product Owner
- d. Solo por los desarrolladores
- e. No lo sé
- f. Otra

3. El trabajo es estimado en:

- a. Story Points, entendiendo el peso/esfuerzo involucrado para llevar a cabo la tarea.
- b. Story Points, asociando esos puntos a una determinada cantidad de tiempo.
- c. En tiempo, asociando cada tarea a una cantidad de tiempo.
- d. Otra

11.1.6 - Interacción Con Otros Equipos

1. ¿Con qué frecuencia interactúa con otros equipos?
 - a. Diariamente
 - b. Semanalmente
 - c. Quincenalmente
 - d. Mensualmente
 - e. Otra

2. ¿Qué tan bien coordinadas están las tareas entre su equipo y otros equipos?
 - a. Excelente
 - b. Muy bien
 - c. Bien
 - d. Regular
 - e. Neutral
 - f. Mal
 - g. Muy mal

11.1.7 - Factores De Fracaso

1. En su experiencia, ¿cuáles son las principales causas del fracaso en el cumplimiento de los plazos de entrega? (Múltiples respuestas)
 - a. Estimaciones incorrectas
 - b. Falta de recursos
 - c. Problemas de comunicación
 - d. Falta de capacitación en metodologías ágiles
 - e. Cambios en los requisitos del proyecto
 - f. Falta de involucramiento de los stakeholders
 - g. No lo sé
 - h. Otra

2. ¿Con qué frecuencia su equipo no cumple con los plazos de entrega?
 - a. Siempre
 - b. Frecuentemente
 - c. A veces
 - d. Raramente

- e. Nunca
 - f. Otra
-
- 3. ¿Qué impacto tienen los fracasos en los plazos de entrega en su equipo? (Múltiples respuestas)
 - a. Alta rotación de personal
 - b. Reducción de moral
 - c. Pérdida de clientes
 - d. Incremento de costos
 - e. Otra

11.1.8 - Evaluación Y Mejoras

- 1. ¿Qué tan preciso considera que es su proceso de estimación actual?
 - a. Muy preciso
 - b. Preciso
 - c. Neutral
 - d. Poco preciso
 - e. Nada preciso
 - f. Otra

- 2. ¿Con qué frecuencia se revisa y ajusta el proceso de trabajo en su equipo?
 - a. Mensualmente
 - b. Trimestralmente
 - c. Anualmente
 - d. Nunca
 - e. Quincenalmente
 - f. No sé
 - g. Otra