Web development based on design-transfer through client adaptations

Jonathan Delgado Guerrero[1[0000-0003-0724-3358]], Leandro Antonelli[2[0000-0003-1388-0337]] and Diego Firmenich[3[0000-0002-7212-4454]]

[1,2] Lifia, Fac. de Informática, UNLP, calle 50 esq 120, La Plata, Bs As, Argentina
[1] SENESCYT, Ecuador
[2]CAETI - Facultad de Tecnología Informática - Universidad Abierta Interamericana
[3]Departamento de Informática, Facultad de Ingeniería, UNPSJB, Chubut, Argentina.
[1]jonathan.delgadog@info.unlp.edu.ar,
[2]lanto@lifia.info.unlp.edu.ar, [3]dafirmenich@ing.unp.edu.ar

Abstract. Designing high-fidelity prototypes that cover both visual and behavioral aspects can be challenging. Using interaction patterns as requirements can help develop such prototypes. Patterns assist product understanding and improve early corrections. In this paper, we propose a method for specifying component interactions using existing interaction patterns in the literature as part of the requirements. We also present an experimental tool that uses client-side adaptation on the web prototype built using interface design tools to allow associating the interaction patterns to the visual components and thus transferring the design and behavior to code. This approach allows the creation of functional components with reusable code that facilitates the implementation in the development phase. The article presents the initial state, detailing the method and the developed tool.

Keywords: Interaction Requirements, Design-transfer, Client Adaptation, High fidelity prototyping.

1 Introducción

En el desarrollo web, los prototipos se utilizan para mostrar el diseño de interfaces de usuario relacionadas con requisitos previamente recopilados. La complejidad del prototipo depende del alcance del proyecto y de la calidad de los requisitos. Una complejidad elevada y una buena calidad de los requisitos aumentarán el tiempo y el consumo de costos del software. Un prototipo de baja complejidad tendrá un menor tiempo de desarrollo, pero el coste dependerá de la calidad de los requisitos.

El desarrollo de prototipos implica la utilización de herramientas y tecnologías para crear un prototipo de interfaz de usuario, desde abstracciones básicas hasta entregables más detallados que brindan una mejor comprensión del diseño de la interfaz.

Estos componentes representan reglas que forman parte de un requisito recogido con una metodología de ingeniería de requisitos. Por ejemplo, si un requisito es realizar una transferencia de dinero, implica la creación de una interfaz de usuario con campos de

entrada para la validación de datos y la retroalimentación. Estos son componentes individuales con su propio comportamiento, diseño y relevancia para el requisito.

La disponibilidad de un prototipo de alta fidelidad en las primeras etapas del desarrollo de software podría brindar una visión más clara al usuario final, permitiendo mejoras y correcciones antes de la codificación. La participación de los usuarios ayuda a refinar los requisitos y agregar escenarios no detectados inicialmente.

Transferir el diseño visual de un prototipo a código HTML y CSS durante el desarrollo facilita la reutilización y minimiza errores y diferencias. Las herramientas de prototipado web permiten inspeccionar los prototipos y posibilita transferir la lógica de validación especificada en requisitos, beneficiando al equipo de desarrollo con un código más elaborado y alineado a las expectativas del usuario.

Este artículo propone un método para incorporar patrones de interacción existentes en la literatura como parte de los requisitos. Además, este trabajo consiste en crear una herramienta experimental, que haga uso de este nuevo método para a través de adaptaciones del lado del cliente, transferir diseño y comportamiento a código HTML/CSS/JS para crear componentes cuya definición en código facilitaría hacerlos reutilizables. Este método tiene por objetivo disminuir el tiempo de desarrollo del software al disponer de componentes funcionales de diseño y comportamiento al inicio de la fase de desarrollo.

El artículo se estructura en seis secciones. La sección 2 presenta el estado del arte, la sección 3 aborda el problema y la contribución del trabajo, la sección 4 describe la metodología de investigación, la sección 5 presenta el plan de evaluación de la propuesta y la sección 6 contiene las conclusiones.

2 Estado del Arte

Existen propuestas que orientan el uso de metodologías para diseñar interfaces de usuario durante la fase de recopilación de requerimientos. [1].

Tenemos metodologías donde se compone una metodología ágil con una estrategia de modelado de prototipado a través de mockups conducidos por modelos para permitir invertir el proceso de desarrollo, logrando así un proceso iterativo basado en el prototipado con las características de un diseño conducido por modelos [2].

En [3] se utilizó la metodología ágil WebTDD para diseñar interfaces de usuario accesibles, centrándose en la participación continua del usuario. Se usó un lenguaje específico del dominio para generar pruebas y seguimientos de los cambios en los requisitos, al tiempo que se creaban prototipos para el análisis y el diseño de la interfaz.

Hay trabajos en donde la metodología de diseño iterativo se combina con estrategias centradas en el usuario para construir modelos de interacción basados en votaciones de usuarios [4]. Otra aplicación de metodologías se presenta en un caso de estudio en el que se utiliza el diseño incremental junto con el modelado web para mejorar la usabilidad de una aplicación durante las fases de análisis, diseño y evaluación del diseño incremental [5]. Estos estudios no se centran en los componentes visuales relacionados con la interacción de datos, sino en el esquema y la navegación. Estos aspectos son importantes para la usabilidad y deben abordarse en fases posteriores del desarrollo de software más relacionadas con la codificación. Además, existen patrones de diseño de

interacción en la literatura que establecen principios básicos de diseño para los controles de la interfaz de usuario, como la coherencia, la simplicidad y la claridad [6].

Las herramientas de creación de prototipos permiten diseñar componentes visuales, tanto sencillos como complejos, que pueden transferirse a código. Existen metodologías para identificar los componentes básicos que se utilizan habitualmente a lo largo del proceso de desarrollo de software [7]. Este enfoque de representar elementos en un nivel más pequeño e indivisible se utiliza para detectar y generar componentes combinándolos y agrupándolos [8]. La adaptación del lado del cliente se utiliza en el desarrollo de herramientas de creación de prototipos para mejorar la colaboración en proyectos de rediseño web. También se emplea en el diseño de interfaces de usuario para mejorar la experiencia del usuario al considerar las diversas interacciones que pueden ocurrir en diferentes dispositivos simultáneamente [9].

3 Declaración del Problema y Contribuciones

Existen métodos para recopilar requisitos y crear prototipos de interfaz de usuario. Sin embargo, no existe ningún método para incorporar la interacción detallada de los componentes visuales relacionados con un requisito concreto. El usuario puede conocer la presentación visual y comportamiento de la aplicación, lo que permite mejoras y correcciones oportunas y reducción de tiempo de desarrollo mediante componentes pequeños. Con base en lo mencionado se busca brindar aporte en los siguientes aspectos:

- Diseñar un método que permita enriquecer los componentes de una interfaz de usuario a través de la relación de los patrones de interacción existentes en la literatura como parte de la etapa de levantamiento de requerimientos.
- Diseñar y crear una herramienta que permita extender la visualización web del prototipo creado en una plataforma de diseño de interfaces a través del uso de adaptación del cliente para poder capturar la funcionalidad del componente y facilitar la transferencia de su diseño y comportamiento hacia código HTML/CSS/JS.
- Evaluar el método y la herramienta mediante una encuesta SUS para medir su usabilidad. Además, elaborar un caso de estudio para evaluarlos con más detalle.

4 Metodología de investigación y enfoque

La investigación comenzará con un enfoque cualitativo en la revisión de la literatura sobre transferencias de diseño de interfaz de usuario. Luego se diseñará un método para descubrir y enriquecer componentes de la interfaz, seguido por la implementación de una herramienta de software experimental. Finalmente, se llevará a cabo un experimento para evaluar y mejorar el método y la herramienta desarrollados.

5 Plan de Evaluación

La evaluación se realizará mediante una encuesta SUS para comprobar si el método y la herramienta son fáciles de utilizar. También se llevará a cabo un caso de estudio

para evaluar con mayor detalle el método y la herramienta. Si es necesario, se realizará un experimento para evaluar aspectos concretos.

6 Conclusiones

Este estudio busca definir un método que integre patrones de diseño de interacción, existentes en la literatura, con los requisitos y los convierta en entradas para el desarrollo de prototipos. Esto permitirá a los usuarios tener una mejor comprensión del producto final y realizar mejoras o correcciones en etapas tempranas con la participación del usuario. Sería beneficioso crear una herramienta experimental para transferir este diseño a componentes reutilizables. Con este método se busca una reducción en el tiempo de desarrollo al contar con componentes funcionales desde el inicio.

7 Etapa Doctoral

Early Stage -ES-

References

[1] T. S. Schmidt *et al.* «Media richness theory in agile development choosing appropriate kinds of prototypes to obtain reliable feedback», *international conference on engineering, technology and innovation (ICE/ITMC)*, IEEE, 2017, pp. 521-530.

[2] M. Urbieta *et al.* «MDWA: a model-driven Web augmentation approach coping with client and server side support», *Softw. S Model.*, vol. 19, pp. 1541-1566, 2020.

[3] N. Medina *et al.*, «An Incremental Approach for Building Accessible and Usable Web Applications, in Web Information Systems Engineering WISE, vol. 6488». Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[4] Z. Hussain *et al.*, «User interface design for a mobile multimedia application: An iterative approach», en *First International Conference on Advances in Computer-Human Interaction*, IEEE, 2008, pp. 189-194.

[5] L. Olsina *et al.*, «Incremental quality improvement in web applications using web model refactoring», en *Workshops: WISE 2007 Nancy, France, December 3, 2007 Proceedings 8*, Springer, 2007, pp. 411-422.

[6] J. Tidwell, *Designing interfaces: Patterns for effective interaction design*. O'Reilly Media, Inc., 2010.

[7] F. Brad, «Atomic Design», *Pittsburgh Pa. Ebook*, 2016.

[8] B. Aşıroğlu *et al.*, «Automatic HTML code generation from mock-up images using machine learning techniques», en *Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT)*, Ieee, 2019, pp. 1-4.

[9] S. Firmenich *et al.*, «Distributed Web browsing: supporting frequent uses and opportunistic requirements», *Univers. Access Inf. Soc.*, vol. 18, pp. 771-784, 2019.