



Tipo de artículo: Artículos originales
Temática: Ingeniería de software
Recibido: 15/03/2023 | Aceptado: 20/06/2023 | Publicado: 30/09/2023

Identificadores persistentes:
DOI: [10.48168/innosoft.s12.a93](https://doi.org/10.48168/innosoft.s12.a93)
ARK: [ark:/42411/s12/a93](https://nbn-resolving.org/ark:/42411/s12/a93)
PURL: [42411/s12/a93](https://nbn-resolving.org/urn:nbn:org:cc:42411-s12-a93)

Cómo integrar Design Thinking en el Ciclo de Vida de Verificación y Validación de Software para Simuladores de Vuelo

How to integrate Design Thinking in the Verification and Validation of Software's Life Cycle for Flight Simulators

Valeria Raquel Filoniuk ^{1[0000-0003-0614-3814]*}, María Soledad Martínez ^{2[0000-0003-2346-9859]}, Ana Claudia Diz ^{3[0000-0002-0585-860X]}, Silvia Edith Arias ^{4[0000-0001-9695-2812]}

¹Dirección de Análisis Operativo. Fuerza Aérea Argentina, Córdoba, Argentina. vfiloniuk@gmail.com

²Dirección de Análisis Operativo. Fuerza Aérea Argentina, Córdoba, Argentina. mariasolemartinez81@gmail.com

³Dirección de Análisis Operativo. Fuerza Aérea Argentina, Córdoba, Argentina. anaclaudiadiz@gmail.com

⁴Universidad Nacional de Córdoba, Universidad Tecnológica Nacional, Córdoba, Argentina. edith.edit@gmail.com

* Autor para correspondencia: vfiloniuk@gmail.com

Resumen

Este artículo presenta la aplicación de la Metodología Design Thinking para dar una solución creativa e innovadora en el Ciclo de Vida de Verificación y Validación de Software en Simuladores de Vuelo, de tal manera de optimizar su funcionalidad. La utilización de este método en las primeras etapas de dicho proceso permite una interacción activa y temprana con el usuario, aumentando la generación de opciones de solución y la identificación de necesidades, consiguiendo contribuir a la calidad del producto final. Es importante asegurar la calidad en el desarrollo de software de simuladores de vuelos, debido a su funcionamiento crucial y a la esencialidad que aportan en la defensa aérea de nuestro país.

Para mejorar el proceso mencionado se trabaja principalmente con tres escenarios de Design Thinking: Ideación, Inspiración e Implementación. El objetivo principal de este artículo es integrar estos escenarios en el Ciclo de Vida de Verificación y Validación de manera que ayuden a desarrollar una nueva forma de testear software, incluyendo al usuario como fuente estratégica. Se espera que los resultados contribuyan a crear productos de calidad, ya que los mismos representan un activo esencial para la formación de pilotos que asumen el desafío de custodiar el cielo argentino.

Palabras clave: Calidad, Design Thinking, Simulador de Vuelo, Usuario, Verificación y Validación de Software.

Abstract

In order to ensure the Flight Simulator's functionality, this article exposes the application of the "Design Thinking" Methodology. This will lead to provide more creative and innovative solutions during the Verification and Validation process in the Flight Simulator Software's Life Cycle. The application of this method in the first stages of the process allows an active and early interaction with user's needs, increasing the building of solutions and the identification of needs, heighten the final product's standards. It is important to ensure quality in the software development of flight's simulators, due to its application is crucial and essential to the air defense of our country.

So as to improve the aforementioned process, Design Thinking works with three main scenarios: Ideation, Inspiration and Implementation. The main objective of this article is to integrate these scenarios with the Verification and Validation of the Software's Life Cycle and develop a new way to test software, including the user as a strategic source. It is expected that the results will contribute to create high quality products being these vital assets for the pilot's training process who daily assume the challenge of guarding the Argentine sky.

Keywords: *Design Thinking, Flight Simulator, Quality, User, Verification and Validation of Software.*

Introducción

Existen una gran variedad de investigaciones que han sugerido que Design Thinking puede tener un impacto positivo en el desarrollo de software, ya que facilita una profunda comprensión de las necesidades del usuario y aumenta la colaboración del equipo [1].

Los proyectos de desarrollo de software se ven afectados debido a factores que impactan en el resultado de los mismos. Estos factores incluyen requerimientos incompletos, demoras, incapacidad para empatizar e incorporar al usuario final y diseñar soluciones que respondan a sus necesidades reales. Todo ello contribuye a incrementar el costo, tiempo, y a disminuir la calidad del producto software [2].

Los pilotos de Fuerza Aérea Argentina utilizan simuladores de vuelo para su formación. Estas herramientas son empleadas para que tengan un entrenamiento previo por medio de software antes de realizar prácticas presenciales y de esta forma tener un mayor conocimiento del funcionamiento de la aeronave [3]. El software utilizado para entrenadores de vuelo representa un activo esencial para la defensa aérea de nuestro país, por lo que se considera de gran criticidad en materia de calidad. En consecuencia, es inevitable incluir a dichos usuarios desde las primeras etapas del proceso de Verificación y Validación de Software, conociendo sus necesidades reales y priorizando sus requerimientos, lo que

permitirá mantenerlos satisfechos. Adicionalmente se obtendrá un software de mejor calidad y más usable, obteniendo una disminución del tiempo invertido por el usuario en su capacitación, reducción de soporte y mantenimiento.

Una alternativa para impulsar el involucramiento del usuario es la Metodología Design Thinking [4], [5], un proceso para incentivar la creatividad e innovación, integrando factores humanos y tecnológicos para la resolución de problemas. Su enfoque centrado en lo humano permite descubrir perspectivas de las personas y usuarios finales [2]. Design Thinking posibilita generar soluciones que se basan en la idea de ponerse en el lugar del usuario final [6], es decir, para quien va dirigido el producto o servicio [7].

Este artículo pretende dar solución a problemas existentes en el proceso de Verificación y Validación de Software para Simuladores de Vuelo. El propósito principal es integrar el enfoque de Design Thinking en las fases de dicho proceso incluyendo al usuario como fuente estratégica y luego analizar los resultados obtenidos.

Materiales y Métodos o Metodología Computacional

Para la presente investigación se revisaron artículos relacionados a técnicas existentes destinadas a integrar Design Thinking (DT) en etapas tempranas del Ciclo de Vida de Verificación y Validación de Software. En las siguientes secciones se muestran conceptos claves relacionados a DT y Verificación y Validación de Software (V&V), y luego se menciona la metodología a aplicar basada en la unificación de ambos conceptos, adaptándola a las necesidades específicas en la organización objeto de estudio.

Design Thinking

Design Thinking permite concebir proyectos informáticos para dar soluciones a los problemas del mundo real. Se centra en las personas, considerando multidisciplinariedad, colaboración y concreción de pensamientos y procesos, los cuales son caminos que llevan a soluciones innovadoras como base para la resolución de problemas, desde el punto de vista del usuario y con el apoyo de prototipos [8].

El proceso de DT se basa en tres pilares importantes [9], [10]:

Inspiración: consiste en identificar la necesidad y el problema para analizar soluciones. Incluye las siguientes etapas:

- Empatizar: comprender lo que el usuario necesita, en términos de lo que siente el usuario.
- Definir: establecer las necesidades del usuario en términos de un problema que se puede resolver.

Ideación: se basa en proponer una solución que satisfaga las necesidades del usuario.

Implementación: desarrollar la solución final, en la que intervienen las siguientes fases:

- Prototipo: crear una solución que cubra una selección de las características requeridas por el usuario, para que pueda experimentarlo.
- Prueba: utilizar la solución real, de manera que el usuario pueda proporcionar comentarios oportunos y el proceso pueda iterar [1].

Verificación y Validación de Software

El *Testing de software* pertenece a una actividad o etapa del proceso de producción de software denominada *Verificación y Validación*. V&V es el nombre genérico dado a las actividades de comprobación que aseguran que el software respeta su especificación y satisface las necesidades de sus usuarios [11]. Si bien estos términos en su uso cotidiano pueden llegar a ser sinónimos, en Ingeniería de Software tienen significados diferentes y cada uno tiene una definición más o menos precisa. Por lo tanto, verificar consiste en comprobar que el software cumple los requerimientos funcionales y no funcionales que se le han especificado; y validar se basa en probar que el software hace lo que el usuario espera [12].

Para llevar a cabo el proceso de V&V, teniendo en cuenta la funcionalidad y complejidad con que operan los simuladores de vuelo, los especialistas los ejecutan en fases que se presentan a continuación:

Tabla 1. Fases del Ciclo de Vida de Verificación y Validación de Software

<i>Primer Fase: Análisis de Requerimientos</i>	<ul style="list-style-type: none"> ✓ Analizar los requerimientos documentados. ✓ Comprender lo que los usuarios esperan que haga el software en base a los requerimientos documentados y priorizarlos.
<i>Segunda Fase: Planificación de Pruebas</i>	<ul style="list-style-type: none"> ✓ Definir el alcance de las pruebas detallando niveles, tipos, estrategias y técnicas de prueba tomando como referencia los requerimientos documentados.
<i>Tercer Fase: Diseño y Desarrollo de Pruebas</i>	<ul style="list-style-type: none"> ✓ Comprobar que los requerimientos documentados estén completos. ✓ Desarrollar casos de prueba en base a los requerimientos documentados. ✓ Probar todas las combinaciones posibles y priorizar cuál de ellas afecta más al producto software.
<i>Cuarta Fase: Ejecución de Pruebas</i>	<ul style="list-style-type: none"> ✓ Documentar novedades y desviaciones del comportamiento del software sin participación del usuario. ✓ Transmitir novedades para que se solucionen.
<i>Quinta Fase: Evaluación de resultados</i>	<ul style="list-style-type: none"> ✓ Validar si los requerimientos implementados cubren las expectativas de los usuarios.
<i>Sexta Fase: Cierre del Proceso</i>	<ul style="list-style-type: none"> ✓ Reportar las novedades del proceso y la aceptación/rechazo del software testeado, sin intervención del usuario final.

La Tabla 1 muestra que los requerimientos son comprendidos y analizados una sola vez y para todo el ciclo y que solo se interactúa con los usuarios en la fase de Evaluación de Resultados. Durante el proceso, los usuarios son ajenos al ciclo de vida de V&V, hasta que el producto se prueba y se determina que hace lo que ellos esperan. Por consiguiente, se llega al final del proceso trabajando en una propuesta que, en determinadas circunstancias, no cumple las expectativas de los usuarios, repercutiendo en la calidad del producto final.

En tal sentido se infiere que podría usarse DT para mejorar el ciclo mencionado. Se ofrece en este marco una metodología ágil, centrada en herramientas innovadoras que se utilizarán a lo largo del proceso y en la interacción continua con los usuarios [13]. Por consiguiente, se implementarán técnicas basadas en la integración de factores humanos y tecnológicos para cada una de las fases enunciadas.

Design Thinking, por lo tanto, puede ser lo suficientemente flexible como para ser tomado como un marco general de desarrollo de soluciones que lo hace adecuado para ser utilizado en los procesos de desarrollo de software, que de hecho

es un campo en el que se requiere un enfoque abierto, creativo y flexible para resolver problemas y encontrar oportunidades de mejora [1].

Resultados y discusión

Utilizar la técnica de Design Thinking en el Ciclo de Vida de V&V, favorece una mejora considerable en la identificación de problemas y necesidades de usuarios, recopilación de nuevas perspectivas de solución, generación de opciones de solución y prototipos que permitan representar dichas ideas [14].

La Tabla 2 expone cómo la metodología citada [14], [15], se adecúa perfectamente a las fases del Ciclo de Vida del Software:

Tabla 2. Integración del Design Thinking en el Ciclo de Vida de Verificación y Validación de Software

<i>Fases del Ciclo de Vida del V&V</i>	<i>Design Thinking</i>		
Análisis de Requerimientos	Empatizar	Observación	<ul style="list-style-type: none"> • Observar qué actividades llevan a cabo los usuarios y cuáles son las más prioritarias.
		Mirar y Escuchar	<ul style="list-style-type: none"> • Pedir a los usuarios que expliquen cómo realizan algunas tareas y que vayan vocalizando lo que pasa por su mente cuando estén trabajando.
		Encuestas	<ul style="list-style-type: none"> • Confeccionar encuestas para que los usuarios expresen sus principales necesidades.
	Definir	Reuniones	<ul style="list-style-type: none"> • Establecer los requerimientos a probar en base a los resultados obtenidos anteriormente.
Planificación de Pruebas	Empatizar	Inspección del Software	<ul style="list-style-type: none"> • Examinar el software a testear con la participación del usuario con el objetivo de identificar qué tipos, estrategias y técnicas de pruebas se ensayarán.
	Definir	Plan de Pruebas	<ul style="list-style-type: none"> • Concretar qué tipos, estrategias y técnicas de pruebas se ensayarán tomando en cuenta las necesidades planteadas por el usuario, a través de un documento de Plan de Pruebas.

Diseño y Desarrollo de Pruebas	Idear	Escenarios	<ul style="list-style-type: none"> • Demostrar que los requerimientos a probar reflejen lo que el usuario espera por medio de la representación de Casos de Uso.
	Prototipar	Casos de Pruebas	<ul style="list-style-type: none"> • Diseñar casos de pruebas, priorizando aquellos que afecten en mayor medida los requisitos del usuario final. • Establecer feedback con el usuario en la resolución de casos de pruebas.
Ejecución de Pruebas	Idear	Planillas de Reportes	<ul style="list-style-type: none"> • Documentar y validar con los usuarios la criticidad de las novedades que contribuyan a solucionar sus necesidades, detallando las desviaciones obtenidas en la ejecución de las pruebas. • Comparar los resultados obtenidos con las necesidades del usuario.
Evaluación de resultados:	Prototipar	Pruebas de Usabilidad	<ul style="list-style-type: none"> • Verificar la facilidad de uso y aprendizaje del software en el entorno real del usuario, a través de la ejecución de un ejercicio práctico de simulación.
	Probar	Pruebas de Aceptación	<ul style="list-style-type: none"> • Validar con el usuario los resultados del ejercicio anterior.
	Probar	Encuestas de Satisfacción	<ul style="list-style-type: none"> • Solicitar comentarios a los usuarios a cerca de la valoración del software testeado.
Cierre del Proceso:	Probar	Informe Final	<ul style="list-style-type: none"> • Dar a conocer al usuario las novedades del proceso y determinar conjuntamente con el mismo, si es necesario llevar a cabo un nuevo ciclo.

Los resultados obtenidos en la aplicación de Design Thinking durante el Ciclo de Vida de V&V se detallan en la Tabla 3, donde podemos observar que es importante tener una comunicación constante con los usuarios para que el software de entrenamiento que ellos operan sea un producto usable y de calidad [8].

Tabla 3. Resultados aplicando Design Thinking

<i>Fases del Ciclo de Vida del V&V</i>	<i>Resultados obtenidos en su ejecución habitual</i>	<i>Resultados obtenidos con la aplicación de Design Thinking</i>
Análisis de Requerimientos	Los requerimientos documentados son analizados, comprendidos y clasificados una sola vez, al comienzo del ciclo, y son utilizados a lo largo de todo el proceso sin involucrar a los usuarios.	Los requerimientos son definidos y priorizados a través del trato directo con los usuarios.
Planificación de Pruebas	Los tipos, estrategias y técnicas de prueba son definidas en base a los requerimientos documentados.	Los tipos, estrategias y técnicas de prueba son identificadas y definidas tomando en cuenta las necesidades planteadas por los usuarios.
Diseño y Desarrollo de Pruebas	Los Casos de Prueba son diseñados y desarrollados en base a requerimientos documentados.	Los Casos de Prueba son diseñados y desarrollados en base a escenarios.
Ejecución de Pruebas	Las desviaciones del comportamiento del software son documentadas y transmitidas sin tener en cuenta la opinión de los usuarios.	Las novedades son documentadas y validadas con el usuario en base a su criticidad, detallando las desviaciones obtenidas en la ejecución de las pruebas.
Evaluación de Resultados	Los requerimientos implementados son validados con el usuario final.	Los requerimientos implementados son validados con el usuario final.
Cierre del Proceso	Se reporta el fin del proceso sin intervención del usuario final.	Se reporta el fin del proceso con intervención del usuario final, determinando si es necesario iterar.

Conclusiones

Los resultados obtenidos en este artículo demuestran que las fases que intervienen en el proceso de Ciclo de Vida de V&V, pueden ser optimizadas mediante la aplicación de Design Thinking. La metodología investigada pretende ser la base para conseguir una mejora sustancial en el desarrollo de software correctamente verificado y validado. Esta debe ser conocida y asimilada por todos los especialistas involucrados en un proyecto software, para impulsar el cambio hacia formas de trabajo más eficientes incluyendo al usuario como fuente estratégica.

Design Thinking constituye una propuesta que propicia la participación del usuario. Su carácter iterativo permite adaptarse a los cambios, estableciendo un mecanismo que permite asegurar la calidad del software y mantener al usuario satisfecho.

Contribución de Autoría

Valeria Raquel Filoniuk: [Conceptualización](#), [Análisis formal](#), [Investigación](#), [Metodología](#), [Redacción - borrador original](#), [Escritura, revisión y edición](#). **María Soledad Martínez:** [Conceptualización](#), [Análisis formal](#), [Investigación](#), [Metodología](#), [Redacción - borrador original](#), [Escritura, revisión y edición](#). **Ana Claudia Diz:** [Conceptualización](#), [Análisis formal](#), [Investigación](#), [Metodología](#), [Redacción - borrador original](#), [Escritura, revisión y edición](#). **Silvia Edith Arias:** [Conceptualización](#), [Análisis formal](#), [Investigación](#), [Metodología](#), [Redacción - borrador original](#), [Escritura, revisión y edición](#).

Referencias

- [1] V. F. C. Osis, D. Q. Soto, A. C. Huarca and J. C. Suyo, “Casos de Estudio de Design Thinking en las etapas de Análisis y Diseño del Desarrollo de Software,” *Revista Innovación y Software*, no.1, vol. 3, pp. 17-29, 2022.
- [2] J. C. Espinoza Vásquez and E. E. Espinoza Zapata, “Marco de trabajo en base a Design Thinking y metodologías ágiles de desarrollo de software”, thesis, Universidad Peruana de Ciencias Aplicadas, Lima, 2017.
- [3] L. A. Baquero Hernández and C. C. León Suarez, “Integración e implementación de realidad virtual para un Simulador de entrenamiento de vuelo de uso libre”, thesis, Fundación Universitaria Los Libertadores, Bogotá, 2022.
- [4] S. I. Pedraza Gutiérrez, J. F. Romero-González, J. C. Güiza Rodríguez and E. W. Giraldo Henao, “Diseño centrado en el usuario y experiencia de usuario en el sistema de control de acceso de la Universidad Libre,” *Revista Científica de Sistemas e Informática*, no.1, vol.3, pp. 426, 2023.

- [5] M. L. F. Rojas, “Pensamiento de diseño y marcos éticos para la Inteligencia Artificial: una mirada a la participación de las múltiples partes interesadas,” *Revista Desafíos*, no.1, vol.35, pp.1, 2023.
- [6] D. M. Ortega Ávila, “Aplicación de la Metodología Design Thinking para el Diseño y Comercialización de Agendas Personalizadas en la Ciudad de Santa Marta”, thesis, Universidad Antonio Nariño, Colombia, 2023.
- [7] E. A. S. Gonzales, C. L. S. Bobadilla, J. C. A. Uñapilco and D. H. Quispe, “Design Thinking para resolver problemas con la selección de métricas en la Calidad del Software,” *Revista Innovación y Software*, no.1, vol.3, p.67-80, 2022.
- [8] J. J. G. Valdivia, S. M. Z. Quentasi, D. M. C Yana, R. E. C. Apaza and Y. P. Vera, “Design Thinking en la Planificación de Pruebas de Software,” *Revista Innovación y Software*, no.2, vol.1, pp. 40-51, 2020.
- [9] E. R. A. Serrato, “Propuesta de Alfabetización Digital para las mujeres beneficiarias del programa “Mujer Es”, de la Fundación Misión y Vida, en el Departamento del Huila, República de Colombia”, Tesis Doctoral, Corporación Universitaria Minuto de Dios, Colombia, 2021.
- [10] A. M. Ospina Rivera, and D. M. González Arévalo “Validación por medio de la aplicación del Design Thinking de la oferta de un servicio de medición de emisiones de gases efecto invernadero para el sector pecuario”, Tesis de Maestría, Universidad EAN, Colombia, 2021.
- [11] L. Merino García, “Desarrollo de un sistema automático de validación y verificación de software,” Thesis, Universidad Pública de Navarra, Pamplona, 2021.
- [12] C. Maximiliano, “Introducción al Testing de Software”, Thesis, Universidad Nacional de Rosario, Santa Fe, 2021.
- [13] M. S. Martínez, D. I. Martínez, V. R. Filoniuk, G. G. Chiappori, A. C. Diz and S. E. Arias, “Aplicación de Norma ISO 9241-11 para la Evaluación de la Usabilidad en Simuladores de Vuelo”, *Revista Innovación y Software*, no.2, vol.3, pp. 70-80, 2022.
- [14] G. I. H. Zapana, N. H. Castro, M. A. S. Tico, E. D. C Choquehuanca and A. D. T. Bejarano, “Aplicación del método Design Thinking en el área de requerimientos de software,” *Revista Innovación y Software*, no.1, vol.2, p. 43-52, 2021.
- [15] B. L. V. Márquez, L. A. I. Hanampa and M. G. M. Portilla, “Design Thinking aplicado al Diseño de Experiencia de usuario”, *Revista Innovación y Software*, no.1, vol. 2, pp.6-19, 2021.