

Implementación de una aplicación web que facilite el uso de técnicas de reducción de casos de prueba del software

Alejandro Beltrán Zuluaga & Geiler Orlando Hipia Mejía

Resumen

Este proyecto se enfoca en la conceptualización, diseño e implementación de una plataforma web que tiene como objetivo principal facilitar el uso de técnicas de reducción de casos de prueba de caja negra, haciendo que sean más accesibles y fáciles de utilizar. En el ciclo de vida del desarrollo de software, las pruebas desempeñan un papel crítico en la detección de fallas y en el aseguramiento de la calidad del producto final. Sin embargo, el proceso de pruebas puede ser costoso y demandar mucho tiempo, debido a que, si el software que se está probando es complejo, con múltiples características y escenarios, se requerirá una cantidad considerable de casos de prueba para lograr una cobertura adecuada. Esto implica más tiempo y recursos para diseñar, ejecutar y mantener esas pruebas. El principal desafío al que nos enfrentamos es la falta de soluciones prácticas y accesibles para la reducción de casos de prueba, así como la curva de aprendizaje y aplicación de estas técnicas, que puede representar una barrera para muchos profesionales y equipos de aseguramiento de calidad. Por lo tanto, este proyecto propone una solución innovadora: una plataforma web dotada de una interfaz intuitiva y amigable, que permita a los usuarios ingresar sus casos de prueba y aplicar técnicas de reducción de manera eficiente y rápida. La plataforma se centrará en técnicas de caja negra, como la partición de equivalencias, el análisis de valores límite y los arreglos ortogonales de Taguchi. Además, se proporcionará una documentación detallada y explicaciones sobre las técnicas implementadas, así como la presentación de resultados claros y relevantes para asegurar un proceso de pruebas exitoso. Como valor añadido, la plataforma contará con una API que permitirá la integración con otras herramientas y sistemas. La expectativa es que esta herramienta optimice tanto el tiempo como los costos asociados al desarrollo de software, potenciando la eficiencia del proceso de pruebas y garantizando la entrega de aplicaciones robustas y seguras.

Introducción

En el desarrollo de software el ciclo de vida del producto final está dado por unas etapas que normalmente son: la planeación, el análisis de requerimientos, el diseño, la implementación, pruebas y mantenimiento. En este proceso complejo y desafiante del desarrollo de software, la calidad del producto es un aspecto fundamental que influye directamente en la satisfacción del usuario y en la confianza que se deposita en la aplicación. Para asegurar dicha calidad, las pruebas de software juegan un papel crucial, y dentro de ellas, las pruebas de caja negra son una de las técnicas más utilizadas y efectivas [1]. El presente proyecto tiene como objetivo abordar el desafío de proveer una herramienta útil en la cual se pueda hacer uso de las técnicas de reducción de casos de prueba de caja negra en el contexto de desarrollo de aplicaciones o software. A lo largo del tiempo las aplicaciones han crecido exponencialmente en tamaño, lo que genera un aumento significativo en la cantidad de casos de prueba necesarios para garantizar la calidad del software y la cobertura adecuada del mismo. En el área de pruebas de un proyecto de software la creación y ejecución de un elevado número de casos de prueba han sido una tarea costosa y trabajosa, ya que

requiere de un personal capacitado para realizarlas, tiempo y esfuerzo. Es aquí donde surge la necesidad de reducir los casos de prueba necesarios de forma fácil y sin comprometer la calidad del producto final. Para la solución de este problema se han presentado diversas propuestas donde la gran mayoría son heurísticos y también hay algunas aplicaciones que solo presentan una técnica y son limitados en cuanto a funcionalidades. Estas propuestas son de gran valor cuando se busca acerca de la reducción del número de casos de prueba ya que son base fundamental para el desarrollo de la aplicación.

Fundamentación teórica

Para el desarrollo del proyecto fue necesario identificar cómo funciona la reducción de casos de prueba dentro del software para encontrar la forma de disminuir el tiempo y el esfuerzo que se le dedica a esta etapa; es así como llegamos a las técnicas de caja negra las cuales nos ayudan a reducir casos de prueba sin necesidad de tener conocimiento interno de la aplicación y pudiendo ser aplicadas a la gran mayoría de tipos de proyecto.

La reducción de casos de prueba implica la disminución del conjunto de pruebas a un subconjunto más pequeño y manejable, sin comprometer la calidad de la validación del software. En nuestro proyecto, la reducción de casos de prueba es el objetivo principal de la herramienta que estamos desarrollando. Los usuarios podrán aplicar diferentes técnicas y estrategias de reducción en la aplicación web para optimizar el proceso de prueba. Al reducir la cantidad de casos de prueba necesarios, se obtienen beneficios significativos, como una mayor eficiencia en el proceso de prueba, ahorro de recursos y tiempo.

Las técnicas de diseño de caja negra, también llamadas pruebas de comportamiento son las que utilizan el análisis de la especificación, tanto funcional como no funcional, sin tener en cuenta la estructura interna del programa para diseñar los casos de prueba y, a diferencia de las pruebas de caja blanca, estas pruebas se suelen realizar durante las últimas etapas de la prueba [2]. El objetivo de las técnicas de caja negra es encontrar errores, funciones incorrectas, errores de

inicialización y finalización, errores en la interfaz y módulos del sistema, mediante un conjunto de datos o pruebas.

Dentro de las técnicas de reducción de casos de prueba de caja negra existen diferentes tipos los cuales cambian la metodología y los pasos a seguir para reducir casos de prueba. En este proyecto se identificaron 3 técnicas las cuales fueron particiones de equivalencia, análisis de valores al límite y arreglos ortogonales.

En la técnica de partición de equivalencia, los valores de entrada del programa o del sistema se dividen en grupos que vayan a tener un comportamiento similar, de manera que puedan ser procesados de la misma forma. Las particiones de equivalencia o clases son aplicables a datos válidos y datos no válidos. También pueden aplicarse a los valores de salida, valores internos, valores relativos al tiempo o a los parámetros de interfaz [3].

La técnica de análisis de valores al límite complementa la técnica de partición de equivalencia u otras pruebas de caja negra. Los valores máximos y mínimos de una partición son sus valores límites [3]. Las pruebas pueden diseñarse para cubrir tanto los valores límites válidos, como no válidos.

Un arreglo ortogonal es un conjunto de columnas que cumplen con una propiedad particular: el producto interno entre cada par de columnas es cero. Esta técnica es especialmente útil en el área de las pruebas de software, ya que permite comprender las relaciones entre diferentes variables. Al utilizar esta técnica, es posible obtener resultados eficientes al combinar diversas

variables en cada prueba, maximizando así la cantidad de valores y variables que se pueden evaluar en un solo caso de prueba. El objetivo de esta técnica es representar todas las posibles combinaciones entre variables y sus valores, lo que acelera el proceso de planificación de las pruebas y permite detectar errores y defectos en menos tiempo y con menos recursos [4].

Resultados

Las áreas principales que se tocaran dentro del área de resultados de el proyecto son: identificación de las técnicas de reducción de casos de prueba donde se van a escoger cada una de las técnicas a implementar dentro de la aplicación web, Implementación de las técnicas donde se muestra cómo se adaptan las técnicas para ser implementadas dentro de la aplicación, desarrollo de la aplicación web donde se muestran las tecnologías usadas dentro de la aplicación y el diseño de la interfaz y la evaluación de la aplicación web donde se verifica que la aplicación funciona como se debe y se dan ejemplos de uso de la aplicación.

Para la identificación de las técnicas lo que se hizo fue escoger las técnicas de reducción de casos de prueba de caja negra que mejor se ajustaran a las necesidades del usuario y de la aplicación. Para esto se seleccionaron 6 de las técnicas de caja negra mas conocidas dentro de la disciplina y se evaluaron con respecto a unos criterios de evaluación los cuales eran: cobertura de casos de prueba, eficiencia en la reducción de casos de prueba, identificación de errores y adaptabilidad. A cada una de las 6 técnicas se les dio una calificación con respecto a los criterios y al final resultaron solo 3 las cuales fueron: particiones de equivalencia, análisis de valores al límite y arreglos ortogonales.

El siguiente capítulo se centra en la implementación de técnicas de reducción de casos de prueba de caja negra que se eligieron

en la identificación de las técnicas. Se describe en detalle cómo se solicitan los parámetros, se generan los casos de prueba y se abordan los desafíos durante la implementación. El capítulo proporciona una visión general de cómo se aplican estas técnicas para facilitar el proceso de pruebas y mejorar la eficiencia en la detección de fallas y aseguramiento de la calidad del software.

En el desarrollo de la aplicación se dan detalles de cómo se hizo la aplicación web, abarcando tanto el desarrollo del backend como del frontend. Se describen las tecnologías utilizadas en cada parte, como Python y FastAPI en el backend, y Next.js de React en el frontend. También se mencionan las ventajas de estas tecnologías, como su alta velocidad de ejecución, facilidad de uso y documentación completa. Además, se hace énfasis en el diseño de la interfaz de usuario de la aplicación. Se destaca la importancia de crear una interfaz intuitiva y estética para captar la atención de los usuarios. Se mencionan herramientas utilizadas, como Bootstrap, que facilitan la creación de interfaces atractivas y responsivas.

Para la ultima etapa del proyecto tenemos la evaluación de la aplicación web, en la cual se detalla el plan de pruebas que tiene como objetivo garantizar la precisión de los resultados, verificar la cobertura de los algoritmos y asegurar el manejo adecuado de entradas inválidas. Se describen los casos de prueba definidos para cada técnica implementada, incluyendo pruebas con parámetros válidos e inválidos. Además de esto se realizan ejemplos los cuales buscan probar la aplicación y compara los resultados que esta arroja, por medio de un ejemplo practico se puede comparar la eficiencia y eficacia de cada una de las técnicas con respecto a los casos de prueba que cada una logra reducir con respecto a probar todas las combinaciones dentro del software. En la tabla (Tabla 1) se pueden observar los

resultados lanzados por cada algoritmo en términos de números de casos prueba comparados con el método exhaustivo.

Técnica	# de casos de prueba
Prueba exhaustiva	395.303
Particiones de equivalencia	9
Valores al límite	29
Arreglos ortogonales	16

Tabla 1: Resultado prueba cuantitativa reducida

Además de el desarrollo de la aplicación web, también se implementó una API como una solución para mejorar la eficiencia y efectividad en la reducción de casos de prueba, esta tiene funcionalidades clave diseñadas específicamente para facilitar el uso de los algoritmos de reducción de casos de prueba.

Discusión

En este proyecto, llevamos a cabo el desarrollo e implementación de una aplicación web que utiliza técnicas de reducción de casos de prueba de caja negra. Identificamos y seleccionamos las técnicas más usadas en el contexto de pruebas. Para cada técnica, evaluamos sus ventajas y desventajas, y con base a esto seleccionamos aquellas que brindarían mayor valor al usuario. Además de la implementación de las técnicas de reducción de casos de prueba, diseñamos una interfaz de usuario intuitiva y fácil de usar para nuestra aplicación web. Siguiendo los principios de diseño centrados en el usuario y la usabilidad, nos aseguramos de proporcionar una experiencia agradable para el usuario. También integramos la capacidad de presentar los resultados generados por la aplicación en formatos Json y XLSX.

Se evidenció a través de ejemplos concretos, que la aplicación de las técnicas de reducción de casos de prueba resultó en una disminución significativa en la cantidad total de casos de prueba necesarios. Esto demuestra la eficacia de dichas técnicas para optimizar y simplificar el proceso de pruebas en la aplicación web desarrollada.

Después de revisar los resultados obtenidos y completar todos los objetivos propuestos inicialmente, identificamos algunas áreas de mejora durante el desarrollo del proyecto. En primer lugar, consideramos que sería interesante implementar métricas que permitan al usuario observar de manera más clara cómo la aplicación reduce los casos de prueba y el porcentaje de reducción logrado por la técnica seleccionada. Esto proporcionaría una mejor comprensión de los beneficios que ofrece la aplicación en términos de optimización de las pruebas.

En resumen, para futuras mejoras consideramos implementar métricas visuales, explorar técnicas más avanzadas de reducción de pruebas y trabajar en la expansión y divulgación de la aplicación en diferentes ámbitos. Estas acciones contribuirán a aumentar su valor y su impacto en la comunidad interesada en las pruebas de software.

Referencias

- [1] Boris Beizer. Black-Box Testing: Techniques for Functional Testing of Software and Systems. 1995
- [2] José Sánchez. Pruebas de Software. Fundamentos y Técnicas. Junio 2015, pp. 39
- [3] ISTQB. International Software Testing Qualifications Board. Spanish Software Testing Qualifications Board con el apoyo del Hispanic America Software Testing Qualifications Board, Certified Tester - Foundation Level, Version 2018, pp. 73
- [4] Montalbán Riera. Optimiza tus pruebas de software con arreglos ortogonales. Abril 2023