Plan de Test

Digital Money House

Equipo 12



# 

[**1. Introducción**](#_4ikbqq8uuzy2)

[1.1 Alcance](#_pv85w9jh612z)

[1.2 Referencias](#_tofw6m85e8j9)

[1.3 Definiciones y abreviaturas](#_u77zp7i8wddj)

[**2. Objetivos del test**](#_rxq2p4girk0r)

[2.1 Antecedentes](#_40car067bamq)

[2.2 Objetivo de la Evaluación](#_k8qpyk92je3n)

[2.3 Motivaciones del Test](#_d444arugoen8)

[**3. ÍTEMS A TESTEAR**](#_kluq4ncvpc4f)

[**4. MÉTODOS**](#_iup6xxyzswtx)

[**5.CASOS DE PRUEBAS**](#_l96vnv7j5e8u)

[**6.CRITERIOS DE ACEPTACIÓN**](#_9o37bsqtp96)

[6.1 Plan de Test](#_6pdigbrd8f9e)

[6.1.1 Criterio de Entrada del Plan de Test](#_4pwcf18oeezs)

[6.1.2 Criterio de salida del Plan de Test](#_22wl7ji91t4c)

[6.1.3 Criterio de suspensión y reanudación](#_yrw4sa6bynnv)

[6.2 Ciclo de Test Criterio de entrada del Ciclo de Test](#_svqqugdq34zs)

# 

# Introducción

## 1.1 Alcance

Este Plan de Test abordará los niveles de testing mencionados a continuación y se incluirán los tipos de testing pertinentes para asegurar la calidad y funcionalidad del software bajo prueba. Cada nivel y tipo de testing se llevará a cabo de acuerdo con los objetivos y requisitos específicos del proyecto.

Niveles de Testing:

Pruebas de Unidad: Este nivel de testing se centrará en probar cada componente individual de software de manera aislada para verificar su correcto funcionamiento. Se enfoca en probar funciones específicas y detectar posibles errores a nivel de código.

Pruebas de Integración: En este nivel, se probarán las interacciones y la integración entre diferentes componentes o módulos del software. El objetivo es asegurar que las partes del sistema funcionen correctamente juntas y que los datos se transmitan y procesen adecuadamente entre ellas.

Pruebas de Sistema: Estas pruebas se realizan en el sistema completo para verificar su comportamiento general y asegurar que cumple con los requisitos establecidos. Se evaluará el sistema como una entidad única, incluyendo su funcionalidad, rendimiento y compatibilidad con el entorno de implementación.

Tipos de Testing:

Pruebas Funcionales: Estas pruebas se centrarán en verificar si el software cumple con los requisitos y especificaciones funcionales establecidos. Se evaluará la funcionalidad del sistema, incluyendo la entrada y salida esperada, la lógica de negocio y la interacción con el usuario.

Pruebas de Usabilidad: Estas pruebas se enfocarán en evaluar la facilidad de uso y la experiencia del usuario. Se buscará determinar si el software es intuitivo, eficiente y satisfactorio para quienes lo utilizan, teniendo en cuenta aspectos como la navegación, el diseño de la interfaz y la accesibilidad.

Pruebas de Confiabilidad: Estas pruebas se realizarán para evaluar la estabilidad y confiabilidad del software. Se buscará identificar posibles fallos o problemas que puedan afectar su funcionamiento, como errores de manejo de errores, recuperación ante fallas y capacidad para manejar altas cargas de trabajo.

Pruebas de Desempeño: Estas pruebas se centrarán en evaluar el rendimiento del software en términos de velocidad, capacidad, escalabilidad y eficiencia. Se buscará determinar si el sistema cumple con los criterios de desempeño establecidos, como tiempos de respuesta aceptables y capacidad para manejar un alto volumen de datos.

Pruebas de Soportabilidad: Estas pruebas se realizarán para evaluar la capacidad del software para ser instalado, ejecutado y mantenerse en diferentes entornos de hardware, software y configuraciones. Se buscará garantizar que el sistema sea compatible con diferentes plataformas y cumpla con los estándares requeridos.

## 1.2 Referencias

Playgruound de Digital House, Materias: Testing I y Testing II

## 1.3 Definiciones y abreviaturas

* QA: Aseguramiento de la Calidad (Quality Assurance, en inglés). Se refiere al conjunto de actividades planificadas y sistemáticas para garantizar que un producto o servicio cumpla con los requisitos de calidad.
* UI: Interfaz de Usuario (User Interface, en inglés). Se refiere a la parte del software o sistema con la que interactúa el usuario.
* UX: Experiencia de Usuario (User Experience, en inglés). Se refiere a cómo percibe y experimenta el usuario al interactuar con un producto o sistema.
* API: Interfaz de Programación de Aplicaciones (Application Programming Interface, en inglés). Es un conjunto de reglas y protocolos que permite la interacción entre diferentes componentes de software.
* DB: Base de Datos (Database, en inglés). Es un conjunto organizado de datos almacenados y accesibles electrónicamente.
* CP: Caso de Prueba
* RD: Reporte de Defectos

# 2. Objetivos del test

## 2.1 Antecedentes

En este Plan de Test, se llevará a cabo el esfuerzo de testing en el entorno de desarrollo de un proyecto de software denominado Digital Money House. El objetivo principal de este proyecto es desarrollar una billetera virtual que proporcione a los usuarios una forma conveniente, segura y eficiente de realizar transacciones financieras y administrar su dinero.

La solución se desarrollará utilizando tecnologías utilizando Vercel y Mongo DB

El proyecto Digital Money House se encuentra en una fase de desarrollo, generando las páginas de Logueo, Registro y Página Principal.. Ha habido un trabajo colaborativo de un equipo multidisciplinario, incluyendo desarrolladores, diseñadores y especialistas en calidad.

## 2.2 Objetivo de la Evaluación

El objetivo del esfuerzo de evaluación en la iteración actual es obtener información sobre la calidad del producto y la satisfacción de los interesados. El enfoque principal será recopilar datos relevantes para evaluar la funcionalidad, el rendimiento y la usabilidad del software, así como para identificar posibles deficiencias o áreas de mejora.

Este objetivo nos permitirá obtener retroalimentación sobre la experiencia del usuario, detectar problemas o errores importantes que puedan afectar la calidad del producto y evaluar el grado en que el software cumple con las expectativas y requisitos establecidos.

Al obtener información sobre la calidad del producto y la satisfacción de los interesados, podremos tomar decisiones informadas sobre los pasos a seguir, priorizar mejoras y ajustes, y garantizar que el producto cumpla con los estándares y las expectativas de calidad establecidos.

## 2.3 Motivaciones del Test

El esfuerzo de testing de esta iteración se ha motivado por varios elementos clave que influyen en la calidad y el éxito del proyecto. Estos elementos son los siguientes:

Riesgos del proyecto: El proyecto en sí puede estar sujeto a riesgos que afecten su éxito, como retrasos en el desarrollo, cambios en los requisitos o limitaciones de recursos. El testing se utilizará para evaluar y comunicar los riesgos del proyecto relacionados con la calidad del software y proporcionar información útil para la toma de decisiones.

Casos de uso y requerimientos: El testing se basará en casos de uso y requerimientos funcionales y no funcionales para verificar que el software cumpla con las expectativas y necesidades de los usuarios. Se evaluará la funcionalidad del software en diferentes escenarios y se verificará que se cumplan los criterios de aceptación definidos.

Elementos de diseño: El testing también se centrará en evaluar los aspectos de diseño del software, como la usabilidad, la accesibilidad y la coherencia visual. Se buscará identificar posibles problemas de diseño que puedan afectar la experiencia del usuario y se propondrán mejoras si es necesario.

Defectos sospechados o fallas: Si existen defectos o fallas conocidos en el software, el testing se centrará en verificar su resolución y asegurarse de que no vuelvan a aparecer. Además, se investigarán y analizarán problemas reportados por los usuarios o identificados en versiones anteriores para evitar su repetición.

Estos elementos principales han motivado el esfuerzo de testing en esta iteración y guiarán el enfoque y la priorización de las actividades de testing para garantizar la calidad del software y la satisfacción de los usuarios.

# 3. ÍTEMS A TESTEAR

En relación al nivel de testing al que se hace referencia en el documento, se utilizarán los siguientes niveles de testing:

Pruebas unitarias, Pruebas de integración y Pruebas de sistema.

A continuación, se proporciona una lista de alto nivel de los ítems principales que serán objeto de test. Esta lista incluye tanto los ítems producidos por el equipo de desarrollo del proyecto:

Componentes desarrollados por el equipo de desarrollo:

Módulos individuales, clases y funciones.

Algoritmos y lógica de negocio.

Interfaces de usuario y elementos de la interfaz gráfica.

Componentes de terceros:

Servicios externos utilizados, como APIs de terceros o servicios en la nube.

Integración con otros sistemas:

Comunicación e interoperabilidad con bases de datos.

Requisitos funcionales y no funcionales:

Requisitos de rendimiento, seguridad, usabilidad y otros aspectos no funcionales.

# 4. MÉTODOS

Para llevar a cabo los tests, se seguirán las siguientes actividades principales:

Planificación de pruebas: Se definirán los objetivos del testing, los alcances, los plazos y los recursos necesarios. También se determinarán las técnicas y herramientas a utilizar, así como los criterios de aceptación y los niveles de cobertura requeridos.

Diseño de casos de prueba: Se desarrollarán los casos de prueba basados en los requisitos funcionales y no funcionales, así como en los escenarios de uso identificados. Se utilizarán técnicas de diseño de pruebas, como la partición de equivalencia, el análisis de valores límite y la combinación de casos, para garantizar una cobertura adecuada.

Preparación del entorno de pruebas: Se configurará un entorno de pruebas adecuado que incluya los sistemas, componentes y datos necesarios para ejecutar las pruebas. Esto puede involucrar la instalación de software, la configuración de bases de datos de prueba y la creación de datos de prueba representativos.

Ejecución de pruebas: Se llevarán a cabo las pruebas según el plan establecido. Se ejecutarán los casos de prueba y se registrarán los resultados obtenidos. Durante esta etapa, se utilizarán técnicas de depuración y herramientas de automatización de pruebas, si están disponibles, para agilizar el proceso y mejorar la eficiencia.

Registro de incidentes: Se registrarán todos los problemas, errores o fallas encontrados durante las pruebas. Se documentarán de manera clara y completa, incluyendo la descripción del problema, los pasos para reproducirlo y cualquier información relevante para su resolución.

Análisis de resultados: Se analizarán los resultados de las pruebas para evaluar la calidad del software y determinar si se cumplen los criterios de aceptación establecidos. Se compararán los resultados obtenidos con las expectativas y se identificarán patrones o tendencias relevantes.

Los criterios de selección de casos de prueba se basarán en varios factores, como la criticidad de las funcionalidades, la probabilidad de fallos, la complejidad de las interacciones y los riesgos identificados. Se seleccionarán casos que cubran una amplia gama de escenarios, incluyendo casos normales, límites y excepcionales, para garantizar una cobertura adecuada.

# 5.CASOS DE PRUEBAS

[Casos de prueba manuales - Equipo 12](https://docs.google.com/spreadsheets/d/1WfovKTi6mBllD9Ex2aZQec7_VcjLC8WjDfVfH4iRtRA/edit#gid=1562412514)

# 6.CRITERIOS DE ACEPTACIÓN

## 6.1 Plan de Test

## 6.1.1 Criterio de Entrada del Plan de Test

El criterio para determinar cuándo la ejecución del Plan de Test puede comenzar será el siguiente:

Todos los documentos y artefactos relevantes, como los requisitos funcionales y no funcionales, el diseño del sistema y los casos de prueba, deben estar completos y aprobados.

El entorno de pruebas debe estar configurado y listo para su uso.

Los recursos necesarios, como el personal de pruebas, las herramientas de testing y los equipos de prueba, deben estar disponibles.

## 6.1.2 Criterio de salida del Plan de Test

El criterio para determinar cuándo la ejecución del Plan de Test está completa o ya no provee ningún beneficio será el siguiente:

Se han ejecutado todos los casos de prueba según lo planificado.

Todos los criterios de aceptación establecidos para los casos de prueba han sido cumplidos.

Se han registrado y resuelto todos los errores críticos identificados durante las pruebas.

Se ha evaluado y aceptado el riesgo restante y se ha determinado que es aceptable para continuar con las siguientes etapas del proyecto.

## 6.1.3 Criterio de suspensión y reanudación

El criterio para determinar si el testing debe ser suspendido o terminado prematuramente antes de que el plan se haya ejecutado totalmente, y bajo qué criterio el testing puede ser reanudado, será el siguiente:

Suspensión: El testing puede ser suspendido si ocurren situaciones como una interrupción del entorno de pruebas, cambios significativos en los requisitos o la arquitectura, falta de recursos críticos o problemas técnicos graves. La decisión de suspender el testing se tomará en base a la evaluación de los impactos y riesgos asociados.

Reanudación: El testing puede ser reanudado una vez que se hayan resuelto las situaciones que llevaron a la suspensión, como la restauración del entorno de pruebas, la estabilización de los requisitos o la obtención de los recursos necesarios. La reanudación del testing se realizará luego de una evaluación de los riesgos y la planificación adecuada de las actividades restantes.

## 6.2 Ciclo de Test

## Criterio de entrada del Ciclo de Test:

El criterio para determinar cuándo el esfuerzo de test para el siguiente Ciclo de Test de este plan puede comenzar será el siguiente:

Se ha completado y aprobado la ejecución del ciclo de test anterior.

Se han realizado las correcciones y ajustes necesarios basados en los resultados y los hallazgos del ciclo de test anterior.

## Criterio de salida del Ciclo de Test:

El criterio para determinar cuándo el esfuerzo de test para el ciclo actual de este plan es considerado suficiente será el siguiente:

Se han ejecutado todos los casos de prueba planificados para el ciclo de test actual.

Se han cumplido los criterios de aceptación establecidos para los casos de prueba.

Se han registrado y resuelto los errores críticos identificados durante las pruebas.

Se ha realizado una evaluación de riesgos y se ha determinado que el nivel de riesgo restante es aceptable.