Digital Money House

**Testing KickOFF**

Criterios de Testing

línea horizontal

# 

# Introducción

Digital Money House es un billetera virtual que tiene como objetivo que los usuarios gestionen su dinero de muchas maneras, como por ejemplo, mediante tarjetas de crédito/débito y transferencias. Este informe describe un plan de pruebas/testing para la aplicación, sus funcionalidades y seguridad.

**¿Cómo escribir un caso de prueba?**

Un caso de prueba debe ser claro, preciso y contener todos los detalles necesarios para que cualquier persona del equipo pueda ejecutarlo. Los elementos básicos incluyen:

1. **Identificador**: Un código único para identificar el caso de prueba (por ejemplo, TC001).
2. **Título**: Una descripción breve del objetivo del caso de prueba.
3. **Precondiciones**: Las condiciones o configuraciones que deben cumplirse antes de ejecutar la prueba.
4. **Pasos a seguir**: Una lista detallada de los pasos necesarios para ejecutar la prueba.
5. **Datos de prueba**: Información o valores específicos necesarios para realizar la prueba.
6. **Resultado esperado**: El comportamiento o salida esperada si el sistema funciona correctamente.
7. **Resultado real**: El comportamiento o salida observada tras ejecutar el caso de prueba (esto se completa durante la ejecución).
8. **Estado**: Resultado de la prueba (por ejemplo, **Aprobado**, **Fallido**, o **Bloqueado**).
9. **Autor**: Persona que construye el caso de prueba.

### ¿Cómo reportar un defecto?

El reporte de un defecto debe incluir toda la información necesaria para reproducir y analizar el problema. Utiliza una herramienta de seguimiento de errores como JIRA. En la carrera Professional Developer utilizamos un archivo tipo PDF para reportar errores.

**Elementos básicos del reporte de defectos:**

1. **Identificador**: Código único del defecto (por ejemplo, BUG001).
2. **Título**: Breve descripción del problema (por ejemplo, "Error al iniciar sesión con credenciales válidas").
3. **Descripción**: Detalles completos del problema, incluyendo qué se esperaba y qué ocurrió.
4. **Pasos para reproducir**: Los pasos necesarios para que el problema se manifieste.
5. **Entorno**: Información sobre el sistema donde ocurrió el problema (navegador, sistema operativo, versión de la aplicación, etc.).
6. **Prioridad/Severidad**: Clasificación del impacto del defecto.
7. **Adjuntos**: Capturas de pantalla, videos, o archivos de log que respalden el defecto.
8. **Estado**: Estado actual del defecto (por ejemplo, **Nuevo**, **En proceso**, **Resuelto**).

**Criterio para incluir un caso de prueba en una suite de humo**

Los casos de prueba de humo deben cumplir con las siguientes condiciones:

1. Relevancia: Verifican las funcionalidades críticas y esenciales del sistema.

2 .Velocidad: Se pueden ejecutar rápidamente.

3. Impacto: Detectan fallos severos que invalidan la ejecución de pruebas más detalladas.

Ejemplo de casos para la suite de humo:

* Inicio de sesión.
* Navegación a páginas principales.
* Realización de una transacción básica.

#### **4. Criterio para incluir un caso de prueba en una suite de regresión**

Los casos de prueba de regresión deben cumplir con las siguientes condiciones:

1. *Cobertura de cambios*: Prueban las funcionalidades afectadas por cambios recientes.

2. *Estabilidad*: Incluyen áreas del sistema que históricamente han presentado defectos.

3. *Impacto funcional*: Validan que las funcionalidades críticas no se vean afectadas.

Ejemplo de casos para la suite de regresión:

* Modificación de datos del usuario tras una actualización en la funcionalidad de edición de perfil.
* Validación del proceso de pago tras un cambio en la integración con el proveedor de pagos**.**

Con respecto al proyecto, en cada sprint se elaboro el siguiente manejo del testing:

**Sprint 1**

Se hizo foco en las funcionalidades relacionadas con el registro de usuarios en la aplicación “Digital Money House”. Los test manuales se realizaron desde el front-end. Las funcionalidades que se probarán son las siguientes:

Registro de usuario, validación de registro por email, inicio de sesión, recuperación de contraseña y cierre de sesión.

**Sprint 2**

Se hizo foco en las primeras funcionalidades desarrolladas en la aplicación “Digital Money House”. Los test manuales se realizaron desde el front-end. Las funcionalidades que se probaron son las siguientes:

Obtener información de la cuenta, registro de tarjeta, eliminación de tarjeta, actualizar alias, ingresar dinero desde una tarjeta asociada a la cuenta

**Sprint 3**

Se hizo foco en las funcionalidades que se van agregando a la aplicación “Digital Money House”. Los test manuales se realizaron desde el front-end. Las funcionalidades que se probarán son las siguientes:

Obtener toda la actividad de la cuenta, obtener el detalle de un movimiento, ingresar dinero a través de una tarjeta asociada a la cuenta

**Sprint 4**

Se hizo foco en las últimas funcionalidades agregadas a la aplicación “Digital Money House”. Los test manuales se realizaron desde el front-end. Las funcionalidades que se probarán son las siguientes:

Transferir dinero desde el saldo disponible en la cuenta