



## Plan de Pruebas - FlowLance

### 1. Contexto del Proyecto

FlowLance es una plataforma en línea diseñada para facilitar la interacción entre freelancers y clientes. Este plan de pruebas ha sido desarrollado para asegurar que las funcionalidades principales del sistema operen según lo esperado y cumplan con los estándares de calidad establecidos. La aplicación se desarrollará utilizando Django para el backend, con PostgreSQL como base de datos, y Selenium se empleará para las pruebas automatizadas.

### 2. Objetivo

El objetivo de este plan de pruebas es diseñar y ejecutar un conjunto de pruebas que cubra las áreas críticas del sistema FlowLance, garantizando que las funcionalidades clave funcionen correctamente y sin errores significativos. Este plan prioriza la detección temprana de defectos y la validación de las interacciones entre los módulos del sistema, asegurando que el producto final cumpla con las expectativas del cliente y los estándares de calidad. Todo el equipo de desarrollo participará en la creación, ejecución y validación de pruebas, integrando estas actividades en el flujo de trabajo diario.

### 3. Esquema de Trabajo

Para nuestro esquema de trabajo, todos los miembros del equipo de desarrollo de FlowLance participarán tanto en la creación de código como en la ejecución de pruebas, con roles flexibles que permiten la colaboración y el intercambio de responsabilidades según las necesidades del proyecto.

#### 3.1 Roles y Responsabilidades

- **Scrum Master (Desarrollador 1):** Además de sus responsabilidades en la coordinación del equipo y la facilitación de ceremonias Scrum, este miembro participará activamente en la creación y ejecución de pruebas, asegurando la correcta integración de las funcionalidades desarrolladas. También supervisará el progreso del plan de pruebas y garantizará que las actividades se alineen con los objetivos del sprint.
- **Product Owner (Desarrollador 2):** Este miembro será responsable de definir y priorizar los casos de prueba, asegurándose de que todas las funcionalidades críticas sean evaluadas. Además, participará en la validación de las pruebas para garantizar que cumplan con los criterios de aceptación, alineándose con las expectativas del cliente y los requisitos del sistema.
- **Ingenieros de Pruebas (Desarrolladores 3, 4 y 5):** Los ingenieros de pruebas, que también forman parte del equipo de desarrollo, se encargarán de



ejecutar los casos de prueba, automatizar las pruebas con Selenium, realizar pruebas manuales, y gestionar la integración continua. Estos roles son flexibles, y los desarrolladores colaborarán activamente entre sí para documentar resultados, reportar defectos, y asegurar que el sistema funcione correctamente en todas sus dimensiones.

## **4. Estrategia de Pruebas**

### **4.1 Desarrollo Guiado por Pruebas (TDD) Adaptado**

El enfoque de pruebas en FlowLance sigue un modelo híbrido que combina TDD (Desarrollo Guiado por Pruebas) con TAD (Test After Development). Durante las primeras etapas del desarrollo, todos los miembros del equipo escriben pruebas unitarias basadas en los casos de uso identificados en las historias de usuario. Conforme avanza el desarrollo y se estabilizan las funcionalidades, se ajustan y amplían las pruebas unitarias para reflejar la implementación final.

### **4.2 Pruebas Unitarias**

Las pruebas unitarias son responsabilidad de todos los desarrolladores, quienes aseguran que cada componente individual del sistema cumpla con su propósito sin introducir errores. Estas pruebas verifican funciones críticas como la autenticación de usuarios, la gestión de perfiles y el manejo de datos en la base de datos PostgreSQL.

### **4.3 Pruebas de Integración**

Las pruebas de integración se centran en verificar la correcta interacción entre los diferentes módulos del sistema. Todos los desarrolladores colaboran en la redacción y ejecución de estas pruebas, asegurando que los módulos del sistema se integren de manera efectiva y que los datos fluyan correctamente entre ellos. Estas pruebas se realizarán antes de la semana 8 para detectar y corregir problemas de integración.

### **4.4 Pruebas Manuales**

Además de las pruebas automatizadas, se llevarán a cabo pruebas manuales enfocadas en la usabilidad y consistencia de la interfaz de usuario. Todo el equipo contribuirá a la revisión de la aplicación en diferentes navegadores y dispositivos, garantizando una experiencia de usuario coherente y accesible.

### **4.5 Pruebas Automatizadas**

Selenium será utilizado por todos los miembros del equipo para automatizar las pruebas de regresión y los flujos de usuario críticos. Estas pruebas se integrarán en el pipeline de CI/CD y se ejecutarán de manera continua, asegurando que cualquier cambio en el código no introduzca errores en el sistema.

### **4.6 Pruebas de Aceptación**



Las pruebas de aceptación se llevarán a cabo en la semana 8, con todo el equipo y el cliente participando en la preparación y ejecución de estas pruebas. El objetivo es validar que FlowLance cumple con todos los requisitos funcionales y que está listo para ser desplegado en producción.

## 5. Planificación y Cronograma

El plan de pruebas se llevará a cabo en las 11 semanas restantes del proyecto, distribuyéndose de la siguiente manera:

### SEMANA ACTIVIDADES

1-2	Revisión de requisitos, diseño de casos de prueba, configuración de entornos de prueba
3-4	Pruebas unitarias y de integración inicial
5-6	Pruebas de integración y automatización con Selenium
7-8	Finalización de pruebas de integración, pruebas de aceptación
9-10	Pruebas de regresión y ajustes finales
11	Documentación de resultados y cierre del proceso de pruebas

## 6. Criterios de Aceptación y Cierre

Para que una funcionalidad sea aceptada, debe pasar con éxito todas las pruebas diseñadas para esa función, cubriendo al menos los escenarios críticos. El ciclo de pruebas se considerará cerrado cuando todos los defectos críticos hayan sido resueltos y la funcionalidad esté documentada y aprobada por el equipo.

## 7. Herramientas de Apoyo

- **Selenium:** Para la automatización de pruebas funcionales y de regresión, enfocadas principalmente en la interfaz de usuario.
- **Postman:** Para la automatización y validación de las API REST, asegurando que las integraciones entre sistemas y los endpoints funcionen correctamente y cumplan con los requisitos.



- **JIRA:** Para la gestión de tareas, seguimiento de defectos y coordinación del equipo durante todo el proceso de desarrollo y pruebas.
- **GitHub Actions:** Para la integración continua y automatización del proceso de pruebas, facilitando la ejecución de pruebas automatizadas en cada cambio de código y asegurando una entrega continua de software de calidad.

## 8. Documentación

La documentación del proceso de pruebas será clave para asegurar un seguimiento detallado y una gestión eficiente de los defectos y resultados. A continuación, se especifican los informes y registros que se generarán durante cada fase de pruebas:

- **Registro de Defectos en JIRA:** Todos los defectos identificados durante las pruebas serán registrados en JIRA. Cada defecto incluirá una descripción detallada, pasos para reproducirlo, el entorno en el que se produjo, la severidad y la prioridad. Los defectos serán asignados a los desarrolladores responsables para su resolución.
- **Informe de Casos de Prueba Ejecutados:** Se generará un informe al finalizar cada sprint que documentará los casos de prueba ejecutados. Este informe incluirá:
  - Escenarios de Configuración: La configuración del escenario para cada módulo de pruebas, como la creación de usuarios de prueba, perfiles o cualquier dato necesario para la ejecución de los casos.
  - Descripción del Caso de Prueba: Una explicación detallada del objetivo de la prueba, las precondiciones, los pasos realizados, los datos de entrada y los resultados esperados.
  - Resultados de las Pruebas: Un resumen de los resultados obtenidos al ejecutar cada caso de prueba, incluyendo si la prueba fue exitosa o fallida. Este informe garantizará que todas las funcionalidades clave hayan sido probadas y que los errores críticos hayan sido abordados antes de finalizar cada sprint.
- **Informe de Pruebas de Integración:** Al finalizar las pruebas de integración, se documentará un informe específico que detalle los resultados de las pruebas de interacción entre módulos, las pruebas de regresión ejecutadas, y cualquier problema de integración resuelto. Este informe será clave para validar la cohesión del sistema antes de las pruebas de aceptación.
- **Informe de Pruebas de Aceptación:** Una vez completadas las pruebas de aceptación en la semana 8, se generará un informe que incluirá los resultados de las pruebas, el cumplimiento de los criterios de aceptación, y la evaluación general de la preparación del sistema para el despliegue en producción. Este



informe será revisado y aprobado por el equipo antes de ser presentado al cliente.

- **Documentación de Cierre de Pruebas:** Al finalizar el ciclo de pruebas, se elaborará un informe de cierre que resumirá todo el proceso de pruebas, incluyendo un resumen de los defectos críticos resueltos, las pruebas realizadas, y la estabilidad final del sistema. Este documento servirá como registro final del estado del producto antes de su lanzamiento.