| PLAN DE PRUEBAS |
| --- |
| **Proyecto:** “Plataforma de Gestión de Servicios y Agenda Profesional” |
|  |
|  |
| Versión*: 1.5* |
| 30/11/2024 |

# Contenido

[**1. Contenido 1**](#_heading=h.d2xiuc4dl1oj)

[**2. Ficha del documento 2**](#_heading=h.gjdgxs)

[2.1. Versionamiento 2](#_heading=h.30j0zll)

[2.2. Integrantes 2](#_heading=h.1fob9te)

[**3. Introducción 3**](#_heading=h.l3jb0xx5l8df)

[3.1. Propósito del Plan de Pruebas 3](#_heading=h.s6no5xv086xf)

[3.2. Alcance 3](#_heading=h.kbkkrj5q0xg7)

[**4. Estrategia de pruebas 4**](#_heading=h.acogqndd2nkv)

[4.1. Tipos de Pruebas 4](#_heading=h.xs4o5cl6xlyz)

[**5. Planificación de Pruebas 5**](#_heading=h.vjx7ecqqwbj6)

[5.1. Casos de Prueba 6](#_heading=h.7kqnrdratd8b)

[**6. Cronograma 7**](#_heading=h.f7zbavzdo0h1)

[**7. Criterios de Aceptación 7**](#_heading=h.aatwfebcv8ij)

[**8. Riesgos y mitigación 8**](#_heading=h.azmse2t1ndf6)

# 

# 

# Ficha del documento

## Versionamiento

| Versión | Fecha | Modificación |
| --- | --- | --- |
| *1.0* | *20/10/2024* | *Creación del documento inicial con el esquema del plan de pruebas.* |
| *1.1* | *21/10/2024* | *Adición del cronograma de pruebas y detalles de planificación inicial.* |
| *1.2* | *22/10/2024* | *Inclusión de casos de prueba específicos y criterios de aceptación.* |
| *1.3* | *25/10/2024* | *Actualización de riesgos identificados y estrategias de mitigación.* |
| *1.4* | *20/11/2024* | *Revisión de resultados de pruebas ejecutadas y ajustes en el cronograma.* |
| *1.5* | *30/11/2024* | *Finalización del documento con métricas de calidad y validación de stakeholders.* |

## Integrantes

| Integrantes del equipo | Rut |
| --- | --- |
| *Jairo Álvarez* | *21.055.328-2* |
| *Francisco López* | *20.883.087-2* |
| *Constanza Painevilo* | *19.561.395-8* |
| *Bastián Rodríguez* | *20.526.431-0* |

**Carrera:** Ingeniería en Informática

**Sede:** San Joaquín

# Introducción

El presente documento detalla el **Plan de Pruebas** diseñado para garantizar la calidad, confiabilidad y funcionalidad de la **Plataforma de Gestión de Servicios y Agenda Profesional**. Este plan abarca una descripción integral de las estrategias, tipos de pruebas, objetivos, alcance y cronograma necesarios para validar que el sistema cumpla con los requisitos establecidos y proporcione una experiencia de usuario excepcional.

En un contexto donde la digitalización de servicios se convierte en un pilar clave para los profesionales independientes y sus clientes, este plan busca establecer un marco claro y eficiente para identificar, prevenir y corregir errores o inconsistencias durante el desarrollo de la plataforma. A través de un enfoque estructurado, se asegurará que todos los componentes del sistema funcionen de manera armónica, cumpliendo tanto con las expectativas del cliente como con los estándares técnicos definidos.

## Propósito del Plan de Pruebas

El propósito principal de este plan es detallar las actividades de pruebas necesarias para validar la calidad, el rendimiento y la usabilidad de los módulos clave de la plataforma. Este plan establece un enfoque metódico para garantizar que:

1. **Funcionalidad:** Todas las funcionalidades descritas en los requisitos estén correctamente implementadas.
2. **Confiabilidad:** El sistema funciona de manera consistente y sin errores bajo diversas condiciones.
3. **Usabilidad:** Los usuarios finales pueden interactuar con la plataforma de manera intuitiva y sin dificultades.
4. **Cumplimiento:** Se cumplen los objetivos del proyecto, alineándose con los estándares de calidad definidos.

## Alcance

El alcance del plan de pruebas incluye los módulos y flujos críticos de la plataforma, asegurando que cada componente sea validado de manera exhaustiva. Los módulos cubiertos son:

1. **Autenticación de Usuarios:** Validación del inicio de sesión, registro y recuperación de contraseñas para garantizar seguridad y accesibilidad.
2. **Publicación de Servicios:** Verificación del correcto funcionamiento del flujo de publicación, edición y eliminación de servicios por parte de los profesionales.
3. **Búsqueda y Filtro de Servicios:** Pruebas en el motor de búsqueda y filtros avanzados para asegurar precisión y rapidez en la localización de servicios.
4. **Reserva de Citas:** Validación de la interacción entre clientes y profesionales, asegurando una gestión fluida de las reservas.
5. **Gestión de Agendas:** Pruebas en el sistema de programación y disponibilidad para garantizar que los profesionales puedan gestionar sus tiempos de manera eficiente.

# Estrategia de pruebas

## Tipos de Pruebas

**Testing Funcional**:

* Verifica que las funcionalidades de la plataforma cumplen con los requisitos definidos.
* Ejemplo: Validar el inicio de sesión, la creación de citas y la gestión de servicios.

**Testing de Usabilidad**:

* Evalúa la experiencia del usuario en términos de diseño, facilidad de uso y accesibilidad.
* Ejemplo: Probar la navegación en dispositivos móviles para garantizar una experiencia fluida.

**Testing de Rendimiento**:

* Mide el tiempo de respuesta, la estabilidad y la capacidad de la plataforma bajo diferentes condiciones de carga.
* Ejemplo: Simular múltiples usuarios accediendo al sistema simultáneamente.

**Testing de Seguridad**:

* Identifica vulnerabilidades en la plataforma y asegura la protección de los datos de los usuarios.
* Ejemplo: Validar la resistencia contra ataques de fuerza bruta en el sistema de autenticación.

**Testing de Compatibilidad**:

* Garantiza que la plataforma funcione correctamente en diferentes dispositivos, sistemas operativos y navegadores.
* Ejemplo: Validar la funcionalidad en Android e iOS.

**Testing de Integración**:

* Verifica que los módulos interactúen correctamente entre sí.
* Ejemplo: Confirmar que la autenticación se integra bien con la gestión de servicios.

**Testing de Regresión**:

* Asegura que las nuevas funcionalidades no afecten las ya existentes.
* Ejemplo: Probar las funciones básicas después de cada actualización del sistema.

**Testing Exploratorio**:

* Descubre defectos inesperados mediante pruebas no estructuradas realizadas por el equipo.
* Ejemplo: Simular comportamientos atípicos de los usuarios.

# Planificación de Pruebas

Las pruebas se desarrollarán de manera paralela al proceso de desarrollo del proyecto, permitiendo identificar y corregir problemas de forma temprana para optimizar la calidad del producto final. A continuación, se detalla el cronograma de actividades de pruebas, alineado con el cronograma general del proyecto:

**Semanas 5 a 8: Testing Funcional e Integración**

* **Objetivo:**Validar que los módulos y funcionalidades individuales cumplan con los requisitos establecidos y que la interacción entre los componentes del sistema sea coherente.
* **Actividades:**
  + Realización de pruebas funcionales en cada módulo (autenticación, publicación de servicios, búsqueda).
  + Pruebas de integración entre frontend y backend, asegurando la correcta transferencia de datos.
  + Detección y corrección de defectos críticos que puedan impactar el flujo principal de la plataforma.

**Semanas 9 a 12: Testing de Usabilidad, Compatibilidad y Seguridad**

* **Objetivo:**Garantizar una experiencia de usuario intuitiva, compatibilidad con diferentes dispositivos y navegadores, y protección de datos sensibles.
* **Actividades:**
  + Pruebas de usabilidad con usuarios simulados para evaluar la facilidad de navegación y comprensión de las funcionalidades.
  + Pruebas de compatibilidad en distintos sistemas operativos, navegadores y dispositivos.
  + Pruebas de seguridad para validar la protección de datos y la resistencia a vulnerabilidades comunes (inyección de SQL, XSS, etc.).

**Semanas 13 a 14: Testing de Rendimiento y Regresión**

* **Objetivo:**Asegurar que la plataforma mantenga un rendimiento óptimo bajo distintas cargas y que las correcciones realizadas no afecten funcionalidades previamente validadas.
* **Actividades:**
  + Pruebas de rendimiento utilizando herramientas como JMeter para simular diferentes volúmenes de tráfico.
  + Pruebas de estrés para evaluar la capacidad del sistema en condiciones extremas.
  + Testing de regresión para verificar que las actualizaciones no introduzcan nuevos defectos en funcionalidades existentes.

**Semana 15: Validación Final y Cierre**

* **Objetivo:**Realizar una revisión exhaustiva del sistema completo para garantizar su preparación para el lanzamiento.
* **Actividades:**
  + Validación final de todos los módulos, asegurando el cumplimiento de los criterios de aceptación.
  + Generación de reportes de pruebas y registro de defectos solucionados.
  + Aprobación del sistema por parte del equipo de QA y stakeholders clave.

## Casos de Prueba

Los casos de prueba están diseñados para validar que la plataforma cumpla con los requisitos funcionales, no funcionales y de seguridad establecidos. Cada caso de prueba se centrará en escenarios clave, asegurando que el sistema funcione según lo esperado y proporcione una experiencia confiable a los usuarios.

**Criterios de Aceptación**

Para considerar las pruebas exitosas, el sistema debe cumplir con los siguientes criterios de aceptación:

1. **Defectos Críticos:**
   * No se permitirá la existencia de defectos críticos en la funcionalidad que afecten los flujos principales del sistema, como autenticación, búsqueda de servicios y gestión de citas.
2. **Tasa de Aprobación de Testing:**
   * La plataforma debe aprobar al menos el **95% de los casos de prueba** ejecutados, garantizando que la mayoría de las funcionalidades estén correctamente implementadas.
3. **Rendimiento del Sistema:**
   * El tiempo de respuesta promedio no debe superar los **2 segundos bajo carga normal**, asegurando una experiencia fluida para los usuarios.
   * En pruebas de estrés, el sistema debe mantener un rendimiento aceptable sin errores fatales.
4. **Seguridad de los Datos:**
   * Las medidas de seguridad implementadas deben garantizar:

* **Protección de datos personales**, cumpliendo con normativas de privacidad (como GDPR o normativas locales aplicables).
* Resistencia a ataques comunes, como inyecciones SQL, cross-site scripting (XSS) y denegación de servicio (DoS).

1. **Usabilidad y Compatibilidad:**
   * El sistema debe ser intuitivo y accesible, proporcionando una navegación fluida en diversos dispositivos y navegadores.
   * Los flujos críticos deben ser fácilmente comprensibles por los usuarios finales sin necesidad de capacitación adicional.

**Notas Adicionales**

* Cualquier defecto identificado durante las pruebas será clasificado según su severidad (crítico, mayor, menor) y deberá ser corregido antes del cierre de la fase correspondiente.
* Los casos de prueba incluirán validaciones tanto positivas como negativas para garantizar la robustez de las funcionalidades.

# Cronograma

El cronograma del plan de pruebas establece una planificación detallada para garantizar que todas las actividades de validación se ejecuten dentro del marco temporal del proyecto. Cada semana está diseñada para abordar un tipo específico de prueba, asegurando una cobertura exhaustiva de los módulos críticos del sistema.

| Semana | Fechas | Actividades |
| --- | --- | --- |
| Semana 1 | 27 de octubre - 2 de noviembre | Testing funcional e integración inicial. |
| Semana 2 | 3 - 9 de noviembre | Testing funcional e integración avanzada. |
| Semana 3 | 10 - 16 de noviembre | Testing de usabilidad, compatibilidad y seguridad. |
| Semana 4 | 17 - 23 de noviembre | Testing de rendimiento y regresión. |
| Semana 5 | 24 - 30 de noviembre | Validación final y cierre. |

# Criterios de Aceptación

El cronograma del plan de pruebas establece una planificación detallada para garantizar que todas las actividades de validación se ejecuten dentro del marco temporal del proyecto. Cada semana está diseñada para abordar un tipo específico de prueba, asegurando una cobertura exhaustiva de los módulos críticos del sistema.

**Criterios Generales**

1. **Cumplimiento de Casos de Prueba**:
   * El 100% de los casos de prueba críticos deben ejecutarse con éxito.
   * Al menos el 95% de los casos de prueba de severidad media y baja deben ser aprobados.
2. **Resolución de Defectos**:
   * Todos los defectos clasificados como **críticos** y **altos** deben estar corregidos y validados.
   * Los defectos de severidad media o baja deben tener un plan de solución definido, siempre que no afecten funcionalidades clave.
3. **Validación del Desempeño**:
   * El sistema debe garantizar un tiempo de respuesta promedio inferior a **2 segundos** bajo carga normal.
   * Soportar al menos **50 usuarios concurrentes** sin afectar la funcionalidad.
4. **Validación de Seguridad**:
   * No deben existir vulnerabilidades críticas en la autenticación, gestión de datos o comunicación del sistema.
   * Los datos sensibles deben estar protegidos mediante cifrado en tránsito y en reposo.
5. **Validación de Usabilidad**:
   * Al menos el 95% de los usuarios finales deben evaluar la interfaz como **intuitiva y funcional**.
   * La navegación debe cumplir con estándares de accesibilidad y experiencia de usuario (UX).
6. **Documentación Completa**:
   * Todos los artefactos relacionados con el plan de pruebas (casos de prueba, resultados, registro de incidencias) deben estar completos y actualizados.
   * Entrega del informe final con métricas de calidad, incidencias detectadas y recomendaciones.
7. **Aprobación de Stakeholders**:
   * Los resultados de las pruebas deben ser revisados y aprobados por el patrocinador del proyecto y los stakeholders clave.

**Margen de Tolerancia**

* Defectos de severidad baja que no afecten la experiencia del usuario o la funcionalidad principal del sistema pueden posponerse para corrección en futuras iteraciones.

Estos criterios aseguran que el sistema cumpla con los estándares requeridos para su aceptación por parte de los stakeholders y su posterior implementación en el entorno productivo.

# Riesgos y mitigación

**Riesgo 1: Retrasos en el Desarrollo**

* **Impacto:** Los retrasos en la entrega de módulos por parte del equipo de desarrollo podrían limitar el tiempo disponible para realizar pruebas exhaustivas.
* **Mitigación:**
  + Implementar un enfoque ágil de pruebas paralelas, iniciando las validaciones en módulos individuales tan pronto como estén disponibles.
  + Realizar reuniones de sincronización semanales entre los equipos de desarrollo y pruebas para identificar y priorizar los módulos más críticos.
  + Asignar un buffer de tiempo en el cronograma para manejar posibles retrasos.

**Riesgo 2: Problemas en la Coordinación entre Equipos**

* **Impacto:** La falta de comunicación efectiva entre los equipos de desarrollo y pruebas puede ocasionar malentendidos y duplicidad de esfuerzos.
* **Mitigación:**
  + Designar un punto de contacto en cada equipo para garantizar la fluidez en la comunicación.
  + Utilizar herramientas colaborativas como Trello o Jira para centralizar la gestión de tareas y seguimiento de defectos.
  + Programar reuniones de revisión conjunta para alinear las prioridades y resolver bloqueos rápidamente.

**Riesgo 3: Defectos Críticos No Detectados**

* **Impacto:** La omisión de defectos críticos durante las pruebas podría impactar negativamente la experiencia del usuario final.
* **Mitigación:**
  + Implementar pruebas exploratorias además de las pruebas planificadas para detectar errores inesperados.
  + Realizar revisiones adicionales en los flujos críticos del sistema.
  + Incorporar revisiones cruzadas entre los miembros del equipo de pruebas para garantizar la exhaustividad.