Plan de Pruebas Inicial - Spintech

**Histórico de Revisiones**

| **Versión** | **Fecha** | **Descripción/cambio** | **autor** |
| --- | --- | --- | --- |
| 1.0 | 10/10/2024 | Versión inicial del plan de pruebas | QA |
| 1.1 | 15/10/2024 | Ajustes en la sección de estrategia de pruebas | QA |

| **Propósito del plan de pruebas**  *Propósito, objetivo, visión que se espera de este plan de pruebas.* |
| --- |
| El propósito del plan de pruebas es asegurar que la Plataforma de Aprendizaje Personalizado para Personas con Discapacidades cumpla con los requisitos y expectativas funcionales y no funcionales establecidas en la fase de diseño y especificación. Este plan de pruebas proporcionará una guía estructurada y detallada para la ejecución de pruebas, con el fin de identificar y corregir posibles defectos antes de su implementación en un entorno de producción.  **Objetivo**  El objetivo del plan de pruebas es validar que cada módulo y funcionalidad de la plataforma funcione correctamente, sea seguro y accesible para todos los usuarios, especialmente para aquellos con discapacidades. Se busca garantizar una experiencia de usuario óptima, verificar que el sistema maneje adecuadamente los datos de los usuarios y validar las integraciones con sistemas de terceros, como pasarelas de pago y servicios de accesibilidad.  **Visión**  Este plan de pruebas está diseñado para asegurar que el sistema sea confiable, eficiente y fácil de usar. La visión de este plan es proporcionar una metodología de pruebas exhaustiva que permita detectar y corregir errores de manera temprana, minimizar riesgos, y ofrecer un sistema de alta calidad que cumpla con los estándares de accesibilidad y seguridad requeridos. Con esto, se espera ofrecer una plataforma inclusiva que facilite el aprendizaje personalizado y garantice una experiencia positiva para todos los usuarios. |

| **Alcance de las pruebas**  *Definición de requisitos de S.W., módulos de Software a probar, Requisitos ambiente de pruebas y Documentación Referenciada, etc.* |
| --- |
| El alcance de este plan de pruebas abarca todos los módulos principales del sistema de la Plataforma de Aprendizaje Personalizado para personas con discapacidades, con el objetivo de asegurar que el sistema funcione de acuerdo con los requisitos y especificaciones definidos.  **Requisitos de Software y Módulos a Probar**   * **Gestión de Usuarios:** Registro, autenticación y administración de permisos para alumnos, profesores. * **Gestión de Cursos:** Creación, modificación y administración de cursos, incluyendo asignación de intérpretes y accesibilidad de contenidos. * **Interfaz de Accesibilidad:** Funcionalidades de accesibilidad, como subtítulos, compatibilidad con dispositivos de asistencia y ajustes de visualización.   **Requisitos de Ambiente de Pruebas**   * **Sistema Operativo:** Windows 10 o superior. * **Navegadores Compatibles:** Google Chrome, Mozilla Firefox y Microsoft Edge en sus versiones actualizadas. * **Herramientas de Pruebas:** Selenium para automatización de pruebas y herramientas manuales de validación de accesibilidad.   **Documentación Referenciada**   * Documento de Especificación de Requerimientos del Software (ERS). * Matriz de Riesgos para la identificación de áreas críticas. * Plan de Gestión de Configuración para versiones del sistema y cambios autorizados. |

| **Definición de roles y responsabilidades**  *Roles y responsabilidades de todos los participantes en el proceso de pruebas de SW.*. | | |
| --- | --- | --- |
| **Rol** | **Responsabilidades** | **Relevancia** |
| Gerente de Proyecto | Supervisión general y aprobación de entregables de pruebas. | Alta |
| QA | Diseño y ejecución de pruebas, reporte de defectos, y cierre de pruebas. | Alta |
| Desarrollador | Implementación de correcciones de defectos y soporte en pruebas unitarias. | Media |

| **Tipos de pruebas a realizar**  *Definir el tipo de pruebas que se debe desarrollar para este proyecto, actividades y responsables.* |
| --- |
| 1. **Pruebas de Unidad**    * **Descripción:** Verificación de cada módulo individual para asegurar que funciona correctamente de forma aislada.    * **Actividades:**      + Pruebas de funciones básicas de cada módulo, como gestión de usuarios y cursos.      + Validación de métodos en el código para asegurar que cada función cumple con los requisitos.    * **Responsable:** Desarrollador (Diego Cortés). 2. **Pruebas de Integración**    * **Descripción:** Verificación de la interacción entre los distintos módulos del sistema, asegurando que funcionan correctamente al combinarse.    * **Actividades:**      + Pruebas de interacción entre el módulo de gestión de cursos y el módulo de usuarios.      + Validación de la integración del sistema de notificaciones y el sistema de accesibilidad.    * **Responsable:** Desarrollador (Diego Cortés). 3. **Pruebas Funcionales**    * **Descripción:** Validación de que el sistema cumple con los requisitos funcionales definidos en el ERS.    * **Actividades:**      + Pruebas de las funcionalidades clave, como la creación de cursos, asignación de intérpretes y gestión de notificaciones.      + Validación de las funcionalidades de accesibilidad, como la activación de subtítulos y ajustes de visualización.    * **Responsable:** QA (Matias Gutiérrez). 4. **Pruebas de Accesibilidad**    * **Descripción:** Validación de las características de accesibilidad de la plataforma, asegurando que personas con discapacidades puedan usar el sistema de manera efectiva.    * **Actividades:**      + Pruebas de compatibilidad con lectores de pantalla y herramientas de accesibilidad.      + Validación de la correcta visualización de subtítulos y opciones de ajustes visuales.    * **Responsable:** QA (Matias Gutiérrez). 5. **Pruebas de Usabilidad**    * **Descripción:** Evaluación de la facilidad de uso de la plataforma, asegurando que sea intuitiva para todos los usuarios, incluidos aquellos con discapacidades.    * **Actividades:**      + Evaluación de la experiencia de usuario en los flujos de registro, inicio de sesión y navegación de cursos.      + Pruebas con usuarios reales para validar la claridad de la interfaz y la accesibilidad de las funcionalidades.    * **Responsable:** QA (Matias Gutiérrez) y Gerente de Proyecto (Bárbara Barraza). 6. **Pruebas de Rendimiento**    * **Descripción:** Validación de que la plataforma maneja adecuadamente la carga de usuarios y responde en tiempos adecuados.    * **Actividades:**      + Pruebas de carga para evaluar el comportamiento de la plataforma con múltiples usuarios simultáneos.      + Pruebas de rendimiento en el tiempo de respuesta de las funciones de accesibilidad y gestión de usuarios.    * **Responsable:** Desarrollador (Diego Cortés). 7. **Pruebas de Seguridad**    * **Descripción:** Verificación de que el sistema protege adecuadamente la información de los usuarios y evita accesos no autorizados.    * **Actividades:**      + Validación de la gestión de contraseñas y autenticación de usuarios.      + Pruebas de acceso no autorizado para asegurar la integridad de la información de los usuarios.    * **Responsable:** QA (Matias Gutiérrez). |

| **Estrategia y técnicas de pruebas a aplicar**  *Definir las estrategias y técnicas de pruebas que se debe desarrollar para este proyecto, actividades y responsables.* |
| --- |
| Estrategia de Pruebas  1. **Pruebas Basadas en Riesgo**  * **Descripción**: Esta estrategia se centrará en identificar y priorizar los componentes críticos del sistema que presentan mayores riesgos. Se le dará prioridad a la prueba de funcionalidades relacionadas con la accesibilidad y la seguridad de datos. * **Actividades**:   + Identificación de áreas de alto riesgo, como el manejo de datos personales y las funciones de accesibilidad.   + Priorización de pruebas para los módulos críticos de la plataforma. * **Responsable**: Gerente de Proyecto (Bárbara Barraza) y QA (Matias Gutiérrez).  1. **Estrategia de Pruebas de Regresión**  * **Descripción**: Esta estrategia asegura que los cambios o actualizaciones en el sistema no afecten funcionalidades ya existentes. Se realizará cada vez que se agreguen nuevas funcionalidades o se corrijan errores. * **Actividades**:   + Ejecución de pruebas regresivas en módulos afectados tras cada actualización o modificación.   + Uso de herramientas de automatización para pruebas repetitivas, especialmente en funcionalidades críticas. * **Responsable**: QA (Matias Gutiérrez) y Desarrollador (Diego Cortés).  1. **Estrategia de Pruebas de Exploración**  * **Descripción**: En esta estrategia, los testers explorarán la plataforma para encontrar defectos inesperados, sobre todo en la interfaz de usuario y en la usabilidad del sistema. * **Actividades**:   + Pruebas exploratorias en los módulos de accesibilidad para verificar la facilidad de uso.   + Revisión de la interfaz y la navegación desde la perspectiva del usuario final. * **Responsable**: QA (Matias Gutiérrez).  2. Técnicas de Pruebas  1. **Técnica de Caja Negra**  * **Descripción**: Esta técnica se aplicará para verificar la funcionalidad del sistema sin conocimiento de la estructura interna. Permite probar las entradas y salidas sin evaluar el código subyacente. * **Actividades**:   + Validación de todas las entradas y salidas de los módulos, como los formularios de registro y la carga de contenidos.   + Pruebas en módulos de accesibilidad y verificación de la salida correcta de datos. * **Responsable**: QA (Matias Gutiérrez).  1. **Técnica de Caja Blanca**  * **Descripción**: Técnica utilizada para probar el código internamente y asegurar que cada camino dentro de una función esté correctamente implementado. * **Actividades**:   + Revisión del flujo de control en las funciones críticas de gestión de usuarios y de accesibilidad.   + Cobertura de caminos para asegurar que cada línea de código funciona como se espera. * **Responsable**: Desarrollador (Diego Cortés).  1. **Técnica de Pruebas de Equivalencia y Análisis de Valor Límite**  * **Descripción**: Técnica usada para reducir el número de casos de prueba, centrando los esfuerzos en entradas de datos representativas y en los límites de las variables. * **Actividades**:   + Creación de casos de prueba de equivalencia en formularios de usuario y acceso.   + Validación de límites para datos como edad, nivel de usuario, y tipo de discapacidad. * **Responsable**: QA (Matias Gutiérrez).  1. **Técnica de Pruebas Automatizadas**  * **Descripción**: Técnica de automatización para pruebas repetitivas, como las pruebas de regresión, con el fin de optimizar el tiempo de prueba y detectar errores de manera eficiente. * **Actividades**:   + Uso de Selenium para automatizar pruebas de funcionalidad en el navegador.   + Automatización de pruebas de regresión para verificar la estabilidad del sistema en cada iteración. * **Responsable**: QA (Matias Gutiérrez). |

| **Definición del proceso de testing**  *Listar y describir todas las actividades a desarrollar en el proceso general de testing, responsables, artefactos, etc.* |
| --- |
| Planificación de las Pruebas  * + **Descripción**: En esta fase, se definen los objetivos y el alcance de las pruebas, el entorno de pruebas, las herramientas necesarias y los criterios de aceptación. También se establecen los tipos de pruebas que se realizarán y se asignan los recursos.   + **Responsables**: Gerente de Proyecto (Bárbara Barraza) y QA (Matias Gutiérrez).   + **Artefactos**: Documento del plan de pruebas inicial, matriz de requerimientos de prueba.  Diseño de Casos de Prueba  * + **Descripción**: Se diseñan los casos de prueba basados en los requisitos funcionales y no funcionales del sistema. Los casos de prueba incluirán las condiciones iniciales, pasos a seguir, datos de prueba y resultados esperados.   + **Responsables**: QA (Matias Gutiérrez).   + **Artefactos**: Casos de prueba documentados, planilla de casos de prueba, matriz de trazabilidad entre requerimientos y casos de prueba.  Configuración del Entorno de Pruebas  * + **Descripción**: Preparación del entorno en el que se ejecutarán las pruebas, que incluye la instalación de software necesario, configuración de base de datos y definición de los perfiles de usuario. Se verifica que el entorno de pruebas sea representativo del entorno de producción.   + **Responsables**: Desarrollador (Diego Cortés) y QA (Matias Gutiérrez).   + **Artefactos**: Documentación de la configuración del entorno de pruebas, lista de verificación de configuración.  Ejecución de las Pruebas  * + **Descripción**: En esta fase, se ejecutan los casos de prueba diseñados. Incluye la ejecución de pruebas funcionales, pruebas de accesibilidad, pruebas de seguridad y pruebas de regresión.   + **Responsables**: QA (Matias Gutiérrez).   + **Artefactos**: Resultados de los casos de prueba, reporte de errores, capturas de pantalla o evidencias de pruebas fallidas.  Reporte de Defectos  * + **Descripción**: Registro de todos los defectos encontrados durante la ejecución de las pruebas. Cada defecto incluye una descripción, pasos para reproducir, severidad, y la prioridad de resolución.   + **Responsables**: QA (Matias Gutiérrez).   + **Artefactos**: Documento de registro de defectos, tabla de trazabilidad de defectos.  Corrección y Re-pruebas  * + **Descripción**: El desarrollador corrige los defectos reportados y el equipo de QA vuelve a ejecutar los casos de prueba para verificar que los errores hayan sido resueltos y que no se hayan introducido nuevos problemas.   + **Responsables**: Desarrollador (Diego Cortés) y QA (Matias Gutiérrez).   + **Artefactos**: Registro de re-pruebas, reporte de resultados de re-pruebas.  Pruebas de Regresión  * + **Descripción**: Después de corregir los errores, se realizan pruebas de regresión para garantizar que las modificaciones no hayan afectado otras funcionalidades del sistema. Estas pruebas se ejecutan tanto manualmente como con scripts automatizados.   + **Responsables**: QA (Matias Gutiérrez).   + **Artefactos**: Resultados de pruebas de regresión, reporte de impacto de las correcciones.  Validación de Criterios de Aceptación  * + **Descripción**: Revisión final de los criterios de aceptación de los requerimientos para asegurarse de que todas las funciones cumplen con los estándares de calidad y accesibilidad definidos en el proyecto.   + **Responsables**: QA (Matias Gutiérrez) y Gerente de Proyecto (Bárbara Barraza).   + **Artefactos**: Documento de validación de criterios de aceptación, lista de verificación de criterios cumplidos.  Reporte de Cierre de Pruebas  * + **Descripción**: Se elabora un informe final que documenta los resultados de todas las pruebas realizadas, el estado de los defectos y las recomendaciones finales. Se analiza el cumplimiento de los objetivos y se documentan las lecciones aprendidas.   + **Responsables**: QA (Matias Gutiérrez) y Gerente de Proyecto (Bárbara Barraza).   + **Artefactos**: Informe de cierre de pruebas, documento de lecciones aprendidas. |

| **Requerimientos del test**  *Describir cual es el ambiente de prueba necesario para ejecutar el testing y quien será el responsable de la preparación del ambiente, así como también de la entrega del S.W. a QA* |
| --- |
| Ambiente de Prueba Necesario  1. **Servidor de Aplicaciones**:    * **Configuración**: Instalación de un servidor de aplicaciones en un entorno similar al de producción (por ejemplo, servidor Apache con XAMPP).    * **Propósito**: Asegurar que el comportamiento de la plataforma sea similar al entorno final en términos de velocidad y capacidad de respuesta. 2. **Base de Datos**:    * **Configuración**: Instalación de una base de datos MySQL en el entorno de pruebas.    * **Propósito**: Validar que la plataforma pueda gestionar datos en un entorno que represente la infraestructura de producción. 3. **Software de Pruebas Automatizadas**:    * **Herramientas**: Integración de Selenium para realizar pruebas automatizadas de regresión y validación de interfaz de usuario.    * **Propósito**: Garantizar la consistencia y accesibilidad de las funciones y asegurar que no se produzcan errores recurrentes después de las actualizaciones. 4. **Configuración de Seguridad y Accesibilidad**:    * **Configuración**: Configuración de pruebas de seguridad en el entorno para validar que los datos personales y sensibles estén protegidos.    * **Propósito**: Asegurar que el sistema cumple con los requisitos de protección de datos y accesibilidad para usuarios con discapacidades. 5. **Equipos de Prueba Múltiples**:    * **Configuración**: Acceso a diferentes dispositivos (computadoras, tablets, dispositivos móviles) y navegadores (Chrome, Firefox, Edge) para realizar pruebas de compatibilidad.    * **Propósito**: Verificar que la plataforma funcione correctamente en diferentes dispositivos y sistemas operativos, asegurando accesibilidad para todos los usuarios.   **Responsable de la Preparación del Ambiente**   * **Responsable:** El Desarrollador (Diego Cortés) será el encargado de configurar el ambiente de pruebas, incluyendo la instalación y configuración de la base de datos y el servidor de aplicaciones. * **Entrega del Software:** Diego Cortés también será el responsable de entregar la versión preliminar del software a QA para su revisión. El proceso incluye realizar verificaciones básicas para confirmar que el entorno de pruebas está completamente funcional antes de la ejecución de pruebas de QA. * **Revisión del Ambiente de Pruebas:** El equipo de QA (Matias Gutiérrez) revisará el entorno de pruebas una vez esté configurado para asegurar que cumple con los requisitos necesarios antes de comenzar la ejecución de las pruebas. |

| **Definición de ciclos de prueba a ejecutar**  *Listar y describir cantidad de ciclos de prueba a realizar en este proyecto, las tareas y actividades para cada ciclo de prueba, responsables, artefactos, etc.* |
| --- |
| Ciclo 1: Pruebas de Unidad  * **Descripción**: En este ciclo se evaluarán individualmente cada uno de los módulos y componentes del sistema, asegurando que cada unidad de código funcione correctamente y cumpla con los requisitos funcionales. * **Tareas y Actividades**:   + Realización de pruebas de unidad en cada módulo clave, como el módulo de gestión de usuarios, módulo de accesibilidad, y módulo de contenidos.   + Identificación y corrección de errores en funciones específicas antes de proceder a una integración más amplia. * **Responsable**: Desarrollador (Diego Cortés) * **Artefactos Generados**:   + Informes de prueba de unidad con resultados detallados de cada módulo.   + Registro de errores y lista de correcciones realizadas.  Ciclo 2: Pruebas de Integración  * **Descripción**: Este ciclo asegura que los módulos funcionan correctamente al integrarse entre sí y que la comunicación entre ellos es eficiente y libre de errores. * **Tareas y Actividades**:   + Ejecución de pruebas de integración que validen la interacción entre módulos como gestión de usuarios, accesibilidad, y presentación de contenido.   + Verificación de la base de datos y conexiones a servidores para asegurar la correcta recuperación y almacenamiento de datos. * **Responsable**: Desarrollador (Diego Cortés) y QA (Matias Gutiérrez) * **Artefactos Generados**:   + Informes de prueba de integración con detalles sobre la interacción de los módulos.   + Documentación de errores de integración y modificaciones aplicadas para mejorar la compatibilidad entre componentes.  Ciclo 3: Pruebas de Sistema y Aceptación  * **Descripción**: En esta fase final, se realizan pruebas completas del sistema para garantizar que todas las funciones del proyecto cumplen con los requisitos y que la plataforma está lista para su implementación. * **Tareas y Actividades**:   + Pruebas de accesibilidad y usabilidad, asegurando que el sistema es completamente accesible para personas con discapacidades.   + Ejecución de pruebas de seguridad, rendimiento, y compatibilidad para validar que el sistema puede manejar diferentes escenarios de uso y que los datos están protegidos.   + Realización de pruebas de aceptación con usuarios finales simulados para obtener retroalimentación y asegurar que el sistema satisface los requisitos. * **Responsable**: QA (Matias Gutiérrez) * **Artefactos Generados**:   + Informe de pruebas de sistema con resultados de accesibilidad, rendimiento y seguridad.   + Registro de feedback de usuarios y ajuste de funcionalidades según los comentarios recibidos.   + Informe de aceptación con conclusiones sobre la preparación del sistema para su despliegue.  Resumen de Ciclos de Prueba  1. **Ciclo 1**: Pruebas de Unidad (Desarrollador) 2. **Ciclo 2**: Pruebas de Integración (Desarrollador y QA) 3. **Ciclo 3**: Pruebas de Sistema y Aceptación (QA) |

| **Calendarización de las actividades de pruebas**  *Listado de actividades, tareas, duración, fechas, responsables, etc.* |
| --- |
| **Actividades y Tareas de Prueba**   1. **Preparación del Entorno de Pruebas**    * **Tareas:** Configuración del entorno, instalación de herramientas, creación de usuarios de prueba.    * **Duración:** 2 días    * **Fechas:** 10-11 de septiembre de 2024    * **Responsable:** QA (Matias Gutiérrez) 2. **Ciclo 1: Pruebas de Unidad**    * **Tareas:** Pruebas de cada módulo (gestión de usuarios, accesibilidad, contenido).    * **Duración:** 1 semana    * **Fechas:** 12-18 de septiembre de 2024    * **Responsable:** Desarrollador (Diego Cortés) 3. **Revisión y Corrección de Errores de Unidad**    * **Tareas:** Identificación de errores, ajustes en el código y pruebas de verificación.    * **Duración:** 3 días    * **Fechas:** 19-21 de septiembre de 2024    * **Responsable:** Desarrollador (Diego Cortés) 4. **Ciclo 2: Pruebas de Integración**    * **Tareas:** Pruebas de interacción entre módulos (usuarios, accesibilidad, presentación de contenido).    * **Duración:** 1 semana    * **Fechas:** 22-28 de septiembre de 2024    * **Responsables:** Desarrollador (Diego Cortés) y QA (Matias Gutiérrez) 5. **Revisión y Corrección de Errores de Integración**    * **Tareas:** Ajustes en comunicación entre módulos y pruebas de verificación de integración.    * **Duración:** 3 días    * **Fechas:** 29 de septiembre - 1 de octubre de 2024    * **Responsable:** QA (Matias Gutiérrez) 6. **Ciclo 3: Pruebas de Sistema y Aceptación**    * **Tareas:** Pruebas de sistema completas, incluidas pruebas de accesibilidad, seguridad y rendimiento; pruebas de aceptación de usuarios simulados.    * **Duración:** 2 semanas    * **Fechas:** 2-15 de octubre de 2024    * **Responsable:** QA (Matias Gutiérrez) 7. **Revisión Final y Ajustes de Sistema**    * **Tareas:** Implementación de feedback, ajuste final de funcionalidades y optimización de rendimiento.    * **Duración:** 1 semana    * **Fechas:** 16-22 de octubre de 2024    * **Responsables:** Desarrollador (Diego Cortés) y QA (Matias Gutiérrez) 8. **Informe Final de Pruebas**    * **Tareas:** Consolidación de resultados, documentación de pruebas, conclusiones y recomendaciones.    * **Duración:** 3 días    * **Fechas:** 23-25 de octubre de 2024    * **Responsable:** QA (Matias Gutiérrez)  | **Actividad** | **Duración** | **Fechas** | **Responsable** | | --- | --- | --- | --- | | Preparación del Entorno de Pruebas | 2 días | 10-11 de septiembre | Matias Gutiérrez (QA) | | Pruebas de Unidad | 1 semana | 12-18 de septiembre | Diego Cortés (Desarrollador) | | Pruebas de Unidad | 3 días | 19-21 de septiembre | Diego Cortés (Desarrollador) | | Pruebas de Integración | 1 semana | 22-28 de septiembre | Diego Cortés (Desarrollador), Matias Gutiérrez (QA) | | Corrección de Errores de Integración | 3 días | 29 de septiembre - 1 de octubre | Matias Gutiérrez (QA) | | Pruebas de Sistema y Aceptación | 2 semanas | 2-15 de octubre | Matias Gutiérrez (QA) | | Revisión Final y Ajustes | 1 semana | 16-22 de octubre | Diego Cortés (Desarrollador), Matias Gutiérrez (QA) | | Informe Final de Pruebas | 3 días | 23-25 de octubre | Matias Gutiérrez (QA) | |
|  |

| **Resumen de riesgos**  *Listado de riesgos relacionados al proceso de pruebas de S.W. Indicar riesgo, magnitud o impacto de este riesgo por etapa en el proceso. Magnitud: Alto, Significativo , Moderado, Inferior y Baja.* | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Magnitud** | **Fase del proceso de pruebas** | | | | | **Riesgo** |
| **Planificación** | **Análisis y diseño** | **Implementación y ejecución** | **Evaluación** | **Cierre** |
| **Alto** | Falta de recursos adecuados | Requisitos ambiguos | Fallas en herramientas de prueba | Inconsistencias en resultados | Errores críticos en documentación | La falta de recursos humanos o tecnológicos podría retrasar el proceso de pruebas |
| **Significativo** | Retraso en la planificación | Cambios en los requisitos | Integración fallida de módulos | Retrasos en corrección de errores | Actualización inadecuada del reporte | Cambios de requisitos en la fase de diseño afectan la implementación y ejecución de pruebas |
| **Moderado** | Demora en la definición del cronograma | Falta de documentación precisa | Errores no detectados en la prueba inicial | Ausencia de pruebas de regresión | Validación insuficiente de datos finales | Riesgo de errores no detectados durante la prueba inicial |
| **Bajo** | Inadecuada asignación de roles | Comunicación limitada en equipo | Resultados de pruebas inexactos | Falta de métricas de desempeño | Ausencia de respaldo de información | Asignación de roles incorrecta afecta la ejecución y precisión de las pruebas |
| **Inferior** | Baja participación en reuniones | Entendimiento limitado de requisitos | Resultados de prueba no almacenados | Omisión de pasos en pruebas | Falta de revisión final adecuada | Baja asistencia a reuniones afecta la coordinación y los resultados del proceso de pruebas |

| **Definir clasificación de los defectos**  *Definir la clasificación de los defectos según su nivel de severidad* | |
| --- | --- |
| **Nivel de Severidad** | **Descripción** |
| **Crítico** | Defectos que impiden el funcionamiento total del sistema o de un módulo clave, afectando a todos los usuarios. Requiere una solución inmediata. |
| **Alto** | Defectos que afectan la funcionalidad principal del sistema pero permiten su uso limitado. Los usuarios pueden acceder parcialmente al sistema, pero con limitaciones importantes. |
| **Moderado** | Defectos que afectan la funcionalidad secundaria o adicional del sistema, sin interrumpir el uso general. Estos problemas deben resolverse, pero no son urgentes. |
| **Bajo** | Defectos menores que no afectan el funcionamiento del sistema, pero que pueden generar molestias o confusión en el usuario. Su resolución es opcional. |
| **Cosmético** | Defectos que no afectan la funcionalidad ni el uso del sistema, pero que tienen que ver con detalles visuales o estéticos (ej. errores en la interfaz de usuario). |

| **Definición de artefactos**  *Listar y describir los artefactos que serán administrados y entregados durante este proceso de prueba.* | |
| --- | --- |
| **Artefacto** | **Descripción** |
| **Casos de Prueba** | Documentos que detallan cada prueba específica a realizar, incluyendo pasos, entradas, condiciones y los resultados esperados para evaluar el funcionamiento del sistema. |
| **Plan de Pruebas** | Documento inicial que define el alcance, enfoque, recursos y cronograma de las actividades de prueba, proporcionando una visión general del proceso de testing. |
| **Registro de Defectos** | Lista donde se documentan todos los defectos encontrados durante el proceso de prueba, incluyendo detalles sobre su descripción, severidad, estado y resolución. |
| **Reporte de Pruebas** | Informe detallado sobre los resultados de cada ciclo de pruebas, incluyendo casos exitosos y fallidos, métricas de calidad y análisis de la estabilidad del sistema. |
| **Evidencias de Prueba** | Capturas de pantalla, registros y archivos que respaldan la ejecución y resultados de las pruebas realizadas para verificar el cumplimiento de los requisitos. |
| **Matriz de Trazabilidad** | Documento que relaciona los casos de prueba con los requisitos del sistema para asegurar que todos los requisitos sean verificados en el proceso de pruebas. |

| **Condiciones de aceptación para cierre del proceso de pruebas**  *Condiciones que se deben cumplir para dar término al proceso de pruebas y margen de tolerancia de aceptación de defectos.* |
| --- |
| 1. **Cumplimiento de los Casos de Prueba:** Todos los casos de prueba críticos y de alta prioridad deben haberse ejecutado con éxito, cumpliendo los requisitos funcionales y no funcionales del sistema. 2. **Margen de Tolerancia de Defectos:** Se acepta un margen de defectos de baja severidad que no afecten la funcionalidad principal del sistema. Los defectos críticos y de alta severidad deben ser corregidos y verificados antes del cierre del proceso. 3. **Validación de Correcciones:** Todos los defectos reportados y resueltos durante el proceso de pruebas deben haber pasado por un ciclo de verificación exitoso para asegurar que no han reintroducido fallas en el sistema. 4. **Conformidad con los requisitos:** Se debe haber verificado que el sistema cumple con todos los requisitos especificados en el documento de requisitos de software (ERS). 5. **Evidencia Documentada:** Todos los reportes de pruebas, resultados, casos de prueba y evidencias de defectos resueltos deben estar debidamente documentados y archivados para una futura referencia y auditoría. 6. **Aprobación del Equipo de QA:** El equipo de QA debe confirmar y aprobar el cumplimiento de los estándares de calidad definidos y verificar que no existen defectos críticos pendientes. |