**Plan de Pruebas de Software**

***[Nombre del Proyecto]***

***Fecha: [dd/mm/aaa]***

**Tabla de contenido**

Historial de Versiones 4

Información del Proyecto 4

Aprobaciones 4

Resumen Ejecutivo 5

Alcance de las Pruebas 5

Elementos de Pruebas 5

Nuevas Funcionalidades a Probar 6

Pruebas de Regresión 6

Funcionalidades a No Probar 7

Enfoque de Pruebas (Estrategia) 7

Criterios de Aceptación o Rechazo 8

Criterios de Aceptación o Rechazo 8

Criterios de Suspensión 8

Criterios de Reanudación 9

Entregables 9

Recursos 10

Requerimientos de Entornos – Hardware 10

Requerimientos de Entornos – Software 10

Herramientas de Pruebas Requeridas 11

Personal 11

Entrenamiento 12

Planificación y Organización 12

Procedimientos para las Pruebas 12

Matriz de Responsabilidades 13

Cronograma 13

Premisas 14

Dependencias y Riesgos 14

**Historial de Versiones**

| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Información del Proyecto**

| Empresa / Organización |  |
| --- | --- |
| Proyecto | Desarrollo plataforma de gestión de reemplazo para profesores de Duoc |
| Fecha de preparación |  |
| Cliente |  |
| Patrocinador principal |  |
| Gerente / Líder de Proyecto |  |
| Gerente / Líder de Pruebas de Software |  |

**Aprobaciones**

| **Nombre y Apellido** | **Cargo** | **Departamento u Organización** | **Fecha** | **Firma** |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Resumen Ejecutivo**

Resumen de todo el contenido del plan de Pruebas de Software, describe cuál es su propósito, establece si es un plan maestro o un plan detallado, identifica el alcance del plan de pruebas en relación con el plan de Proyecto de Software, restricciones (por ejemplo de recursos o presupuesto), alcance del esfuerzo de pruebas entre otros aspectos.

**Alcance de las Pruebas**

Las pruebas cubrirán todos los módulos funcionales del software, con un enfoque específico en las funcionalidades nuevas y modificadas, así como en las interacciones entre los diferentes componentes del sistema. El alcance incluye:

Pruebas funcionales para asegurar que el sistema cumpla con los requisitos establecidos.

Pruebas de rendimiento, escalabilidad y carga, especialmente para asegurar que el sistema puede manejar el volumen de usuarios y datos previstos.

**Elementos de Pruebas**

Las pruebas se realizarán en los siguientes elementos del sistema:

Módulos de gestión de usuarios (registro, login, y gestión de permisos).

Módulos de funcionalidad docente (creación de perfiles, asignación de clases y horarios).

Módulos de reportes (generación y descarga de reportes en formatos PDF y Excel).

Módulos de cálculo de horas y bonos (cálculo de horas de reemplazo y bonos acumulativos).

Interfaz de usuario para asegurar que sea intuitiva y fácil de usar.

**Nuevas Funcionalidades a Probar**

Desde el punto de vista del usuario, las siguientes funcionalidades son nuevas o modificadas y deben ser probadas:

Creación de perfil de docente: Se verificará que el administrador pueda crear perfiles correctamente, con la capacidad de agregar horarios, asignar clases y gestionar licencias.

Cálculo y clasificación de horas trabajadas: Se probará la correcta clasificación de las horas trabajadas como horas de programación o bono.

Generación de reportes: Se validará la correcta descarga y visualización de los reportes de horas trabajadas y clases de reemplazo.

**Pruebas de Regresión**

Las funcionalidades no directamente involucradas en las modificaciones del sistema pero que podrían verse afectadas incluyen:

Módulos de autenticación: Validar que el acceso al sistema y las validaciones de permisos siguen funcionando correctamente.

**Funcionalidades a No Probar**

Las siguientes funcionalidades no se van a probar durante este ciclo de pruebas:

Módulos de facturación y contabilidad: No forman parte de las funcionalidades del proyecto actual y no se verán afectadas por los cambios realizados.

Integración con otros sistemas externos: No se realizará pruebas de integración con sistemas no directamente relacionados.

Riesgos asumidos: La no inclusión de estas funcionalidades implica el riesgo de que, si hay cambios posteriores, puedan afectarse sin ser detectados previamente.

**Enfoque de Pruebas (Estrategia)**

El enfoque de pruebas se centrará en las siguientes áreas:

Pruebas funcionales: Se realizarán pruebas unitarias, de integración y de sistema para verificar que las nuevas funcionalidades cumplen con los requisitos.

Pruebas de rendimiento: Se realizará una prueba de carga y estrés para evaluar el comportamiento del sistema bajo condiciones extremas.

Pruebas de seguridad: Evaluación de la protección contra accesos no autorizados.

Pruebas de compatibilidad: Verificación de la funcionalidad del sistema en diferentes navegadores y dispositivos.

**Criterios de Aceptación o Rechazo**

**Criterios de Aceptación o Rechazo**

Los criterios para aceptar las pruebas incluyen:

100% de las pruebas unitarias completadas exitosamente.

Al menos un 95% de casos de prueba exitosos.

Cobertura de código de al menos el 90%.

Todos los defectos críticos y de alta prioridad corregidos.

**Criterios de Suspensión**

Las pruebas se suspenderán si:

Se encuentran defectos críticos que impidan la ejecución de más pruebas.

El número de casos de prueba fallidos supera el 25%.

Existen errores en la infraestructura de pruebas que no pueden resolverse rápidamente.

**Criterios de Reanudación**

Las pruebas se reanudarán cuando:

Los defectos críticos hayan sido corregidos y verificados.

La infraestructura de pruebas esté operativa nuevamente.

El número de casos fallidos haya sido reducido a un nivel aceptable.

**Entregables**

Los entregables del plan de pruebas incluyen:

Documento del Plan de Pruebas.

Casos de prueba.

Logs de errores.

Reportes de incidencias.

Evidencias de pruebas realizadas.

**Recursos**

**Requerimientos de Entornos – Hardware**

Servidores de aplicación, bases de datos y redes necesarias para la prueba del sistema.

Equipos de cómputo para los testers, con conectividad a la red.

**Requerimientos de Entornos – Software**

Acceso al sistema de pruebas y a las bases de datos en entornos de desarrollo y pre-producción.

Instalación de herramientas de prueba como Selenium.

**Herramientas de Pruebas Requeridas**

Herramientas de automatización: Selenium

Herramientas de gestión de pruebas: Jira.

**Personal**

Líder de pruebas (1).

Analistas de pruebas (5).

Especialistas en automatización de pruebas (2).

**Entrenamiento**

El equipo de pruebas recibirá formación en el sistema y las herramientas de prueba necesarias (Selenium).

**Planificación y Organización**

**Procedimientos para las Pruebas**

Siguiendo la metodología en cascada, las pruebas se llevarán a cabo de manera secuencial, después de la fase de desarrollo del sistema. Las pruebas están alineadas con la fase de entrega y validación final del sistema, y se llevarán a cabo en bloques definidos después de cada fase del proyecto.

Pruebas unitarias: Aseguran que cada módulo desarrollado funcione correctamente de forma aislada.

Pruebas de integración: Verifican la interacción entre los diferentes módulos del sistema.

Pruebas del sistema: Evaluación global de todas las funcionalidades del sistema y su comportamiento como un conjunto integrado.

Pruebas de aceptación: Validación final de que el sistema cumple con los requisitos especificados por el cliente y el proyecto.

**Matriz de Responsabilidades**

Lista cada una de las personas integrantes del equipo de QA y sus responsabilidades. Se puede hacer uso de una Matriz RACI (Responsable, Aprobador, Consultado, Informado).

**Cronograma**

Debe estar basado en estimaciones de actividades realizadas por el equipo de prueba. En él se Identifican los hitos relevantes en las pruebas de software, se establecen las dependencias (actividades predecesoras) y demás aspectos componentes de un cronograma.

**Premisas**

El software estará en su versión estable y final después de completar cada fase de desarrollo.

Todos los defectos críticos y de alta prioridad identificados en la fase anterior deben ser corregidos antes de continuar con las pruebas siguientes.

No se pasarán a nuevas fases hasta que los resultados de la fase de pruebas actual hayan sido validados.

**Dependencias y Riesgos**

Aquí se listan los riesgos asociados con el proceso de pruebas de software, por ejemplo, algunas fuentes de riesgos suelen ser:

* Dependencias con Desarrollos.
* Dependencias con otros proyectos.
* Disponibilidad de recursos.
* Restricciones de tiempo.
* Premisas que resulten no ser ciertas.

Los riesgos se pueden clasificar en función de su probabilidad e impacto, cada uno debe contemplar un plan de mitigación para evitar que ocurra o plan de contingencia cuando el riesgo no puede mitigarse y tiene que aceptarse.

Dependencia de la disponibilidad de datos de prueba: Algunas funcionalidades del sistema pueden requerir datos específicos para ser probadas, lo que puede retrasar las pruebas si esos datos no están listos.

Riesgo de recursos limitados: La falta de testers o recursos para ejecutar las pruebas puede atrasar el cronograma establecido.

Riesgo de cambios en los requisitos: Si los requisitos no están completamente definidos al inicio, cualquier cambio a mitad del proceso puede afectar las fases posteriores.