**Plan de Pruebas de Software**

***[JobConnect]***

***Fecha: [07/10/2024]***

**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

**Historial de Versiones**

| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| --- | --- | --- | --- | --- |
| 07-10-2024 | 01 | CA | JobConnect |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Información del Proyecto**

| Empresa / Organización | JobConnect |
| --- | --- |
| Proyecto | JobConnect |
| Fecha de preparación | 22/08/2024 |
| Cliente | JobConnect |
| Patrocinador principal | Gerente operacional de JobConnect |
| Gerente / Líder de Proyecto | Felipe Urbina |
| Gerente / Líder de Pruebas de Software | Sebastian Duran |

**Aprobaciones**

| **Nombre y Apellido** | **Cargo** | **Departamento u Organización** | **Fecha** | **Firma** |
| --- | --- | --- | --- | --- |
| Felipe Urbina | Líder del proyecto | JobConnect | 07/10/2024 | asdf |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**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.

Este plan de pruebas tiene como objetivo garantizar la calidad de una aplicación web que facilita la creación y solicitud de trabajos básicos, como jardinería o gasfitería. Es un plan detallado que incluye pruebas funcionales y no funcionales, alineado con el plan general del proyecto de software. El alcance cubre funcionalidades clave como la creación de perfiles de usuarios, búsqueda de trabajos y el sistema de chat en tiempo real. Aunque existen restricciones de presupuesto y recursos, el esfuerzo de pruebas se centrará en los casos críticos, utilizando tanto pruebas manuales como automatizadas para asegurar una cobertura adecuada.

**Alcance de las Pruebas**

**Elementos de Pruebas**

Listado de todos los módulos, componentes o elementos que se van a probar. Si es de alto nivel, se listan las áreas funcionales (módulos o procesos que cubre el Testing), por otro lado, si es de un nivel detallado se listan los programas, unidades o módulos.

Se probarán los siguientes módulos y componentes de la aplicación web:

* **Gestión de Usuarios**: Creación y modificación de perfiles para trabajadores y empleadores.
* **Búsqueda de Trabajos**: Funcionalidad de búsqueda y filtrado de ofertas laborales.
* **Sistema de Chat**: Interacción en tiempo real entre usuarios.
* **Calificaciones y Reseñas**: Implementación del sistema de feedback.

**Nuevas Funcionalidades a Probar**

Es un listado de lo que se va a probar “Desde el Punto de vista del Usuario”. No es una descripción técnica del software sino sus características y funcionalidades. Se incluyen tanto las que son nuevas como las que se están modificando.

Desde el punto de vista del usuario, se incluirán las siguientes características:

* Creación y gestión de perfiles de usuario.
* Solicitud y creación de trabajos básicos.
* Funcionalidad de chat en tiempo real.
* Sistema de calificaciones y reseñas para mejorar la confianza en la plataforma.
* Interfaz de usuario rediseñada para una navegación más intuitiva.

**Pruebas de Regresión**

Listado de las funcionalidades no directamente involucradas en el desarrollo, pero cuyos componentes están siendo afectados y por ende deben probarse para asegurar que continúan funcionando adecuadamente. Al igual que en el punto anterior, se describen desde el punto de vista del usuario.

Se llevarán a cabo pruebas en las siguientes funcionalidades no directamente involucradas en el desarrollo, pero que podrían verse afectadas:

* **Notificaciones**: Asegurar que las alertas sobre nuevos trabajos y mensajes sigan funcionando correctamente.
* **Configuración de Perfil**: Confirmar que las opciones de configuración del perfil no se vean afectadas por los nuevos cambios.

**Funcionalidades a No Probar**

Listado de las funcionalidades que NO se van a probar. Debe incluir información de las razones por las cuales no se van a probar y los riesgos que se están asumiendo.

Se excluyen las siguientes funcionalidades del proceso de pruebas:

* Integraciones con terceros (como plataformas de pago) que se implementarán en una fase futura.
* Funcionalidades de análisis y estadísticas de uso de la plataforma.  
  Las razones para no probar estas funcionalidades incluyen limitaciones de tiempo y recursos, y se asumen riesgos en cuanto a la estabilidad de estas áreas.

**Enfoque de Pruebas (Estrategia)**

La Estrategia de Pruebas puede definirse como un documento aparte, o puede ser incluido dentro del Plan de Pruebas según su extensión. Aquí pueden definirse los tipos de pruebas a realizar (funcionales, de desempeño, de interfaces, no funcionales, etc.), requerimientos especiales de las pruebas, configuraciones a probar, subconjuntos de datos a considerar, nivel de pruebas de regresión, entre otros aspectos.

La estrategia de pruebas incluirá:

* **Tipos de Pruebas**: Pruebas funcionales, pruebas de rendimiento y pruebas de interfaz.
* **Configuraciones a Probar**: Pruebas en diferentes navegadores y dispositivos móviles.
* **Subconjuntos de Datos**: Utilización de datos de prueba representativos para simular escenarios de usuario real.
* **Nivel de Pruebas de Regresión**: Se llevará a cabo un conjunto de pruebas de regresión para cada nueva funcionalidad implementada, asegurando que no se introduzcan errores en las áreas existentes.

**Criterios de Aceptación o Rechazo**

**Criterios de Aceptación o Rechazo**

Son los criterios que serán considerados para dar por completado el Plan de Pruebas de Software, por ejemplo: Completar 100% de pruebas unitarias, cierto porcentaje de casos exitosos, cobertura de todos los componentes y líneas de código, porcentaje de defectos corregidos, entre otros.

**Criterios de Aceptación**:  
El plan de pruebas será considerado exitoso si:

* Se ha ejecutado el 100% de los casos de prueba planificados.
* Al menos el 95% de los defectos críticos han sido corregidos.

**Criterios de Rechazo**:  
El plan de pruebas se rechazará si:

* El porcentaje de éxito en las pruebas funcionales es inferior al 80%.
* Queda un número significativo de defectos críticos sin resolver.

**Criterios de Suspensión**

Establece claramente bajo qué condiciones se detienen un conjunto de casos de pruebas, por ejemplo en caso de existir defectos que impidan la ejecución de más casos de pruebas, cierto porcentaje de casos fallidos, o cualquier otro que se especifique.

Las pruebas se suspenderán si:

* Se encuentran defectos críticos que impidan continuar con las pruebas.
* Más del 30% de los casos de prueba fallan en una ejecución específica.

**Criterios de Reanudación**

Luego de haber suspendido las pruebas, aquí se establece bajo qué criterios se reanudarán.

Las pruebas se reanudarán cuando:

* Los defectos críticos hayan sido corregidos.
* El equipo de desarrollo de por confirmado la estabilidad del sistema.

**Entregables**

Establece que se entregará como parte de la ejecución del plan, por ejemplo: Documento de Plan de Pruebas, Casos de Pruebas, Especificación de Diseño de Casos, Logs de errores, Reportes de incidencias, evidencias de pruebas, reportes emitidos por herramientas de pruebas y cualquier otro que se establezca.

**Plan de Pruebas**: Documento que detalla la estrategia, objetivos, recursos, cronograma y tipos de pruebas que se realizarán para el software.

**Casos de Prueba**: Conjunto de pruebas a realizar para validar las funcionalidades del sistema, incluyendo los inputs, outputs esperados, y criterios de aceptación.

**Resultados de las Pruebas**: Registro formal de los resultados obtenidos tras la ejecución de los casos de prueba, indicando si las funcionalidades fueron validadas correctamente o fallaron.

**Recursos**

**Requerimientos de Entornos – Hardware**

Lista de los requerimientos de equipos, hardware y red necesarios para completar las actividades del Plan de Pruebas de Software. Incluye Servidores de Aplicación, Bases de Datos, Equipos de PC que necesitan los Testers, Conectividad a la red (incluyendo accesos), entre otros.

Para el desarrollo y ejecución del plan de pruebas, se utilizarán los siguientes recursos de hardware:

* **Servidores de Aplicación**: Un servidor local alojará la aplicación web y la base de datos para simular el entorno real.
* **Equipos PC**: Cada tester utilizará su propio equipo para acceder y probar el sistema. No se requieren PC de alto rendimiento, pero sí con conexión estable a la red local.
* **Conectividad a la red**: Los testers deben tener acceso a la red local donde está ubicado el servidor, con permisos de acceso a las bases de datos y al sistema de prueba.

**Requerimientos de Entornos – Software**

Lista de los requerimientos de software necesarios para completar las actividades de prueba, puede incluir accesos a Sistemas (en entorno de pruebas) y Bases de Datos, así como instalación de software en los Computadores asignados a los Testers.

Los siguientes elementos de software serán necesarios para realizar las pruebas:

* **Visual Studio Code**: Entorno de desarrollo para realizar modificaciones y analizar el código.
* **Python y Django**: Framework para ejecutar la aplicación web y verificar que todas las funcionalidades estén correctamente implementadas.
* **Acceso a Bases de Datos(Mysql)**: Se requerirá acceso a la base de datos local configurada para el entorno de pruebas.

**Herramientas de Pruebas Requeridas**

Especifica las herramientas de software, metodologías o técnicas especiales empleadas en las pruebas, por ejemplo Herramientas de Automatización de Pruebas, Software de Gestión de Pruebas, entre otros.

Se utilizarán las siguientes herramientas y metodologías para ejecutar las pruebas:

* **Metodología SCRUM**: Para la organización del equipo de pruebas, con roles asignados, sprints, backlog y reuniones diarias.
* **Herramientas de Pruebas Manuales**: Dado el enfoque en un proyecto universitario, no se usará automatización compleja, pero se realizarán pruebas manuales exhaustivas en los casos de prueba definidos.

**Personal**

Lista del personal necesario para completar las actividades de pruebas, especificando sus roles, por ejemplo: Un (1) Líder de Pruebas, Cinco (5) Analista de Pruebas (Testers), Dos (2) especialistas en automatización de pruebas, entre otros.

El equipo de pruebas está compuesto por:

* **Líder de Pruebas**: Felipe Urbina, responsable de la coordinación de las pruebas y la evaluación de resultados.
* **Analista de Pruebas**: Sebastián Contreras, encargado de ejecutar los casos de prueba manualmente y documentar los resultados.
* **Especialista en Automatización de Pruebas**: Sebastián Durán, se encargará de evaluar si es necesario automatizar ciertas pruebas críticas.

**Entrenamiento**

Necesidades de entrenamiento en el Sistema o Aplicación, así como en las herramientas de prueba a utilizar.

Dado que es un proyecto universitario, el entrenamiento en el uso del sistema y las herramientas de pruebas será básico:

* **Herramientas Gratuitas**: Se utilizarán herramientas como Visual Studio Code, Python, y otros software libre, que no requieren formación avanzada.
* **Capacitación en la Aplicación Web**: Los testers recibirán una breve capacitación para familiarizarse con la funcionalidad de la plataforma web de creación y solicitud de trabajos.

**Planificación y Organización**

**Procedimientos para las Pruebas**

Especifica los procedimientos o metodología de pruebas a emplear durante la ejecución del plan de pruebas de software.

En el procedimiento para las pruebas se considerarán las siguientes opciones:

**Sprints**: Las pruebas se llevarán a cabo dentro de cada ciclo de desarrollo (sprint), validando las nuevas funcionalidades y ajustando cualquier error detectado de forma inmediata.

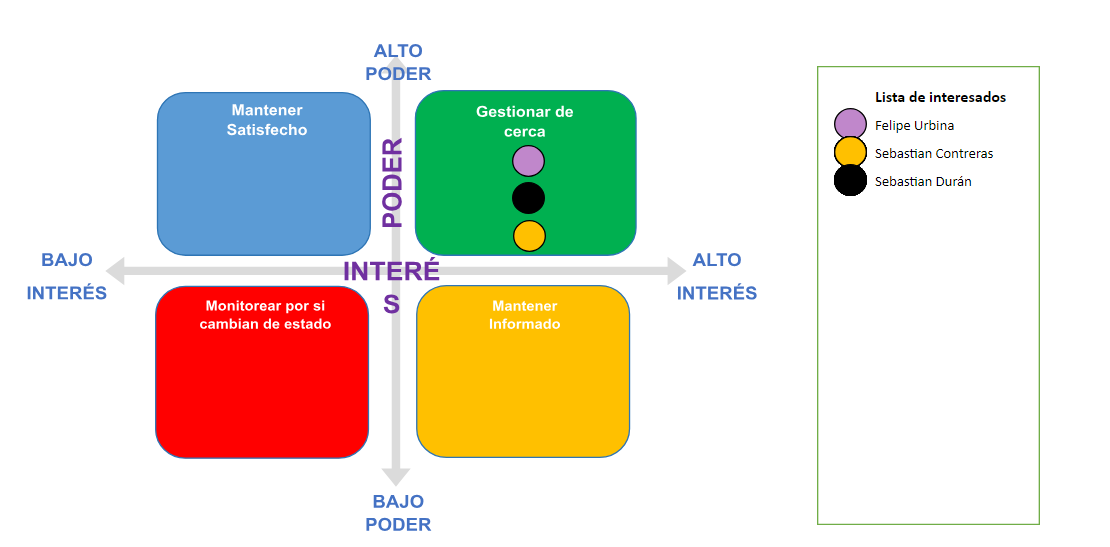
**Reuniones Diarias (Daily Scrum)**: Se revisa diariamente el progreso de las pruebas, identificando bloqueos y próximos pasos.

**Revisión y Retrospectiva de Sprint**: Al final de cada sprint, se evalúan los resultados de las pruebas y se discuten mejoras en el proceso de prueba para futuros sprints.

**Pruebas Funcionales y de Regresión**: Se probarán las funcionalidades nuevas y se verificarán las existentes para garantizar que no haya regresiones.

**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.

Teniendo en cuenta el cronograma de actividades del proyecto, lo ideal para poder organizar el plan de pruebas sería el siguiente:

| Actividad de Prueba | Fecha de Inicio | Fecha de Fin | Descripción |
| --- | --- | --- | --- |
| Diseño de Casos de Prueba | 05-10-2024 | 08-10-2024 | Se diseñarán los casos de prueba para validar las funcionalidades desarrolladas. |
| Revisión y Validación de Casos de Prueba | 09-10-2024 | 10-10-2024 | El equipo validará los casos de prueba antes de la ejecución formal. |
| Ejecución de Pruebas Funcionales (Web) | 16-10-2024 | 23-10-2024 | Se probarán las funcionalidades de la plataforma web, verificando que todas las funcionalidades cumplan con los requisitos. |
| Pruebas de Integración (Web) | 24-10-2024 | 26-10-2024 | Se integrarán las distintas partes del sistema para verificar su correcto funcionamiento. |
| Pruebas Unitarias y de Integración | 12-11-2024 | 13-11-2024 | Validación de módulos individuales y la integración con otros componentes del sistema. |
| Prueba de Usuario y Ajuste | 14-11-2024 | 15-11-2024 | Se realizarán pruebas con usuarios finales y ajustes basados en sus comentarios. |
| Aplicación de Pruebas Finales | 18-11-2024 | 18-11-2024 | Ejecución de pruebas finales, previo a la fase de cierre del proyecto. |
| Corrección de Errores Detectados | 19-11-2024 | 20-11-2024 | Se corregirán los errores identificados durante las pruebas finales. |