**Plan de Pruebas de Software**

***ConstructiveMaster***

***Fecha: 8/08/2024***

**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

Premisas 14

Dependencias y Riesgos 14

Referencias 15

Glosario 15

**Historial de Versiones**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| 8/08/2024 | 1.0 | Deiby Gerleyn Taborda Giraldo |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Información del Proyecto**

|  |  |
| --- | --- |
| Empresa / Organización |  |
| Proyecto | ConstructiveMaster |
| 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**

Este plan maestro de pruebas tiene como propósito asegurar que el sitio web en desarrollo cumpla con los estándares de calidad requeridos antes de su lanzamiento al mercado. Se realizarán pruebas exhaustivas en las funcionalidades críticas del sitio, incluyendo sistemas de autenticación, formularios de categorías y subcategorías, formulario de incorporación de los profesionales, y procesos de pago. El plan cubre las pruebas en los navegadores más populares y en dispositivos móviles y de escritorio, garantizando una experiencia de usuario consistente y segura.

Debido a restricciones de presupuesto y tiempo, se priorizarán las pruebas de las funcionalidades más críticas y los dispositivos más utilizados. El equipo de pruebas contará con un testers y un entorno de pruebas limitado, lo que requerirá una planificación cuidadosa para maximizar la eficiencia del proceso de pruebas.

El esfuerzo de pruebas estará alineado con los objetivos generales del proyecto, enfocándose en la estabilidad, seguridad y usabilidad del sitio. Las pruebas de seguridad recibirán atención prioritaria debido a la naturaleza sensible de los datos manejados. Este plan proporciona una visión general de los tipos de pruebas, criterios de aceptación, y recursos necesarios para llevar a cabo un proceso de pruebas efectivo.

**Alcance de las Pruebas**

**Elementos de Pruebas**

* Formulario de registro de usuario.
* Formulario de Inicio de sesión.
* Formulario que permite enviar una solicitud de incorporación.
* Formulario que permite añadir un administrador.
* Formulario de envió de solicitudes de trabajo.
* Asignación automática de profesional.
* Formulario que permite añadir categorías.
* Formulario que permite añadir subcategorías.
* Componente TablaAdmin y sus funcionalidades.
* Formularios que editan cada registro permitido de la base de datos.
* Componente BotonDelete.
* Componente BotonEditar.
* Componente ConfirmDelete.
* Componente ConfirmEdit.
* Formulario de administrador para ingresar un nuevo cliente.
* Formulario de administrador para editar los datos de los clientes.
* Componente de confirmación para eliminar un registro.
* Componente de confirmación para aceptar una solicitud de un profesional y trabajos.
* Formulario de administrador para ingresar un nuevo profesional.
* Formulario de administrador para editar los datos de un profesional.
* Formulario de administrador para ingresar una nueva categoría.
* Formulario de administrador para editar las propiedades de una categoría.
* Barra de búsqueda de profesionales.

**Nuevas Funcionalidades a Probar**

* **Pasarela de pagos:** permitir el pago de los porcentajes de cada trabajo por parte de los profesionales de manera online.
* **Algoritmo de asignación automática de un profesional:** Permitir que un algoritmo se encargue de la asignación automática de un profesional a una solicitud de trabajo.

**Pruebas de Regresión**

* Verificar que después de agregar una nueva opción en el menú, las otras opciones siguen funcionando sin problemas.
* Asegurarse de que, al mejorar la página de perfil de los profesionales, la página de búsqueda aún funciona correctamente.
* Asegurarse de que cada que se le añada un nuevo campo al componente de TablaAdmin no se generen errores.
* Asegurarse de que, el formulario reutilizable cada que se le añada un campo no genere ningún error.

**Funcionalidades a No Probar**

Todas las funcionalidades, sin excepción alguna se probarán.

**Enfoque de Pruebas (Estrategia)**

**. Tipos de Pruebas a Realizar:**

* **Pruebas Funcionales**:
  + Verificar que el formulario de registro para usuarios y profesionales funcione correctamente.
  + Comprobar que los usuarios puedan buscar y filtrar profesionales por especialidad, ubicación, etc.
  + Validar que los profesionales puedan actualizar su perfil con información y archivos (como currículums).
* **Pruebas de Desempeño**:
  + Evaluar la velocidad de carga del sitio cuando 100 usuarios intentan acceder al mismo tiempo.
  + Medir el tiempo de respuesta del sitio al realizar una búsqueda con múltiples filtros.
* **Pruebas de Interfaces**:
  + Comprobar que la página de inicio se vea bien en diferentes dispositivos (móviles, tablets, PC).
  + Revisar que los botones y enlaces sean fácilmente visibles y funcionales en todas las resoluciones.

**2. Requerimientos Especiales de las Pruebas:**

* Las pruebas deben incluir dispositivos con diferentes versiones de Android e iOS para garantizar compatibilidad.

**3. Configuraciones a Probar:**

* **Navegadores**: Google Chrome, Firefox, Safari.
* **Sistemas Operativos**: Windows 10, macOS.
* **Dispositivos**: Móviles (Samsung Galaxy S21, Xiaomi Redmi 11), Tablets (iPad Pro).

**4. Datos a Considerar:**

* Crear usuarios ficticios con diferentes perfiles: un cliente, un profesional en arquitectura, un electricista, etc.
* Generar escenarios donde el profesional tenga evaluaciones positivas y negativas para verificar cómo se muestran.

**5. Nivel de Pruebas de Regresión:**

* Cada vez que se agregue o modifique una función (como un nuevo filtro en la búsqueda), se deben revisar las funciones relacionadas:
  + Que las búsquedas sin filtros sigan funcionando bien.
  + Que los perfiles de los profesionales se muestren correctamente después de aplicar filtros.
  + Que el proceso de contratación siga sin errores después de modificar alguna parte de la interfaz.

**Criterios de Aceptación o Rechazo**

**Criterios de Aceptación o Rechazo**

1. **Completar el 100% de las pruebas unitarias**: Asegurarse de que se hayan revisado y aprobado todos los pequeños componentes del sistema de manera individual.
2. **Cierto porcentaje de casos exitosos**: Definir que, por ejemplo, el 95% de todas las pruebas realizadas deben pasar sin errores.
3. **Cobertura de todos los componentes y líneas de código**: Confirmar que todas las partes del software han sido probadas y que no se dejó ninguna sin revisar.
4. **Porcentaje de defectos corregidos**: Establecer que, antes de dar por terminado el proceso, se debe haber corregido al menos el 90% de los errores encontrados durante las pruebas.

**Criterios de Suspensión**

* **Defectos críticos**: Si encuentras un error tan grande que impide continuar con las pruebas, como que el sistema se caiga o no permita avanzar de una pantalla a otra.
* **Alto porcentaje de fallos**: Si un gran número de pruebas están fallando, como más del 50%, puede ser mejor detener todo y revisar qué está mal antes de seguir probando.
* **Problemas técnicos**: Si el entorno de pruebas (servidores, base de datos, herramientas) no funciona correctamente y no permite ejecutar las pruebas de manera efectiva.

**Criterios de Reanudación**

* **Corrección de defectos críticos**: Si el problema que causó la suspensión ha sido solucionado y el sistema ahora puede ser probado sin interrupciones.
* **Revisar y validar las correcciones**: Verificar que las soluciones aplicadas no generaron nuevos errores y que el problema original ya no existe.
* **Estabilidad del entorno de pruebas**: Asegurarte de que los servidores, herramientas y recursos necesarios para realizar las pruebas estén funcionando correctamente.
* **Reevaluación de los casos de prueba**: Determinar si es necesario ajustar o rehacer algunas pruebas debido a los cambios realizados.

**Entregables**

* **Documento de plan de pruebas:** Un documento que explica cómo se realizaron las pruebas, qué se probó y qué herramientas se usaron.
* **Logs de errores**: Un registro de todos los errores que se encontraron durante las pruebas.
* **Evidencias de pruebas**: Capturas de pantalla, videos o registros que demuestran que las pruebas se realizaron y cuáles fueron los resultados.

**Recursos**

**Requerimientos de Entornos – Hardware**

RAM 4GB

Procesador AMD Ryzen 3 2.60GHz, Radeon Graphics.

Sistema operativo 64 bits

Red wifi

Conexión a internet por cable

**Requerimientos de Entornos – Software**

XAMPP 8.2.12

Node js

Editor de código (VS code, Sublime text 3)

**Herramientas de Pruebas Requeridas**

Cypress

Selenium

**Personal**

Un desarrollador de software que cumple la función de testers.

**Entrenamiento**

* 1. **Entrenamiento del sitio web:**
* Cómo enviar una solicitud para incorporarse en el equipo de ConstructiveMaster.
* Cómo enviar una solicitud de trabajo.
* Cómo gestionar el panel de administrador de manera general y detallada.
* Cómo se presentarán los profesionales en las interfaces de los clientes.
* Cómo se gestiona las cuentas de los profesionales y los clientes.
* Cómo se mostrarán las categorías y subcategorías en las interfaces.
* Cómo se manejará el manejo de imágenes de las categorías y el manejo de hojas de vida.
  1. **Entrenamiento en las herramientas de prueba:**
* Cómo escribir y ejecutar scripts de prueba.
* Cómo configurar la herramienta para que interactúe con el sistema.
* Cómo analizar los resultados y generar reportes.

**Planificación y Organización**

**Procedimientos para las Pruebas**

**1. Preparación de Entorno de Pruebas**

* **Configuración del Entorno**: Configurar los servidores y equipos necesarios de acuerdo con los requerimientos de hardware y software especificados. Asegurarse de que las herramientas de pruebas estén correctamente instaladas y configuradas.
* **Datos de Prueba**: Preparar y cargar datos de prueba representativos en el entorno para simular escenarios reales de uso.

**2. Diseño de Casos de Prueba**

* **Definición de Casos de Prueba**: Crear casos de prueba detallados basados en los requisitos funcionales y no funcionales del software. Cada caso de prueba debe incluir condiciones iniciales, pasos a seguir, resultados esperados y criterios de aceptación.
* **Revisión de Casos de Prueba**: Revisar los casos de prueba con el equipo de desarrollo y otras partes interesadas para asegurar que cubren todos los requisitos y escenarios necesarios.

**3. Ejecución de Pruebas**

* **Ejecución de Pruebas Manuales**: Llevar a cabo las pruebas manuales según los casos de prueba diseñados, documentando los resultados y cualquier defecto encontrado.
* **Ejecución de Pruebas Automatizadas**: Ejecutar pruebas automatizadas utilizando herramientas como Cypress, Selenium, y otros frameworks de automatización. Asegurarse de que las pruebas automatizadas se integren en el proceso de integración continua.

**4. Registro y Gestión de Defectos**

* **Documentación de Defectos**: Registrar todos los defectos encontrados durante las pruebas en el sistema de gestión de incidencias (por ejemplo, Trello o Notion). Incluir detalles como la severidad, los pasos para reproducir el defecto, y cualquier evidencia relevante.
* **Gestión de Defectos**: Priorizar y asignar defectos para su corrección. Realizar un seguimiento del estado de los defectos y asegurar que se resuelvan antes de la liberación del software.

**5. Revisión y Evaluación**

* **Revisión de Resultados de Pruebas**: Revisar los resultados de las pruebas para asegurar que todos los casos de prueba han sido ejecutados y que los criterios de aceptación han sido cumplidos.
* **Evaluación de Cobertura de Pruebas**: Evaluar la cobertura de pruebas para asegurar que todos los módulos y funcionalidades críticas han sido probados.

**6. Reporte de Resultados**

* **Generación de Informes**: Crear informes detallados sobre los resultados de las pruebas, incluyendo casos de prueba ejecutados, defectos encontrados, y métricas de rendimiento. Utilizar herramientas de gestión como Notion para centralizar la documentación.
* **Revisión de Informes**: Revisar y discutir los informes con el equipo de desarrollo y otras partes interesadas para asegurar que se entienden los resultados y se planifican las acciones correctivas necesarias.

**7. Cierre de Pruebas**

* **Verificación de Correcciones**: Asegurarse de que todos los defectos críticos y de alta severidad han sido corregidos y que las correcciones han sido validadas.

**Revisión Final**: Realizar una revisión final del proceso de pruebas para identificar lecciones aprendidas y oportunidades de mejora para futuros ciclos de pruebas.

**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).

Todas las responsabilidades las llevará a cabo **Deiby Gerleyn Taborda** como único desarrollador del software y que por ende es la persona que conoce de inicio a fin cómo funciona el sitio web.

**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.

El cronograma de pruebas se desarrollará basado en estimaciones de tiempo de cada actividad de prueba, asegurando la cobertura completa y permitiendo la entrega a tiempo. Se identificarán hitos críticos, como el inicio y finalización de pruebas unitarias, funcionales, de regresión, y entrega de informes finales.

**Premisas**

Las premisas relacionadas con las tareas de pruebas de software, incluyendo limitaciones de tiempo, disponibilidad de recursos que se asumen, uso de una metodología de pruebas, uso de una herramienta, entre otros.

**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.

**Referencias**

Lista de todos los documentos que pueden citarse como apoyo o para ampliar el contenido del plan de pruebas. Algunos ejemplos de lo que se puede hacer referencia aquí son:

* Plan de Proyecto.
* Especificaciones de Requerimientos.
* Diseño General.
* Diseño Detallado.
* Procedimientos y estándares de Desarrollo.
* Procedimientos y estándares de Pruebas.
* Metodologías, Procedimientos y estándares corporativos.

**Glosario**

Definiciones de términos usados en la documentación, y general sobre el área de pruebas.