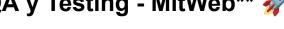
Proyecto de QA y Testing - MitWeb 🚀



Planificación

@### **Definición de Objetivos del Proyecto de Testing - MitWeb**

@#### Objetivo Principal:

Garantizar la robustez y confiabilidad del sistema de gestión de ventas "MitWeb" mediante un proceso de testing exhaustivo, contribuyendo así a una experiencia de usuario sin problemas y a la mejora continua de la calidad del software.

@#### Objetivos Específicos:

1. **Asegurar la Funcionalidad Completa:**

- Verificar que todas las funciones clave de "MitWeb" operen según lo especificado en los requisitos funcionales.

2. **Garantizar la Seguridad del Sistema:**

- Identificar y corregir posibles vulnerabilidades de seguridad.
- Validar el cumplimiento de las mejores prácticas de seguridad establecidas.

3. **Optimizar el Rendimiento:**

- Evaluar el rendimiento del sistema bajo diferentes cargas de trabajo.
- Identificar y corregir posibles cuellos de botella para mejorar la eficiencia.

4. **Compatibilidad Multiplataforma:**

- Verificar la compatibilidad del software en diferentes sistemas operativos y navegadores.
 - Asegurar una experiencia consistente para todos los usuarios.

5. **Validación de Datos:**

- Verificar la integridad y validez de los datos procesados por "MitWeb".
- Identificar y corregir cualquier error en la manipulación de datos.

Criterios de Éxito:

- **Cobertura de Pruebas:**

- Lograr una cobertura del 95% en casos de prueba diseñados.
- Incluir pruebas unitarias, de integración, de sistema y de aceptación del usuario.

- **Tasa de Defectos:**

- Mantener la tasa de defectos por debajo del 2% en relación con el número total de casos de prueba ejecutados.

- **Eficiencia del Proceso:**

- Completar el ciclo de vida del testing dentro del plazo establecido.
- Minimizar el tiempo de inactividad durante las actualizaciones y correcciones.

- **Retroalimentación Positiva del Usuario: **

- Recopilar comentarios positivos de los usuarios en relación con la estabilidad y usabilidad de "MitWeb".

- **Cumplimiento de Normativas:**

- Asegurar el cumplimiento de las normativas y estándares de la industria, especialmente aquellos relacionados con la protección de datos y la privacidad del usuario.

Recolección de Requisitos:

Análisis de Requisitos:

1. **Examinar detenidamente la documentación de requisitos para entender la funcionalidad del software.**

2. **Reuniones con Stakeholders:**

- Organizar reuniones para clarificar requisitos con el equipo de desarrollo e interesados.

3. **Creación de Matriz de Rastreo:**

- Desarrollar una matriz que relacione requisitos con casos de prueba para facilitar el seguimiento.

4. **Identificación de Escenarios de Prueba:**

- Identificar y documentar escenarios de prueba basados en requisitos, abordando casos positivos y negativos.

5. **Validación con Stakeholders:**

- Presentar escenarios a los interesados para validar su alineación con expectativas y ajustar según la retroalimentación.

6. **Priorización de Escenarios:**

- Priorizar escenarios según importancia y relevancia para enfocar esfuerzos de prueba.

7. **Documentación Detallada:**

- Documentar requisitos y escenarios detalladamente, incluyendo datos de entrada y criterios de aceptación.

8. **Revisión con el Equipo de Desarrollo:**

- Revisar escenarios con el equipo de desarrollo para alinearlos con la implementación y ajustar según sugerencias.

Casos de Prueba para el Registro de Usuarios:

√ #### Caso de Prueba para la Validación del Formulario:

- **Objetivo:**

- Verificar que el formulario de registro acepta datos válidos y muestra mensajes de error para entradas inválidas.

- **Pasos:**

- 1. Llenar el formulario con datos válidos.
- 2. Verificar que el sistema acepte la entrada.
- 3. Verificar que el usuario sea redirigido correctamente después de un registro exitoso.
- 4. Repetir el proceso, pero esta vez introducir datos inválidos en cada campo.
- 5. Verificar que se muestren mensajes de error adecuados.

📏 #### Caso de Prueba para la Partición de Equivalencia de Edades:

- **Objetivo:**

- Validar la funcionalidad de registro para diferentes rangos de edades.

- **Pasos:**

- 1. Registrar un usuario con la edad mínima permitida.
- 2. Verificar que el sistema acepte la entrada.
- 3. Registrar un usuario con una edad dentro del rango normal.
- Verificar que el sistema acepte la entrada.
- 5. Registrar un usuario con la edad máxima permitida.
- 6. Verificar que el sistema acepte la entrada.

🔐 #### Caso de Prueba para el Análisis de Valores Límite de Contraseñas:

- **Objetivo:**

- Validar la capacidad del sistema para manejar contraseñas de longitud mínima y máxima.

- **Pasos:**

- 1. Intentar registrar un usuario con una contraseña de longitud mínima.
- 2. Verificar que el sistema acepte la entrada.

- 3. Intentar registrar un usuario con una contraseña de longitud máxima.
- 4. Verificar que el sistema acepte la entrada.
- 5. Intentar registrar un usuario con una contraseña justo por debajo del límite mínimo.
- 6. Verificar que se muestre un mensaje de error.
- 7. Intentar registrar un usuario con una contraseña justo por encima del límite máximo.
- 8. Verificar que se muestre un mensaje de error.

📝 #### Caso de Prueba para la Cobertura de Campos Obligatorios:

- **Objetivo:**

- Asegurarse de que todos los campos obligatorios se validen correctamente.

- **Pasos:**

- 1. Intentar registrar un usuario sin completar ningún campo obligatorio.
- 2. Verificar que se muestren mensajes de error para cada campo obligatorio.
- 3. Completar solo los campos obligatorios y verificar que el sistema acepte la entrada.

🔐 #### Caso de Prueba para la Seguridad:

- **Objetivo:**

- Validar la seguridad del sistema contra posibles ataques.

- **Pasos:**

- 1. Intentar registrar un usuario con una contraseña débil.
- 2. Verificar que se muestre un mensaje de advertencia sobre la debilidad de la contraseña.
 - 3. Intentar inyectar código malicioso en los campos del formulario.
 - 4. Verificar que el sistema filtre y bloquee correctamente la entrada.

🗱 ### Configuración del Entorno de Pruebas:

Objetivo:

Configurar un entorno de pruebas que refleje fielmente el entorno de producción, garantizando que todas las herramientas necesarias estén instaladas y configuradas correctamente.

Pasos:

1. **Configuración del Sistema Operativo: **

- Asegurar que el sistema operativo en el entorno de pruebas sea consistente con el entorno de producción.
 - Instalar la misma versión del sistema operativo que se utiliza en producción.
 - Configurar configuraciones de red para simular condiciones de producción.

2. **Instalación del Software Base:**

- Instalar componentes de software esenciales para la aplicación.
- Instalar servidores web, bases de datos y otros componentes necesarios.
- Configurar los servicios para que reflejen la configuración de producción.

3. **Despliegue de la Aplicación:**

- Desplegar la aplicación en el entorno de pruebas.
- Utilizar los mismos scripts y procesos de implementación que en producción.
- Configurar la aplicación para interactuar con servicios simulados si es necesario.

4. **Configuración de Herramientas de Pruebas:**

- Asegurarse de que todas las herramientas de prueba estén listas y configuradas.
- Instalar y configurar herramientas de prueba, como Cypress/Selenium para pruebas automatizadas y JIRA/Trello para seguimiento de defectos.
 - Configurar entornos virtuales para pruebas manuales y automatizadas.

5. **Configuración de Datos de Pruebas:**

- Preparar datos de prueba que reflejan escenarios del mundo real.
- Crear scripts para cargar datos de prueba en la base de datos.
- Garantizar que los datos sean anónimos y cumplan con las regulaciones de privacidad.

6. **Seguridad del Entorno:**

- Configurar permisos y accesos de usuario según los roles definidos.
- Implementar certificados SSL si la aplicación lo requiere.

√ ### Validación:

Pruebas de Conectividad:

- **Objetivo:**

- Verificar la conectividad entre los diferentes componentes del entorno de pruebas.

- **Acciones:**

- Ejecutar pruebas de ping y tráfico de red simulado.
- Verificar la comunicación entre la aplicación y la base de datos.

********** Pruebas de Configuración:

- **Objetivo:**

- Asegurarse de que todas las configuraciones estén correctas.

- **Acciones:**

- Revisar registros de configuración para verificar la consistencia.
- Ejecutar scripts de configuración y verificar los resultados.

m=#### Pruebas de Herramientas:

- **Objetivo:**

- Validar que las herramientas de prueba estén operativas.

- **Acciones:**

- Ejecutar pruebas de prueba rápida en las herramientas de prueba.
- Verificar la integración entre herramientas, por ejemplo, entre JIRA y el sistema de automatización.

Pruebas de Datos de Prueba:

- **Objetivo:**

- Confirmar que los datos de prueba sean adecuados.

- **Acciones:**

- Ejecutar casos de prueba utilizando datos de prueba y verificar la coherencia de los resultados.

Resultados y Documentación:

📻 #### Registro de Configuración:

- **Objetivo:**

- Documentar la configuración del entorno de pruebas.

- **Acciones:**

- Crear un documento detallado que describa todas las configuraciones y ajustes realizados.

- **Objetivo:**

- Preparar un informe indicando la exitosa configuración del entorno de pruebas.

- **Acciones:**

- Presentar el informe al equipo de desarrollo y de prueba.
- Obtener la aprobación para proceder con las pruebas.

m### Desarrollo de Scripts de Prueba:

- **Objetivo:**

- Escribir scripts de prueba automatizados para casos de prueba seleccionados.

- Utilizar un marco de prueba adecuado y seguir las mejores prácticas de programación.

Ejecución de Pruebas:

- **Objetivo:**

- Ejecutar pruebas manuales y automatizadas.
- Registrar resultados y documentar cualquier defecto encontrado.

Evaluación y Corrección:

🐛 #### Análisis de Defectos: 🐛

- **Objetivo:**

- Clasificar y priorizar los defectos encontrados.
- Documentar información detallada sobre cada defecto.

Retesting:

- **Objetivo:**

- Realizar pruebas para verificar que las correcciones aplicadas a áreas específicas del código han solucionado adecuadamente los problemas identificados inicialmente.

🔄 #### Pruebas de Regresión:

- **Objetivo:**

- Realizar pruebas para asegurarse de que las correcciones no hayan introducido nuevos problemas en áreas no afectadas por los cambios.

₩### Documentación y Entrega:

Documentación:

- **Objetivo:**

- Documentar todos los aspectos del proceso de prueba.
- Preparar informes de prueba que destaquen la cobertura y los resultados.

✓ #### Revisión y Aprobación:

- **Obietivo:**

- Revisar los resultados de las pruebas con el equipo de desarrollo y los interesados.
- Obtener la aprobación para proceder con la entrega.

Entrega:

- **Objetivo:**

- Proporcionar el informe final de pruebas y documentación relevante.
- Asegurarse de que todos los criterios de aceptación se cumplan.

🔄### Seguimiento y Mejora Continua: 🚀

- **Objetivo:**

- Realizar un seguimiento continuo y mejorar el proceso de pruebas.

- **Acciones:**

- Analizar resultados periódicamente.
- Identificar oportunidades de mejora y ajustar el proceso según sea necesario.

Revisión Post-implementación:

- **Objetivo:**

- Evaluar el rendimiento del software en producción.
- Recopilar retroalimentación del usuario y evaluar cualquier problema no detectado durante las pruebas.

- **Enfoque:**

- Análisis de resultados post-implementación.
- Recopilación activa de comentarios de usuarios para evaluar la experiencia.

Retrospectiva del Proceso de Pruebas:

- **Objetivo:**

- Evaluar el rendimiento del software en producción y mejorar el proceso de pruebas.

- **Pasos:**

- Análisis detallado de los resultados.
- Recopilación de retroalimentación del usuario y evaluación de problemas no detectados durante las pruebas.

٠.,