**Plan de Pruebas**

**1. ¿Cómo escribir un caso de prueba?**

Un caso de prueba debe ser claro, estructurado y fácil de ejecutar. A continuación, se presentan los elementos clave que debe contener:

**Estructura de un Caso de Prueba**

* **ID del Caso de Prueba:** Identificador único (Ejemplo: TC-001).
* **Título:** Breve descripción del objetivo del caso (Ejemplo: "Validar registro de usuario").
* **Descripción:** Explicación detallada de lo que se va a probar.
* **Precondiciones:** Condiciones necesarias antes de ejecutar la prueba (Ejemplo: "El usuario no debe estar registrado").
* **Pasos de Ejecución:** Lista numerada de los pasos para ejecutar la prueba.
* **Datos de Prueba:** Valores de entrada requeridos para ejecutar el caso de prueba.
* **Resultado Esperado:** Lo que se espera que ocurra si la funcionalidad funciona correctamente.
* **Resultado Obtenido:** Resultado real después de la ejecución de la prueba.
* **Estado:** Puede ser "Aprobado" o "Fallido", dependiendo del resultado.
* **Notas/Comentarios:** Observaciones adicionales, como logs o capturas de pantalla.

**Ejemplo de Caso de Prueba:**

|  |  |
| --- | --- |
| ID | TC-001 |
| Título | Validar registro de usuario |
| Descripción | Verificar que un usuario pueda registrarse con datos válidos. |
| Precondiciones | Usuario no registrado. |
| Pasos de Ejecución | 1. Ir a la página de registro. 2. Ingresar datos válidos. 3. Hacer clic en "Registrarse". |
| Datos de Prueba | Nombre: Juan Pérez, Email: [juan@example.com](mailto:juan@example.com), Contraseña: Password123 |
| Resultado Esperado | El usuario es registrado con éxito y recibe un mensaje de confirmación. |
| Resultado Obtenido | - |
| Estado | - |
| Notas | - |

**2. ¿Cómo reportar un defecto?**

Cuando un caso de prueba falla, se debe documentar el defecto correctamente en un sistema de gestión de incidencias como Jira, Trello o TestRail.

**Estructura de un Reporte de Defecto**

* **ID del Defecto:** Identificador único (Ejemplo: BUG-001).
* **Título:** Breve descripción del problema (Ejemplo: "Error 500 en el registro de usuario").
* **Descripción:** Explicación detallada del defecto.
* **Severidad:** Puede ser "Crítica", "Alta", "Media" o "Baja".
* **Prioridad:** Puede ser "Alta", "Media" o "Baja", según la urgencia de su corrección.
* **Pasos para Reproducir:** Lista detallada de pasos para replicar el defecto.
* **Resultado Esperado:** Lo que debería ocurrir si el sistema funcionara correctamente.
* **Resultado Obtenido:** Lo que realmente ocurre.
* **Evidencia:** Capturas de pantalla, logs o grabaciones de la falla.
* **Estado:** "Abierto", "En proceso", "Resuelto", "Cerrado".
* **Responsable:** Persona o equipo asignado para corregir el defecto.

**Ejemplo de Reporte de Defecto:**

|  |  |
| --- | --- |
| ID | BUG-001 |
| Título | Error 500 en el registro de usuario |
| Descripción | Al intentar registrar un usuario con un email duplicado, el sistema devuelve un error 500 en lugar de un mensaje de validación. |
| Severidad | Alta |
| Prioridad | Alta |
| Pasos para Reproducir | 1. Ir a la página de registro. 2. Ingresar un email ya registrado. 3. Hacer clic en "Registrarse". |
| Resultado Esperado | Mostrar un mensaje "El email ya está registrado" con código HTTP 400. |
| Resultado Obtenido | Se muestra un error 500. |
| Evidencia | Captura de pantalla adjunta. |
| Estado | Abierto |
| Responsable | Equipo de Backend |

**3. Criterios para Incluir un Caso de Prueba en una Suite de Humo**

La **suite de humo** incluye los casos de prueba más críticos para validar que la aplicación es estable tras una nueva implementación.

**Criterios para incluir un caso de prueba en la suite de humo:**

* Valida funcionalidades esenciales como autenticación, acceso y operaciones básicas.
* Debe ejecutarse rápidamente y no depender de datos complejos.
* Debe cubrir los flujos principales sin los cuales la aplicación no funciona.
* Debe ejecutarse antes de pruebas más exhaustivas (como regresión o exploratorias).

**Ejemplo de Casos de Prueba en la Suite de Humo:**

1. **Registro de usuario** (Si no funciona, ningún usuario podrá acceder al sistema).
2. **Inicio de sesión** (Validar que los usuarios pueden autenticarse).
3. **Carga de la página principal** (Garantizar que la aplicación responde correctamente).
4. **Acceso a la base de datos** (Validar conectividad con la BD sin errores).

**4. Criterios para Incluir un Caso de Prueba en una Suite de Regresión**

La **suite de regresión** valida que nuevas actualizaciones no introduzcan defectos en funcionalidades previamente desarrolladas.

**Criterios para incluir un caso de prueba en la suite de regresión:**

* Casos de prueba relacionados con funcionalidades críticas del sistema.
* Casos de prueba de defectos previamente corregidos para evitar regresiones.
* Pruebas que validan la integración entre diferentes módulos.
* Casos con alto impacto en la experiencia del usuario o en procesos clave.
* Casos que involucran cálculos, seguridad y permisos de usuario.

**Ejemplo de Casos de Prueba en la Suite de Regresión:**

1. **Registro de usuario con datos válidos e inválidos.**
2. **Inicio de sesión con credenciales correctas e incorrectas.**
3. **Flujo de compra con diferentes métodos de pago.**
4. **Gestión de permisos de usuario y roles.**
5. **Carga y actualización de datos en la base de datos.**
6. **Pruebas de integración entre microservicios.**

**5. Conclusión**

Este plan de pruebas proporciona una guía clara para estructurar los casos de prueba, reportar defectos y organizar las pruebas en suites de humo y regresión. Al seguir estos criterios, se garantiza una validación efectiva del sistema, asegurando su estabilidad y calidad antes de cada liberación.