

**Plan de Pruebas de Software para una la aplicación de Escritorio en Java
para la Gestión de un Consultorio Odontológico**

Maria Jose Velez Gutierrez

Yuliet Faizuli Pachon Caro

Jorge Miller Gutierrez Ospina

Analisis y desarrollo de software

Instructor

Ivan Leonardo Medina Gomez

SENA

Septiembre 2024

Introducción

El presente documento describe el “Plan de Pruebas de Software” que se aplicará a la aplicación de escritorio en Java para la gestión de un consultorio odontológico “Maria Jose Velez Gutierrez”. La finalidad de este plan es garantizar que el sistema cumple con los requisitos funcionales y no funcionales mediante un proceso de evaluación estructurado, que incluye la definición de los tipos de pruebas a realizar, las herramientas utilizadas y la gestión de riesgos. A través de este plan, se busca identificar posibles errores en el sistema y asegurar su calidad antes de la implementación en el entorno real.

1. Análisis de los Requerimientos de Software

Para realizar un plan de pruebas adecuado para la aplicación de escritorio que gestiona un consultorio odontológico, es fundamental identificar claramente los requerimientos tanto funcionales como no funcionales. A continuación, se describen los principales requerimientos del software, incluyendo las funcionalidades del sistema y los criterios que deben ser probados.

Funcionalidades Principales:

Login y Registro de Usuarios:

Descripción: Los usuarios deben poder registrarse e iniciar sesión en la aplicación. Los roles de usuario incluyen administrador, odontólogo y auxiliar.

Requerimientos:

Registro de nuevos usuarios con roles asignados.

Inicio de sesión seguro y basado en encriptación de contraseñas.

Soporte para recuperación de contraseñas.

Validaciones de usuario y verificación de credenciales.

Objetivo: Garantizar que solo usuarios autorizados accedan al sistema con sus roles específicos.

Encriptación de Contraseñas:

Descripción: Las contraseñas deben estar protegidas mediante algoritmos de encriptación seguros (por ejemplo, BCrypt o SHA-3).

Requerimientos:

Las contraseñas deben ser encriptadas antes de almacenarse en la base de datos.

Las contraseñas deben compararse de manera segura durante el inicio de sesión.

Verificar que las contraseñas no sean almacenadas en texto plano.

Objetivo: Proteger la privacidad y seguridad de los datos de los usuarios.

Gestión de Usuarios:

Descripción: El sistema debe permitir a los administradores gestionar (crear, editar, eliminar) usuarios del sistema.

Requerimientos:

Verificación de que los usuarios no dupliquen sus credenciales (unicidad de nombre de usuario).

Validaciones de datos durante el registro y actualización de usuarios.

Permitir la modificación de roles asignados sin comprometer la seguridad.

Objetivo: Asegurar una gestión eficiente de los usuarios con permisos y roles adecuados.

Gestión de Odontólogos:

Descripción: Los odontólogos deben poder registrarse y se debe gestionar su información, horarios y asignación de citas.

Requerimientos:

Registro de nuevos odontólogos.

Modificación y actualización de los datos personales y profesionales de los odontólogos.

Control de disponibilidad y horarios de trabajo.

Objetivo: Facilitar la administración de odontólogos y asegurar que las citas se gestionen correctamente según disponibilidad.

Gestión de Pacientes:

Descripción: Los pacientes deben poder registrarse y se debe gestionar su información personal y médica.

Requerimientos:

Registro, actualización y eliminación de datos de los pacientes.

Consulta de historiales médicos y registros de citas pasadas.

Protección de datos sensibles de los pacientes (cumplimiento con normativas como GDPR).

Objetivo: Mantener un registro preciso y seguro de los pacientes y su historial médico.

Gestión de Citas Odontológicas:

Descripción: El sistema debe permitir la programación, modificación y cancelación de citas entre los pacientes y los odontólogos.

Requerimientos:

Permitir a los usuarios agendar citas según la disponibilidad del odontólogo.

Capacidad para modificar o cancelar citas de manera fácil.

Notificaciones de recordatorio de citas para pacientes y odontólogos.

Verificación de conflictos de horario en las citas agendadas.

Objetivo: Facilitar la coordinación de citas entre pacientes y odontólogos de manera eficiente y organizada.

Facturación:

Descripción: El sistema debe gestionar la facturación de los tratamientos y servicios proporcionados.

Requerimientos:

Generación automática de facturas con detalles de los tratamientos, fechas y costos.

Registro y control de pagos realizados por los pacientes.

Capacidad de emitir facturas impresas.

Objetivo: Asegurar que todas las transacciones y pagos sean registrados y gestionados de manera precisa.

Gestión de Inventario:

Descripción: El sistema debe gestionar los suministros odontológicos, monitoreando el inventario y generando alertas de stock bajo.

Requerimientos:

Registro de entradas y salidas de productos odontológicos.

Control del stock y alerta de niveles mínimos de productos.

Reportes de uso de materiales y su relación con los tratamientos aplicados.

Objetivo: Mantener un control eficiente del inventario para asegurar que los insumos estén disponibles cuando se necesiten.

Requerimientos No Funcionales:**Seguridad:**

Descripción: Se deben implementar políticas de seguridad para proteger los datos sensibles de los usuarios y pacientes, así como para prevenir accesos no autorizados.

Requerimientos:

Protección de contraseñas con algoritmos de encriptación fuertes.

Control de acceso basado en roles.

Cumplimiento con normativas de seguridad de datos como la GDPR (si aplica).

Objetivo: Proteger la integridad, confidencialidad y disponibilidad de los datos.

Usabilidad:

Descripción: La aplicación debe ser intuitiva y fácil de usar, con una interfaz gráfica amigable para el personal del consultorio.

Requerimientos:

Diseños de formularios y menús fáciles de navegar.

Mensajes de error y confirmación claros.

Objetivo: Asegurar una experiencia de usuario sin complicaciones.

Escalabilidad:

Descripción: El sistema debe poder manejar un aumento en el número de usuarios, pacientes y datos sin degradar el rendimiento.

Objetivo: Asegurar que la aplicación pueda adaptarse a un mayor volumen de datos y usuarios en el futuro.

Rendimiento:

Descripción: La aplicación debe ser rápida y eficiente en la gestión de datos, sin demoras notables en las búsquedas, consultas o actualizaciones.

Objetivo: Garantizar un rendimiento óptimo durante el uso normal del sistema.

2. Identificación de Funcionalidades Existentes

La aplicación para la gestión de un consultorio odontológico cuenta con varias funcionalidades clave que aseguran el correcto manejo de las operaciones del consultorio. Estas funcionalidades se han dividido en diferentes módulos que permiten gestionar diversas áreas, incluyendo usuarios, pacientes, odontólogos, citas, facturación, inventario y generación de reportes. A continuación, se detallan las funcionalidades ya implementadas:

1. Gestión de Pacientes:

- **Descripción:** La aplicación permite gestionar de forma completa la información de los pacientes.
- **Funcionalidades:**
 - **Registro de pacientes:** Posibilidad de registrar nuevos pacientes, incluyendo sus datos personales y médicos.
 - **Edición de datos:** Capacidad para modificar la información personal y médica de los pacientes.
 - **Eliminación de pacientes:** Funcionalidad para eliminar a los pacientes del sistema de forma segura.
 - **Consulta:** Se puede buscar y consultar la información detallada de un paciente específico, incluyendo su historial médico y de citas.
- **Objetivo:** Facilitar un control eficaz sobre el registro de pacientes y mantener un historial actualizado de su información y citas.

2. Gestión de Odontólogos:

- **Descripción:** El sistema permite gestionar los datos de los odontólogos que trabajan en el consultorio.
- **Funcionalidades:**
 - **Registro de odontólogos:** Permite agregar nuevos odontólogos con su información profesional, como especialidad y disponibilidad de horarios.
 - **Edición de odontólogos:** Los datos de los odontólogos pueden ser actualizados, como cambios en la disponibilidad o actualización de información de contacto.
 - **Consulta de información:** Se puede visualizar la información completa de los odontólogos, permitiendo al administrador gestionar las asignaciones de citas.
- **Objetivo:** Gestionar de manera eficiente la plantilla de odontólogos y su disponibilidad en el sistema.

3. Gestión de Citas Odontológicas:

- **Descripción:** El módulo de citas permite registrar y administrar las citas entre los pacientes y los odontólogos.
- **Funcionalidades:**
 - **Asignación de citas:** Se pueden programar citas según la disponibilidad de los odontólogos y las preferencias de los pacientes.
 - **Edición y cancelación:** Permite modificar o cancelar las citas de acuerdo con los cambios de disponibilidad o necesidades del paciente u odontólogo.
 - **Consulta de citas:** Se pueden consultar las citas pendientes o pasadas, tanto por paciente como por odontólogo, con detalle de fechas y horarios.
- **Objetivo:** Proveer una gestión fluida y organizada de las citas entre los pacientes y los odontólogos, asegurando un control eficiente del calendario del consultorio.

4. Acceso y Autenticación:

- **Descripción:** El sistema cuenta con un mecanismo de acceso seguro mediante un módulo de login.
- **Funcionalidades:**
 - **Inicio de sesión:** Los usuarios deben ingresar sus credenciales para acceder al sistema, donde las contraseñas están encriptadas con algoritmos seguros (BCrypt o SHA-3).
 - **Control de acceso basado en roles:** Los usuarios tienen permisos según su rol (administrador, auxiliar), lo que garantiza que solo puedan acceder a las áreas correspondientes de la aplicación.
- **Objetivo:** Garantizar que solo los usuarios autorizados accedan a la aplicación y que lo hagan con los permisos adecuados a su rol.

5. Gestión de Facturación:

- **Descripción:** La aplicación maneja la facturación de los servicios proporcionados en el consultorio.
- **Funcionalidades:**
 - **Generación de facturas:** Al registrar o completar un tratamiento, se generan facturas detalladas con los costos de los servicios.
 - **Registro de pagos:** Control de los pagos realizados por los pacientes, permitiendo un seguimiento claro de las transacciones.
 - **Consulta de facturación:** Se pueden consultar facturas pasadas y pagos asociados.
- **Objetivo:** Facilitar el control de los pagos y servicios facturados, proporcionando un manejo eficiente de la parte financiera del consultorio.

6. Gestión de Inventario:

- **Descripción:** El sistema permite llevar un control de los suministros médicos y odontológicos en el consultorio.
- **Funcionalidades:**
 - **Registro de insumos:** Se pueden registrar los insumos y materiales utilizados en el consultorio.
 - **Control de stock:** Monitoreo del inventario disponible, con alertas en caso de niveles bajos de stock.

- **Reportes de uso:** Generación de reportes sobre el consumo de materiales y productos para mantener el control de los recursos.
- **Objetivo:** Asegurar un control eficiente del inventario para evitar escasez de suministros y garantizar que los tratamientos cuenten con los recursos necesarios.

7. Generación de Reportes PDF:

- **Descripción:** El sistema permite generar reportes detallados en formato PDF para diversos aspectos del consultorio.
- **Funcionalidades:**
 - **Reportes de pacientes:** Generación de reportes detallados con la información de los pacientes y sus historiales médicos.
 - **Reportes de citas:** Listado de citas agendadas, tanto por paciente como por odontólogo, con fechas y horarios.
 - **Reportes de facturación:** Generación de reportes de facturación que detallan los servicios proporcionados y pagos realizados.
- **Objetivo:** Proveer una forma de generar documentación oficial o informes para el análisis y gestión del consultorio.

3. Identificación de Funcionalidades Nuevas a Probar

Para asegurar el crecimiento continuo y la mejora de la aplicación de gestión del consultorio odontológico, se implementarán y probarán varias funcionalidades nuevas. Estas mejoras están diseñadas para optimizar la seguridad, la usabilidad y la gestión de datos en el sistema. A continuación, se detallan las funcionalidades nuevas a probar:

1. Implementación de Roles Avanzados:

- **Descripción:** Esta funcionalidad permitirá establecer un control más granular sobre los permisos de los usuarios en función de su rol específico (administrador, odontólogo, asistente).
- **Objetivo:** Verificar que cada rol tenga acceso únicamente a las funciones que le corresponden. Por ejemplo:
 - **Administrador:** Acceso completo a todas las funciones del sistema, incluyendo gestión de usuarios y reportes.

- **Odontólogo:** Acceso a la gestión de sus pacientes y citas, pero sin poder modificar datos de otros odontólogos o del sistema en general.
- **Auxiliar:** Acceso a la programación de citas y gestión básica de pacientes, pero sin permisos para realizar cambios en la facturación o configuraciones del sistema.
- **Pruebas a Realizar:**
 - Validar el acceso a diferentes funciones según el rol del usuario.
 - Probar la creación, modificación y eliminación de roles para verificar que los cambios se reflejan correctamente en los permisos de acceso.

2. Encriptación de Contraseñas SHA-3:

- **Descripción:** Se implementará el algoritmo de encriptación SHA-3 para asegurar que las contraseñas de los usuarios se gestionen de manera segura en la base de datos.
- **Objetivo:** Asegurar que las contraseñas no se almacenen en texto claro y que el proceso de autenticación sea robusto.
- **Pruebas a Realizar:**
 - Probar la creación de usuarios con contraseñas encriptadas.
 - Verificar que las contraseñas encriptadas se validen correctamente durante el inicio de sesión.
 - Realizar pruebas de seguridad para verificar que no se puedan recuperar las contraseñas originales a partir de las encriptadas.

3. Registro de la Última Sesión de Usuario:

- **Descripción:** Esta funcionalidad permitirá al sistema almacenar la fecha y hora de la última sesión de cada usuario, facilitando un seguimiento del uso de la aplicación.
- **Objetivo:** Asegurar que se registre correctamente cada vez que un usuario inicia sesión, y que esta información sea accesible para los administradores.
- **Pruebas a Realizar:**
 - Verificar que la fecha y hora se registren correctamente en la base de datos al iniciar sesión.

- Probar la visualización de esta información en el perfil de usuario y asegurarse de que se actualice cada vez que el usuario inicia sesión.

4. Cambio de Temas (Oscuro/Claro):

- **Descripción:** La aplicación implementará una opción para que los usuarios puedan alternar entre un modo oscuro y un modo claro en la interfaz gráfica.
- **Objetivo:** Mejorar la experiencia del usuario y la usabilidad, permitiendo a los usuarios elegir el tema que prefieren para trabajar.
- **Pruebas a Realizar:**
 - Verificar que el cambio de tema se aplique correctamente en toda la interfaz.
 - Probar que la selección del tema se guarde en la configuración del usuario, de modo que persista entre sesiones.

5. Manejo de la Fecha de Registro de Odontólogos:

- **Descripción:** Esta funcionalidad asegurará que la fecha de registro de un odontólogo no se modifique automáticamente cuando se actualizan otros datos del perfil.
- **Objetivo:** Mantener la integridad de la información histórica de los odontólogos en el sistema.
- **Pruebas a Realizar:**
 - Probar la actualización de otros datos de un odontólogo (como nombre, especialidad, etc.) y verificar que la fecha de registro permanezca intacta.
 - Realizar pruebas de visualización para confirmar que la fecha de registro se muestra correctamente en el perfil del odontólogo.

4. Estrategia de Pruebas

Para abordar el proceso de pruebas de manera eficiente y asegurar que la aplicación de gestión del consultorio odontológico cumpla con los estándares de calidad esperados, se definirá una estrategia de pruebas integral. Esta estrategia abarcará los siguientes tipos de pruebas:

1. Pruebas Unitarias:

- **Descripción:** Las pruebas unitarias se enfocarán en validar cada una de las funciones clave implementadas en las clases de control, como Ctrl_Usuario, Ctrl_Pacientes, Ctrl_Odontologos, y Ctrl_Citas. Estas pruebas asegurarán que cada componente funcione correctamente de manera aislada.
- **Herramienta:** Se utilizará **JUnit**, una herramienta de pruebas unitarias para Java, que permitirá:
 - Verificar la lógica de negocio en cada clase de control.
 - Probar métodos individuales para asegurarse de que devuelvan los resultados esperados y manejen correctamente las excepciones.
 - Aislar errores en funciones específicas antes de avanzar a pruebas de integración.
- **Ejemplo de Casos de Prueba:**
 - Comprobar que el método de registro de un nuevo usuario en Ctrl_Usuario retorne el resultado esperado.
 - Verificar que el método de búsqueda de pacientes en Ctrl_Pacientes devuelva los datos correctos.

2. Pruebas de Integración:

- **Descripción:** Las pruebas de integración se centrarán en evaluar cómo interactúan los diferentes módulos del sistema entre sí. Es crucial que los componentes trabajen en conjunto sin problemas.
- **Áreas de Enfoque:**
 - Validar la correcta interacción entre el módulo de gestión de pacientes y el módulo de gestión de citas.
 - Probar la comunicación entre la gestión de odontólogos y el registro de citas para asegurarse de que los odontólogos asignados se reflejen correctamente en el sistema.
- **Ejemplo de Casos de Prueba:**
 - Confirmar que al registrar una cita, se actualice correctamente la disponibilidad del odontólogo correspondiente.
 - Verificar que la información del paciente se muestre correctamente al consultar citas programadas.

3. Pruebas Funcionales:

- **Descripción:** Estas pruebas se centrarán en verificar que el sistema cumpla con los requisitos funcionales especificados, como la gestión adecuada de pacientes, citas, odontólogos, y la autenticación de usuarios.
- **Objetivo:** Asegurar que cada funcionalidad de la aplicación funcione como se espera según los requisitos del sistema.
- **Ejemplo de Casos de Prueba:**
 - Validar que el sistema permita registrar y eliminar pacientes correctamente.
 - Probar que los usuarios puedan iniciar sesión y que se aplique el control de acceso basado en roles.

4. Pruebas de Seguridad:

- **Descripción:** Las pruebas de seguridad evaluarán la implementación de la encriptación de contraseñas y asegurarán que la autenticación y el acceso al sistema estén protegidos.
- **Áreas de Enfoque:**
 - Comprobar que las contraseñas se almacenen de manera segura usando SHA-3.
 - Validar que los usuarios no autorizados no puedan acceder a funciones restringidas.
- **Ejemplo de Casos de Prueba:**
 - Probar la resistencia de la autenticación al intentar acceder con contraseñas incorrectas.
 - Evaluar el acceso a la aplicación con diferentes roles y asegurar que cada rol tenga las restricciones adecuadas.

5. Pruebas de Rendimiento:

- **Descripción:** Las pruebas de rendimiento se enfocarán en medir la capacidad del sistema para manejar cargas de trabajo altas, especialmente en momentos de alta demanda, como consultas simultáneas.
- **Objetivo:** Asegurar que la aplicación mantenga tiempos de respuesta aceptables y no falle bajo condiciones de carga pesada.

- **Ejemplo de Casos de Prueba:**

- Realizar pruebas de carga simulando múltiples usuarios accediendo a la aplicación al mismo tiempo.
- Medir el tiempo de respuesta al cargar registros de pacientes o citas en la base de datos.

6. Pruebas de Usabilidad:

- **Descripción:** Las pruebas de usabilidad se centrarán en la experiencia del usuario al interactuar con la interfaz gráfica de la aplicación.
- **Objetivo:** Asegurar que la interfaz sea intuitiva, fácil de navegar y que cumpla con las expectativas de los usuarios finales, que en este caso son los empleados del consultorio.
- **Ejemplo de Casos de Prueba:**
 - Realizar sesiones de prueba con usuarios reales para obtener retroalimentación sobre la navegación y la disposición de los elementos en la interfaz.
 - Evaluar la claridad de los mensajes de error y la facilidad con la que los usuarios pueden completar tareas como registrar pacientes o gestionar citas.

5. Entornos de Trabajo Requeridos

Software:

- **Sistema Operativo:** Windows 10 o superior.
- **IDE de desarrollo:** NetBeans 12.5 o IntelliJ IDEA.
- **Java SDK:** JDK 17 o superior.
- **Base de Datos:** MySQL 8.0.
- **Herramienta de pruebas:** JUnit 5.0 para pruebas unitarias, Selenium para pruebas de interfaz gráfica.

Hardware:

- **Procesador:** Intel i5 o superior.

- **RAM:** 8 GB como mínimo.
- **Disco Duro:** 256 GB de almacenamiento disponible.
- **Resolución de pantalla:** Mínimo 1366x768 para pruebas de interfaz gráfica.

6. Metodologías y Procedimientos

Metodología de Pruebas:

Se utilizará la metodología “Agile Testing” en conjunto con “TDD (Test-Driven Development)”. Esto significa que se escribirán las pruebas antes de implementar nuevas funcionalidades o módulos, asegurando que el código cumpla con los requisitos desde el principio.

Procedimientos de Pruebas:

1. “Escribir pruebas unitarias y de integración” utilizando JUnit para cada nuevo módulo desarrollado.
2. “Realizar pruebas manuales funcionales” sobre las funcionalidades principales como gestión de pacientes, odontólogos, citas y reportes.
3. “Ejecución de pruebas de rendimiento” mediante la simulación de usuarios concurrentes accediendo al sistema.
4. “Documentar los resultados” de las pruebas en un informe detallado.

7. Cronograma y Planificación de las Pruebas

Actividad	Duración Estimada	Responsable	Fecha de Inicio	Fecha de Fin
Análisis de Requerimientos	2 días	Yuliet Faizuli Pachon		
Diseño de Pruebas Unitarias (JUnit)	3 días	Equipo de desarrollo		
Ejecución de Pruebas Unitarias	2 días	Jorge Miller Gutierrez		
Diseño de Pruebas de Integración	3 días	Equipo de desarrollo		
Ejecución de Pruebas de Integración	2 días	Equipo de desarrollo		
Pruebas Funcionales Manuales	4 días	Equipo de pruebas		

Pruebas de Seguridad	2 días	Jorge Miller Gutierrez		
Pruebas de Rendimiento	3 días	Equipo de pruebas		
Documentación de Resultados	2 días	Yuliet Faizuli Pachon		

8. Registro de Pruebas (Diseño de Artefactos)

Para llevar un control exhaustivo del proceso de pruebas y asegurar que todas las funcionalidades de la aplicación sean evaluadas correctamente, se utilizará una **matriz de pruebas**. Este artefacto servirá como un registro estructurado que permitirá documentar cada prueba realizada, sus resultados y cualquier observación pertinente. A continuación se detalla la estructura de la matriz de pruebas:

Estructura de la Matriz de Pruebas

ID	Funcionalidad probada	Descripción de la prueba	Resultado esperado	Resultado obtenido	Estado (Aprobado/Rechazado)	Comentarios
001	Registro de Pacientes	Registrar un paciente con todos los campos llenos	Paciente registrado correctamente	Paciente registrado correctamente	Aprobado	Ninguno
002	Gestión de Citas	Programar cita para un paciente	Cita registrada correctamente	Error al guardar cita	Aprobado	Ninguno
003	Encriptación de Contraseñas	Probar el registro de un usuario con contraseña encriptada	Contraseña almacenada de forma segura	Contraseña almacenada de forma segura	Aprobado	Ninguno
004	Autenticación	Iniciar sesión con credenciales válidas	Acceso concedido	Acceso concedido	Aprobado	Ninguno
005	Gestión de Odontólogos	Registrar un odontólogo con todos los campos llenos	Odontólogo registrado correctamente	Odontólogo registrado correctamente	Aprobado	Ninguno
006	Gestión de Pacientes	Consultar la lista de pacientes registrados	Listado de pacientes correcto	Listado de pacientes correcto	Aprobado	Ninguno
007	Generación de Reportes	Generar un reporte de pacientes en PDF	Reporte generado correctamente	Reporte generado correctamente	Aprobado	Ninguno
008	Cambio de Temas	Cambiar de tema oscuro a claro	Interfaz gráfica en modo claro	Interfaz gráfica en modo claro	Aprobado	Ninguno

009	Registro de Última Sesión	Almacenar y mostrar la fecha y hora de la última sesión	Fecha y hora almacenadas correctamente	Fecha y hora almacenadas correctamente	Aprobado	Ninguno
010	Pruebas de Usabilidad	Evaluar la facilidad de uso del menú principal	Menú intuitivo y fácil de navegar	Menú intuitivo y fácil de navegar	Aprobado	Ninguno

Descripción de los Campos:

- **ID de la prueba:** Un identificador único para cada prueba, facilitando el seguimiento y referencia.
- **Funcionalidad probada:** La funcionalidad específica del sistema que se está evaluando.
- **Descripción de la prueba:** Una breve explicación de lo que se pretende probar.
- **Resultado esperado:** La respuesta que se espera del sistema si la funcionalidad se implementa correctamente.
- **Resultado obtenido:** Lo que realmente ocurrió al ejecutar la prueba, documentando cualquier discrepancia con el resultado esperado.
- **Estado (Aprobado/Rechazado):** Indica si la prueba fue exitosa (Aprobado) o si falló (Rechazado).
- **Comentarios:** Observaciones adicionales, como recomendaciones para corregir errores o notas sobre el comportamiento del sistema.

9. Selección de Herramientas de Pruebas

Las herramientas seleccionadas para implementar las pruebas son:

1. **JUnit:** Utilizada para las pruebas unitarias y de integración. Permite verificar que los métodos y clases individuales funcionen correctamente.
2. **Selenium:** Herramienta seleccionada para las pruebas de interfaz gráfica. Automáticamente interactúa con la GUI y valida la usabilidad del sistema.
3. **JMeter:** Se empleará para realizar pruebas de rendimiento, simulando múltiples usuarios accediendo simultáneamente a la aplicación para detectar cuellos de botella.

10. Identificación de Riesgos y Contingencias

Durante la ejecución de las pruebas de software, se han identificado varios riesgos que pueden afectar el cronograma, los resultados y la calidad del proceso de prueba. A continuación, se presenta una tabla con los riesgos identificados, su impacto, probabilidad y un plan de contingencia sugerido para cada uno.

Riesgo	Impacto	Probabilidad	Plan de Contingencia
Fallo en la base de datos durante las pruebas	Retraso en la ejecución de las pruebas	Media	Realizar backups frecuentes de la base de datos y contar con un entorno de prueba adicional.
Incompatibilidad con hardware o software	Imposibilidad de realizar pruebas en ciertos entornos	Baja	Definir con antelación los entornos requeridos y garantizar que todos los equipos cumplen con los requisitos.
Retrasos en la disponibilidad de los testers	Retraso en la entrega de resultados	Media	Crear un equipo de respaldo que pueda asumir responsabilidades en caso de que el equipo principal no esté disponible.
Errores de configuración del entorno de pruebas	Fallos en la ejecución de las pruebas	Baja	Documentar detalladamente la configuración del entorno de pruebas para evitar errores durante la instalación.
Nuevos cambios en los requerimientos de software	Incremento en el alcance de las pruebas	Alta	Adoptar una metodología Agile para facilitar la integración de cambios sin alterar el cronograma de las pruebas.
Desconocimiento de herramientas de pruebas	Problemas durante la ejecución y análisis	Media	Realizar capacitaciones en las herramientas seleccionadas antes de iniciar el proceso de pruebas.

Descripción de los Riesgos:

1. Fallo en la base de datos durante las pruebas:

- **Impacto:** Puede causar retrasos significativos en el cronograma de pruebas si se pierde información crítica.
- **Probabilidad:** Media. Aunque no es común, los fallos pueden ocurrir debido a errores humanos o problemas de sistema.

- **Plan de Contingencia:** Implementar un sistema de backups automáticos para asegurar que la información no se pierda, además de tener un entorno de prueba separado para minimizar el riesgo.

2. Incompatibilidad con hardware o software:

- **Impacto:** Puede llevar a que no se puedan realizar pruebas en ciertas configuraciones, limitando la cobertura de las pruebas.
- **Probabilidad:** Baja. Se puede mitigar con una planificación adecuada.
- **Plan de Contingencia:** Realizar un análisis de compatibilidad de hardware y software antes de comenzar las pruebas.

3. Retrasos en la disponibilidad de los testers:

- **Impacto:** La entrega de resultados se verá afectada, lo que puede llevar a demoras en el lanzamiento del software.
- **Probabilidad:** Media. Puede ocurrir debido a imprevistos personales o laborales.
- **Plan de Contingencia:** Tener un equipo de respaldo listo para intervenir cuando sea necesario.

4. Errores de configuración del entorno de pruebas:

- **Impacto:** Puede causar fallos en la ejecución de pruebas, invalidando resultados y retrasando el cronograma.
- **Probabilidad:** Baja. Con una buena preparación se puede minimizar.
- **Plan de Contingencia:** Crear una documentación clara sobre la configuración del entorno para asegurar que todos los testers sigan el mismo procedimiento.

5. Nuevos cambios en los requerimientos de software:

- **Impacto:** Aumenta el alcance y la complejidad de las pruebas, lo que puede comprometer los plazos establecidos.
- **Probabilidad:** Alta. Es común que los requerimientos cambien durante el desarrollo.
- **Plan de Contingencia:** Utilizar una metodología Agile que permita ajustes y cambios sin afectar significativamente el cronograma de pruebas.

6. Desconocimiento de herramientas de pruebas:

- **Impacto:** Puede provocar problemas durante la ejecución y análisis de las pruebas, llevando a resultados erróneos o incompletos.

- **Probabilidad:** Media. Los testers pueden no estar familiarizados con las herramientas elegidas.
- **Plan de Contingencia:** Organizar sesiones de capacitación sobre las herramientas de pruebas seleccionadas para garantizar que todos los testers estén adecuadamente preparados.

11. Cronograma Detallado de Pruebas

A continuación, se presenta el cronograma detallado que define las fechas exactas para la ejecución de las diferentes pruebas, con el objetivo de asegurar que todas las actividades de prueba se completen dentro del plazo estipulado:

Fase de Prueba	Duración	Fecha de Inicio	Fecha de Fin
Análisis de Requerimientos	2 días		
Diseño de Pruebas Unitarias	3 días		
Ejecución de Pruebas Unitarias	2 días		
Pruebas de Integración	3 días		
Pruebas Funcionales	4 días		
Pruebas de Seguridad	2 días		
Pruebas de Rendimiento	3 días		
Pruebas de Usabilidad	2 días		
Documentación de Resultados	2 días		

Este cronograma puede ajustarse de acuerdo con los cambios que se presenten durante el desarrollo y las pruebas de software.

12. Artefactos de Registro

El proceso de pruebas incluirá la creación de artefactos que permitan registrar cada una de las pruebas realizadas, el estado de las mismas y los resultados obtenidos. Los principales artefactos son:

- **Matriz de Pruebas:** Documento que lista todas las pruebas ejecutadas con su ID, funcionalidad probada, resultado esperado y resultado obtenido.
- **Registro de Incidencias:** Donde se documentarán los errores encontrados durante las pruebas, su severidad, y las acciones tomadas para corregirlos.

- **Informe de Resultados:** Documento final que resume los resultados generales de las pruebas, indicando cuántas pruebas fueron aprobadas, rechazadas y pendientes de revisión.

13. Conclusiones

El plan de pruebas descrito en este documento tiene como objetivo garantizar que la ****aplicación de escritorio para la gestión de un consultorio odontológico** “ funcione correctamente, cumpla con los requerimientos y sea fácil de usar por los usuarios finales. Mediante la implementación de pruebas unitarias, de integración, funcionales, de seguridad y de rendimiento, se espera identificar y corregir posibles errores antes del lanzamiento final.

Se ha establecido una estrategia clara que cubre tanto las funcionalidades actuales como las nuevas, asegurando que se lleven a cabo pruebas exhaustivas en todos los módulos clave, incluyendo la gestión de pacientes, citas y odontólogos. Además, se ha prestado atención especial a la seguridad de la información, mediante la implementación de encriptación para contraseñas y roles de acceso.

El uso de herramientas como JUnit, Selenium y JMeter ayudará a automatizar y asegurar la calidad de las pruebas, reduciendo el riesgo de errores en la fase de producción. Se han identificado los riesgos y contingencias necesarios para mitigar cualquier problema durante la ejecución del plan, asegurando un proceso fluido y coordinado, antes de su implementación final.