

Plan de Pruebas:



APP MI CESFAM

Fecha: 27 / 09 / 2023



Índice



Historial de Versiones	3
Información del Proyecto	3
Aprobaciones	3
Introducción.....	4
Enfoque de Agile Testing	4
Tipos de prueba a realizar.....	5
Pruebas Unitarias :	5
Pruebas de Regresión:.....	5
Pruebas de Integración:.....	5
Pruebas Funcionales:	6
Pruebas de Usabilidad:.....	6
Pruebas de Seguridad:.....	6
Estrategias de prueba en cada Sprint	8
Sprint 1 – 3: Diseño preliminar, estructura y funciones básicas.....	8
Sprint 4: Reserva de horas	8
Sprint 5: Farmacia CESFAM.....	8
Sprint 6 – 7: Notificaciones y revisión de exámenes.....	8
Sprint 8: Integración final	8
Herramientas a utilizar.....	9
Conclusión	10



Historial de Versiones

Fecha	Versión	Autor	Organización	Descripción
27/09/2024	1.0	Álvaro Salinas	APP Mi SpA.	Primera versión de plan de pruebas
06/11/2024	2.0	Joaquín Mora	APP Mi SpA.	Inclusión de casos de pruebas

Información del Proyecto

Empresa / Organización	App Mi SpA.
Proyecto	APP “Mi CESFAM”
Fecha de preparación	28/08/2024
Cliente	Corporación municipal de Peñalolén
Patrocinador principal	CITT
Gerente / Líder de Proyecto	Daniel Belmar
Gerente / Líder de Pruebas de Software	Equipo QA

Aprobaciones

Nombre y Apellido	Cargo	Departamento	Fecha	Firma
Daniel Belmar	Jefe de Proyectos	TI	27/09/2024	<i>Daniel Belmar</i>
Álvaro Salinas	Desarrollador / QA	TI	27/09/2024	<i>Alvaro Salinas</i>
Vicente Vargas	DBA / QA	TI	27/09/2024	<i>Vicente Vargas</i>
Joaquín Mora	Desarrollador / QA	TI	27/09/2024	<i>Joaquin Mora</i>

Introducción

El proyecto MI CESFAM consiste en el desarrollo de una aplicación móvil para dispositivos Android, iOS y WEB, cuyo objetivo es mejorar la eficiencia de la gestión en centros médicos CESFAM y mejorar la experiencia de atención de los pacientes. La aplicación permite reservar citas, realizar consultas sobre medicamentos, recibir notificaciones sobre vacunas, y revisar exámenes médicos, entre otras funcionalidades.

En este informe se detalla el enfoque de Agile Testing aplicado en el proyecto, incluyendo los tipos de pruebas realizadas, la estrategia de testing en cada sprint, las herramientas utilizadas y la participación del equipo de desarrollo y testing.

Enfoque de Agile Testing

El proceso de Agile Testing se lleva a cabo en paralelo con el desarrollo durante cada sprint, asegurando que las pruebas se realicen de manera continua y adaptativa. Las pruebas se dividen en ciclos cortos para ajustarse al marco de trabajo SCRUM, utilizando un enfoque de test-driven development (TDD) en algunas funcionalidades críticas y automated testing para las regresiones.

Objetivos del Agile Testing:

- Identificar y corregir defectos de manera temprana en el ciclo de desarrollo.
- Garantizar que las funcionalidades desarrolladas en cada sprint cumplan con los criterios de aceptación definidos por los stakeholders.
- Validar que la aplicación sea segura, eficiente y fácil de usar para los usuarios finales.
- Asegurar la integración continua del código mediante pruebas automatizadas.

Tipos de Pruebas a Realizar

Pruebas Unitarias :

Estas pruebas se realizarán a nivel de código, probando unidades de código aisladas para asegurarse de que cada componente funcione de manera correcta .

- A nivel general, con la implementación de la base de datos Firebase, esto limita hasta cierto punto la ejecución de módulos totalmente independientes debido a la conexión con registro e identificación de usuarios.
- Al ingresar a otra pestaña mientras no se inicie sesión, la página no cargará y no mostrará información o layout alguno.
- Cada pestaña se despliega completamente, con los botones, imágenes y textos asociados.
- Para la solicitud de recetas, se genera un archivo pdf asociado que se muestra en la aplicación.
- Para la solicitud de medicamentos, se genera la solicitud del mismo y se resta del conteo total de la misma.
- Al visualizar la solicitud de horas médicas, el calendario asociado está actualizado al año, mes y día correspondiente.
- Al momento de generar una notificación, captura el correo electrónico del usuario conectado para enviar el mensaje.

Pruebas de Regresión:

Cuando se integran nuevas funcionalidades, se realizan pruebas de regresión automatizadas para que así garanticemos que las funcionalidades que ya existen no se vean afectadas.

- Inicio de sesión con datos correctos, y que acceda a las vistas correspondientes
- Inicio de sesión con datos incorrectos, tanto contraseña como nombre de usuario, y que rechace el acceso a cualquier vista
- Generación de reserva de hora nueva, que se envíe y actualice en el sistema, notificaciones incluidas
- Generación de reserva de hora sobre una ya existente, que no se envíe/sobrescriba, y notifique sobre la invalidez de la acción
- Solicitud de receta médica, con archivo pdf asociado que contenga la información de dicho usuario
- Solicitud de examen médico disponible, generando notificación al correo electrónico asociado de la misma e información incluida.

Pruebas de Integración:

Con estas pruebas podemos validar como el sistema interactúa entre sí, ya sea probando módulos o componentes del sistema y así asegurando que se comuniquen de manera correcta.

- Al iniciar sesión en Login, se identifica la información de usuario y roles asociados para mostrar las pestañas y accesos asociados en la aplicación.

- Las solicitudes realizadas por usuarios pacientes pueden ser visualizadas desde las vistas de médicos/farmacéuticos, según corresponda.
- Al realizar una solicitud de medicamento, se resta del total disponible de dicho ítem en la base de datos, fuera de modificaciones directas/manuales por usuarios con los permisos.

Pruebas Funcionales:

Las pruebas funcionales se enfocan en verificar que cada funcionalidad de la aplicación cumpla con los requerimientos especificados.

- Cada botón para navegar entre pestañas funciona y dirige a la página indicada.
- Una vez se inicia sesión exitosamente, la información asociada se mantiene entre páginas para mostrar la información a la que se tiene acceso.
- Los roles de usuario permiten o limitan la visualización de información o pestañas distintas.
- Los datos de caché para navegar se eliminan al cerrar la sesión en el dispositivo.
- El sitio se muestra de forma similar entre la versión web de escritorio y móvil (app).

Pruebas de Usabilidad:

Estas pruebas verificarán que la aplicación sea intuitiva y fácil de usar para los pacientes y personal del CESFAM

- Tanto el Header como el Footer no se hacen intrusivos ni bloquean elementos de las pestañas.
- Elementos de uso y de diseño se encuentran centrados y no se salen según el tamaño de la vista.
- Imágenes están relacionadas a cada pestaña.
- Botones tienen un tamaño y diseño fácil de diferenciar y notar.
- Los colores asociados tienen cohesión entre sí y no se hacen molestos o caóticos de visualizar.
- Se contrasta adecuadamente el texto con el fondo general y botones.
- Las vistas de roles muestran adecuadamente la información general, como botones, íconos de navegación y data de usuario asociada.

Pruebas de Seguridad:

Estas pruebas se aplican para identificar las vulnerabilidades que puede tener el sistema con el manejo de los datos sensibles, citas médicas y resultados de exámenes, etc.

- La información de caché tiene un token de seguridad para prevenir múltiples accesos simultáneos.
- Se genera un caché para contener la información de usuario correspondiente.
- No hay forma para usuarios sin roles tipo administrador o similar para cambiar sus roles o accesos.

- Cada vista de rol permite a cada usuario ver la información que le corresponde según su posición, y no la de terceros a menos que sea un operativo médico o desarrollador.
- Existen restricciones para el ingreso de contraseña, como límite de intentos, símbolos o caracteres inválidos, mensajes sin especificar de la causa de error, entre otros.
- La información externa desplegada, como exámenes, recetas y otros, debe estar asociada a un correo electrónico como mínimo para tener un respaldo de identificación a la cual enviarla, fuera de la vista como app.
- Para el ingreso de contraseña, se restringe el ingreso de lógica de código para acceder ilegalmente a los contenidos (1=true y contenidos similares).

Estrategias de Pruebas en cada Sprint

En cada sprint, vamos a seguir una estrategia que nos ayude a asegurar la calidad de cada Sprint. De esta forma, cada iteración y el proyecto en su conjunto se efectuarán de manera exitosa.

Sprints 1 – 3: Diseño preliminar, estructura y funciones básicas

Enfoque: Pruebas unitarias y de integración para asegurarnos de que la base de la app esté bien definida. Se realizan pruebas manuales de la navegación.

Entregables: Pruebas de las primeras pantallas y navegación básica de la app, como inicio de sesión y menú principal.

Sprint 4: Reserva de Horas

Enfoque: Pruebas funcionales para validar que el sistema realiza la reserva de horas funcione de la mejor manera.

Entregables: Pruebas automatizadas para garantizar que los usuarios puedan seleccionar fechas y horas sin problemas.

Sprint 5: Farmacia CESFAM

Enfoque: Pruebas de Integración para verificar que el canje de medicamentos u otros se comunique de manera correcta con el inventario de la farmacia.

Entregables: Validar el flujo de la reserva para el canje de medicamentos u otros, desde la selección hasta la confirmación de que si está disponible o no en el CESFAM.

Sprints 6 – 7: Notificaciones y Revisión de Exámenes

Enfoque: Probar las notificaciones push y funcionalidad de revisión de exámenes. Se realizarán pruebas de carga para así asegurarnos que el sistema pueda manejar múltiples usuarios que reciben la notificación.

Entregables: Verificar el envío y recepción de las notificaciones

Sprint 8: Integración Final

Enfoque: Realizaremos Pruebas de regresión completas, prueba de rendimiento y pruebas de seguridad para así asegurarnos que la aplicación funcione de la mejor manera y no presente vulnerabilidades.

Entregables: Validar la APP completa y asegurando la integración de todos los componentes de manera exitosa y de calidad.

Herramientas a Utilizar

- Selenium: Para pruebas automatizadas de regresión.
- Postman: Para pruebas de API en el backend.
- Firebase Test Lab: Para ejecutar pruebas en dispositivos reales Android e IOS.

Conclusión

El enfoque Agile Testing será importante para asegurarnos de la calidad de nuestro sistema, debemos ser capaces de ajustar lo necesario y desarrollar de manera correcta para así obtener un producto viable y que sea sostenible en el tiempo.

El tiempo está en nuestra contra, y la posibilidad de ejecutar estas pruebas en distintos periodos de tiempo y tanto por secciones como en conjunto nos permite, en teoría, tener todo listo antes de la fecha de examen y así tener nuestro sistema listo para producción y poder enfocarnos en pulir detalles menores que no afecten la funcionalidad de la misma.