



**FACULTAD DE CIENCIAS E INGENIERÍA
CARRERA DE INGENIERÍA DE SOFTWARE**

TEMA:

Agendamiento de citas personales

AUTORES:

Aucancela Lascano Ronny Polivio

Aguilar Yagual Brithanny Stefania

Bustamante Tomalá José Eduardo

Franco Piñan Alinton Andrés

Tumbaco Alvarado Katherine Fabiana

ASIGNATURA:

Introducción a la Ingeniería de Software

DOCENTE:

Guevara Serrano Jorge Dumar

FECHA DE ENTREGA:

5 de noviembre del 2025

PERIODO:

agosto 2025 a diciembre 2025

MILAGRO-ECUADOR

Agendamiento de citas personales.

➤ **Descripción del sistema:**

La *Agenda Personal de Citas* es una aplicación digital diseñada para ayudar a los usuarios a gestionar y llevar un control eficiente de sus compromisos y actividades personales. Su función principal es permitir el registro, visualización y organización de citas con información detallada, como la fecha, hora y descripción del evento.

El sistema ofrece una interfaz sencilla e intuitiva que facilita el control y seguimiento de los compromisos personales, generando como resultado una lista ordenada de todas las citas agendadas. De esta manera, el usuario puede mantener una planificación clara y evitar olvidos o confusiones en su calendario diario.

Generalmente, este tipo de sistemas es utilizado por personas que buscan optimizar su tiempo y mejorar la organización de sus actividades cotidianas. No obstante, como todo sistema informático, puede presentar limitaciones y aspectos susceptibles de mejora que influyen en su rendimiento o en la experiencia del usuario.

➤ **Investigación, conceptos teóricos sobre el mantenimiento de software según Sommerville y Pressman.**

• **¿Qué es el mantenimiento de software?**

Cuando un software ya está terminado y entregado, no significa que ya no se toca nunca más. Igual que una casa, un carro o un celular, el software necesita cuidados.

El mantenimiento de software es todo lo que hacemos después de entregar el sistema para que siga funcionando bien, se adapte a nuevas necesidades y no se “dañe” con el tiempo.

• **Tipos de mantenimiento:**

Tipo	Para que sirve	Explicación
Correctivo	Arreglar problemas	Cuando aparecen errores o fallos inesperados y hay que corregirlos. Ej.: se cae la aplicación y hay que repararla.
Adaptativo	Ajustarse a cambios externos	El software debe funcionar en nuevos entornos o tecnologías. Ej.: actualizo mi programa porque Windows lanzó una nueva versión.
Perfectivo	Mejorar y agregar cosas	Se agregan funciones nuevas o se mejora el rendimiento para que el usuario esté más feliz. Ej.: mejorar la velocidad o añadir un botón que pidieron.
Preventivo	Evitar problemas futuros	Se hacen ajustes para prevenir errores y mantener el código sano. Ej.: limpiar código viejo o reorganizar funciones.

- **Costos del mantenimiento:**

Mantener el software suele costar mucho más que crearlo. A lo largo de la vida del sistema, el mantenimiento puede representar entre el 60% y el 80% del costo total.

- **Por qué cuesta tanto:**

Porque el software debe seguir funcionando sin importar cuantos cambios aparezcan (nuevas reglas, nuevos dispositivos, nuevas necesidades).

Porque entender y modificar sistemas viejos puede ser complicado, porque corregir un problema tarde es más caro que prevenirlo.

Cosas que aumentan el costo	Cosas que reducen el costo
Mala documentación	Buen diseño desde el principio
Código difícil de entender	Documentación clara
Cambios constantes de requisitos	Código limpio y organizado
Tecnología antigua o desactualizada	Automatizar pruebas y procesos

- **Etapas del mantenimiento según Sommerville y Pressman:**

Sommerville	Pressman
Sommerville propone este flujo: Ver qué hay que cambiar Diseñar y modificar el programa Actualizar la documentación Probar que todo sigue funcionando Es como: revisar → cambiar → documentar → probar	Pressman menciona prácticamente lo mismo, pero más detallado: Identificar el problema o solicitud Analizar el impacto del cambio Diseñar la solución Implementar (programar) Probar Entregar al usuario Registrar para historial

➤ **Anализar el caso elegido, identificando situaciones problemática o área de mejora.**

Se identificaron dos áreas problemáticas principales en el sistema actual de **Agendamiento de citas personales**:

- **Situación 1 – Falta de sincronización entre dispositivos**

Actualmente, la aplicación guarda las citas en el almacenamiento local del dispositivo. Esto significa que, si el usuario cambia de teléfono, formatea su equipo o reinstala la aplicación, pierde todos sus datos.

Además, no puede consultar ni editar sus citas desde otros dispositivos, lo cual **limita la portabilidad y accesibilidad** del sistema.

Impacto: pérdida de información, baja disponibilidad y mala experiencia del usuario.

- **Situación 2 – Recordatorios poco configurables**

El sistema envía notificaciones o recordatorios en un tiempo fijo antes del evento, sin permitir personalización.

Esto no se ajusta a las necesidades de todos los usuarios, que podrían requerir avisos con diferente anticipación o repetición (por ejemplo, cada 15 minutos, una hora antes o el día anterior).

Impacto: disminuye la utilidad del sistema como asistente personal, reduciendo la satisfacción y eficiencia del usuario.

- **Situación 3 – Falta de seguridad y protección de datos**

Actualmente, el sistema no cuenta con mecanismos avanzados de protección para la información del usuario.

Las citas, nombres o detalles personales pueden quedar expuestos si el dispositivo es compartido o se accede a los archivos sin autenticación.

Además, no se aplica cifrado en el almacenamiento local, lo que **representa un riesgo de privacidad**.

Impacto: vulnerabilidad ante pérdida o robo de información personal, incumplimiento de buenas prácticas de seguridad y menor confianza del usuario.

➤ **Determinar los tipos de mantenimientos aplicables:**

- **Mantenimiento correctivo:**

Aplicación: necesario cuando se detecten fallos en las funciones principales del sistema.

Possible problema	Requerimiento asociado
Error al guardar una cita o el sistema no muestra confirmación	RF1 – Registro de citas
Fallo al mostrar el calendario o aparece vacío	RF2 – Visualización del calendario
Notificaciones que no se envían en el tiempo establecido	RF4 – Notificaciones y recordatorios

- **Mantenimiento Perfectivo:**

Aplicación: mejora del sistema para agregar funcionalidades o aumentar la usabilidad.

Mejora posible	Justificación
Añadir opción de sincronización con Google Calendar o iCloud	Extiende la utilidad y alcance del sistema
Agregar filtros de búsqueda por palabras clave o tipo de cita	Complementa RF5 – Historial de citas
Implementar modo oscuro o personalización visual	Mejora la experiencia de usuario (usabilidad)

- **Mantenimiento Adaptativo:**

Aplicación: cambios necesarios cuando el entorno tecnológico se modifica.

Cambio del entorno	Impacto
Actualización de versión del sistema operativo (Android, Windows, etc.)	La aplicación podría dejar de funcionar si no se adapta
Migración del almacenamiento local a la nube (por ejemplo, Firebase, OneDrive)	Requiere ajustes de arquitectura y medidas de seguridad
Compatibilidad con nuevos dispositivos móviles o Tablet	La interfaz debe adaptarse al nuevo hardware

- **Mantenimiento Preventivo**

Aplicación: modificaciones internas para prevenir fallos futuros y mejorar la mantenibilidad del código.

Acción preventiva	Relación con RNF
Optimizar consultas para evitar lentitud al aumentar la cantidad de citas	RNF1 – Tiempo de respuesta del sistema
Revisar mecanismos de cifrado periódicamente para evitar vulnerabilidades	RNF3 – Protección de datos personales
Depurar funciones duplicadas y reducir la complejidad del código	Mejora la mantenibilidad del software

➤ **Nombrar y describir el cambio funcional.**

- **Cambio funcional propuesto:** Implementación de sincronización en la nube para respaldo y acceso multiplataforma.

Descripción del cambio: Se propone añadir la funcionalidad de sincronización automática de citas en la nube, permitiendo que el usuario pueda acceder a su agenda desde distintos dispositivos y mantener sus datos respaldados de manera segura.

- **Qué mejora o corrige**

Problema actual	Mejora propuesta
La aplicación almacena las citas localmente, lo que limita su uso a un solo dispositivo y existe riesgo de pérdida de información.	Se garantiza acceso desde cualquier dispositivo y respaldo automático, evitando pérdida de datos.

- **Implementación**

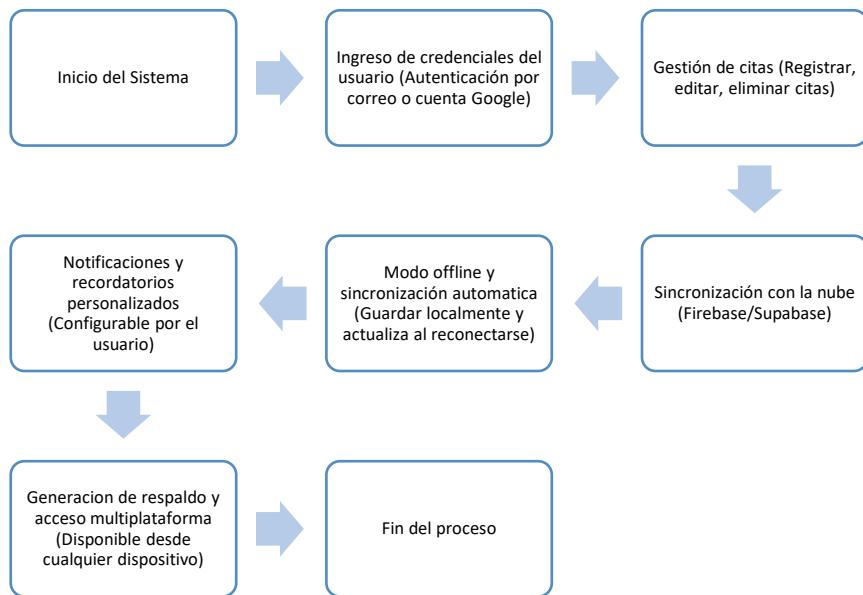
- ✓ Integrar un servicio de almacenamiento en la nube (Firebase, Supabase o similar).
- ✓ Guardar citas en una base de datos remota vinculada a una cuenta de usuario.
- ✓ Implementar autenticación (correo/Google/Apple ID).
- ✓ Sincronización en tiempo real para mantener datos consistentes en todos los dispositivos.
- ✓ Modo offline: si no hay conexión, las citas se almacenan localmente y se sincronizan después.

- **Impacto en mantenibilidad o calidad**

Aspectos	Impacto
Mantenibilidad	Mejora, ya que los datos centralizados facilitan gestión e implementación futura de nuevas funciones.
Calidad del producto	Incrementa fiabilidad y disponibilidad de la información.
Experiencia del usuario	Significativa mejora, pues puede acceder a sus citas en diferentes dispositivos y evitar pérdida de datos.
Escalabilidad	Permite futuras mejoras, como sincronización con calendarios externos o recordatorios por correo.

➤ **Esquema o diagrama:**

Diagrama del Sistema de Agendamiento de Citas Personales mejorado con el módulo de sincronización en la nube y acceso multiplataforma.



➤ **Reflexión final:**

El desarrollo de este trabajo permitió comprender de forma práctica por qué el mantenimiento de software es una parte esencial dentro del ciclo de vida de cualquier sistema. A través del análisis de la “Agenda de citas personales”, se identificaron dificultades reales que afectan el funcionamiento, la seguridad y la experiencia del usuario. Esto evidenció que un sistema no termina cuando se crea, sino que requiere atención constante para seguir siendo útil y confiable.

Durante la investigación se aplicaron los tipos de mantenimiento propuestos por Sommerville y Pressman correctivo, adaptativo, perfectivo y preventivo, comprobando que cada uno cumple un papel importante para garantizar la estabilidad del sistema. La implementación de la sincronización en la nube se destacó como una mejora clave, ya que

facilita el acceso a la información, protege los datos y eleva la satisfacción de los usuarios. En conclusión, mantener un sistema no solo implica resolver fallos, sino también planificar su evolución. Este trabajo reforzó la importancia de cuidar la calidad y sostenibilidad del software mediante buenas prácticas de desarrollo, con el fin de asegurar soluciones más seguras, duraderas y adaptables a las necesidades de las personas.