

UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE SOFTWARE



TEMA:

DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL CONTROL Y GESTIÓN DE ASISTENCIA DE LOS EMPLEADOS, EN EL GAD MUNICIPAL DE COTACACHI UTILIZANDO METODOLOGÍAS ÁGILES Y TECNOLOGÍA DE DESARROLLO NATIVO.

Trabajo de grado previo a la obtención del título de Ingeniero de Software presentado ante la ilustre Universidad Técnica del Norte.

AUTOR:

Carlos Fabricio Pozo Chulde

DIRECTOR:

Ing. MacArthur Ortega Bustamante, MSc

Ibarra-Ecuador

2026



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	0402092167		
APELLIDOS Y NOMBRES:	Pozo Chulde Carlos Fabricio		
DIRECCIÓN:	Cristóbal Colón – Montufar - Carchi		
EMAIL:	cfpozoc@utn.edu.ec		
TELÉFONO FIJO:		TELÉFONO MÓVIL:	0993085021

DATOS DE LA OBRA	
TÍTULO:	DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL CONTROL Y GESTIÓN DE ASISTENCIA DE LOS EMPLEADOS, EN EL GAD MUNICIPAL DE COTACACHI UTILIZANDO METODOLOGÍAS ÁGILES Y TECNOLOGÍA DE DESARROLLO NATIVO.
AUTOR (ES):	POZO CHULDE CARLOS FABRICIO.
FECHA DE APROBACIÓN: DD/MM/AAAA	06/03/2026
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TÍTULO POR EL QUE OPTA:	INGENIERÍA EN SOFTWARE
ASESOR /DIRECTOR:	Ing. MacArthur Ortega Bustamante, MSc



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

2. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 06 días del mes de marzo de 2026.

EL AUTOR:

.....

Nombre: Pozo Chulde Carlos Fabricio

CERTIFICACIÓN DIRECTOR

Ibarra 06 de marzo del 2026

CERTIFICACIÓN DIRECTOR DEL TRABAJO DE TITULACIÓN

Por medio del presente yo Ing. MacArthur Ortega Bustamante, MSc certifico que el Sr. Pozo Chulde Carlos Fabricio portador de la cédula de ciudadanía número 0402092167, ha trabajado en el proyecto de grado “**Desarrollo de una aplicación móvil para el control y gestión de asistencia de los empleados, en el Gad Municipal de Cotacachi utilizando metodologías ágiles y tecnología de desarrollo nativo.**”, previo a la obtención del Título de Ingeniero en Software. Este trabajo se ha realizado con responsabilidad, lo cual certifico con honor a la verdad.

Atentamente

Ing. MacArthur Ortega Bustamante, MSc
DIRECTOR DE TRABAJO DE GRADO

DEDICATORIA

Dedico este trabajo en primer lugar a Dios, porque me dio las fuerzas para seguir adelante en la vida. También se lo dedico a mis padres, Martha y Campo, que siempre han sido mi mayor ejemplo. Gracias por todo lo que sacrificaron por mí, por cada consejo que me dieron con mucho amor y por enseñarme que ser perseverante y honesto es el mejor camino para cumplir los sueños. Cuando las cosas se pusieron muy difíciles, con cansancio y dudas queriendo pararme, fue ver cómo ustedes nunca se rindieron lo que me dio fuerzas para continuar. Por eso, este logro también es gracias a sus enseñanzas, a su esfuerzo de todos los días y a su amor que nunca me faltó.

A mi hermano Eduardo, porque siempre estuvo a mi lado cuando el camino se puso difícil. Gracias por creer y confiar en mí, por darme ánimos en las noches de estudio y estar a mi lado incondicionalmente.

A mi novia Elena le dedico este trabajo por tu paciencia, por entenderme siempre y por apoyarme sin condiciones. Gracias por estar a mi lado en cada paso, por escucharme cuando estaba muy estresado y por creer en mí incluso cuando yo mismo dudaba. Tu amor ha sido mi refugio y la motivación más grande para seguir.

A mis amigos Oliver, Henry, Andy, Theo, Kevin y Edwin los incluyo con mucha gratitud, porque me acompañaron en este camino lleno de retos. Cada momento juntos me enseñó a conocer más etapas de la vida, gracias por brindarme su amistad en los momentos de felicidad y tristeza. Cada uno de ustedes siempre recordare como parte de mi vida.

-Pozo Chulde Carlos Fabricio

AGRADECIMIENTOS

Quiero agradecer de corazón a la UTN por ser el lugar donde crecí como profesional y como persona, desarrollando mis conocimientos y aprendiendo valores tan importantes como la ética, la responsabilidad y el compromiso con la sociedad; de manera muy especial agradezco al Municipio de Cotacachi por abrirme las puertas, confiar en mí y darme el apoyo técnico y el espacio necesario para llevar a cabo este proyecto y poner en práctica todo lo que aprendí en la universidad; también le doy las gracias al MSc. MacArthur Ortega, director de este trabajo, por su guía cercana, sus consejos para hacerlo bien y por exigirme siempre lo mejor, lo que fue clave para que el estudio quedara con calidad y seriedad; y al MSc. Diego Trejo, asesor del proyecto, por sus comentarios útiles, sus aportes técnicos y por apoyarme en todo momento, ayudándome mucho a mejorar y terminar este trabajo de la mejor forma; a todos ellos mi más sincero respeto y agradecimiento, porque sin su ayuda y dedicación no habría podido llegar a esta meta tan importante.

-Pozo Chulde Carlos Fabricio

Índice	
RESUMEN	12
ABSTRACT	13
INTRODUCCIÓN	14
Tema	14
Problema	14
Objetivos	15
Alcance	16
Justificación	17
CAPITULO I	19
Marco Teórico	19
1.1 Tecnologías de desarrollo nativo	19
1.1.1 Plataformas de desarrollo móvil (Android/iOS)	19
1.1.2 Android	20
1.1.3 Kotlin o Java	20
1.1.4 iOS	20
1.1.5 Arquitectura basada en el patrón Modelo Vista Controlador (MVC)	21
1.2 Herramientas y entornos de desarrollo (Android Studio, Xcode)	21
1.2.1 Android Studio	21
1.2.2 Xcode	21
1.2.3 Metodologías Ágiles	22
1.3 Principios y Prácticas de las Metodologías Ágiles	22
1.3.1 Valores del Manifiesto Ágil	22
1.3.2 Principios del Manifiesto Ágil	22
1.3.3 Prácticas comunes	23
1.4 Scrum y Kanban: Características y Comparaciones	24
1.4.1 Kanban	26
1.4.2 Flexibilidad	26
1.4.3 Mejora Continua	27
1.5 Sistemas de Gestión de Asistencia	28
1.5.1 Tecnologías Existentes y sus Limitaciones	28
1.5.2 Importancia de la Automatización en la Gestión de Asistencia	28
1.5.3 Precisión y Confiabilidad:	28
1.5.4 Eficiencia Operativa:	29
1.5.5 Mejora en la Gestión de Recursos Humanos:	29
CAPÍTULO II	31
DESARROLLO	31

2.1.7 Arquitectura del software	42
Lenguajes de programación	44
Frameworks y entorno de ejecución	44
Base de datos	44
Herramientas de desarrollo	44
Herramientas de control de versiones	45
Plataformas y servicios en la nube	45
Herramientas de geolocalización.....	45
Contenedorización	45
2.2.3 Sprint 2	49
Revisión Sprint 2.....	50
2.2.4 Sprint 3 — Empleados + Reportes + Notificaciones (HU3, HU4, HU9) .	53
2.2.5 Sprint 4 — Eventos, puntos móviles y cierre (HU5, HU6).....	56
Revisión Sprint 4.....	56
Retrospectiva Sprint 4.....	56
2.3. Fase 3: Postjuego	58
III. VALIDACIÓN DE RESULTADOS.....	60
3.1. Metodología.....	60
3.1.1. Población y muestra	60
3.1.2 Instrumento de recolección de datos.....	60
3.1.3 Escala de Likert	61
3.2 Análisis de Datos.....	61
3.2.1 Control de asistencia- P1 - Registro sin dificultades	61
Interpretación de resultados.....	61
3.2.2 Control de asistencia -P2- Inicio de sesión complicado.....	62
Interpretación de resultados.....	62
3.2.3 Control de asistencia -P3- Facilidad de uso	63
Interpretación de resultados.....	63
3.2.4 Usabilidad -P4- Tiempo de carga excesivo	64
Interpretación de resultados.....	64
3.2.5 Usabilidad -P5- Botones claramente identificados	65
Interpretación de resultados.....	65
3.2.6 Usabilidad -P6 - Errores frecuentes.....	66
Interpretación de resultados.....	66
3.2.7 Usabilidad -P7- Uso sin ayuda técnica.....	67
Interpretación de resultados.....	67
3.2.8 Eficiencia y rendimiento -P8- Respuesta rápida.....	68

Interpretación de resultados.....	68
3.2.9 Eficiencia y rendimiento -P9- Satisfacción general	69
Interpretación de resultados.....	69
3.2.10 Satisfacción del usuario -P10- Facilita el control de asistencia	70
Interpretación de resultados.....	70
Conclusiones.....	71
Recomendaciones.....	72
Anexos.....	76

Índice de Figuras

Figura 1. Árbol de Problemas.....	15
Figura 2. Sistemas operativos para dispositivos móviles más usados[8]	19
Figura 3. Proceso de Scrum.....	25
Figura 4. Diagrama de caso de Uso - Sistema de Asistencia (Web + Backend).....	37
Figura 5. Diagrama de caso de Uso - App Móvil (Flutter) + Backend.	38
Figura 6. Diagrama de caso de Uso - Módulo Notificaciones.....	38
Figura 7. Diagrama de caso de Uso - Módulo Asistencia.	39
Figura 8. Diagrama de secuencias - Login + Registro de Token FCM.....	39
Figura 9. Diagrama de secuencias - Registro de Asistencia (GPS + Radio).....	40
Figura 10. Diagrama de secuencias - Consulta de Reportes de Asistencia	40
Figura 11. Diagrama de secuencias - Notificación Push en Foreground.	40
Figura 12. Diagrama de secuencias - Notificación Push.....	40
Figura 13. Diagrama de secuencias - Refresh de Token FCM.....	41
Figura 14. Arquitectura del Software.	43
Figura 15. Modelo de datos Lógico.....	43
Figura 16. Login Web.....	47
Figura 17. Login móvil.....	48
Figura 18. Home login empleado.	49
Figura 19. Gestión de Lugares.....	51
Figura 20. Eventos móviles.	52
Figura 21. Gestión de empleados.	54
Figura 22. Notificaciones web.....	55
Figura 23. Identificación de notificaciones móviles.....	55
Figura 24. Gestión de Eventos web.	57
Figura 25. Lista de Eventos web.	57

Índice de Tablas

Tabla 1 Trabajos Relacionados.....	18
Tabla 2. Planificación Sprint 0.	31
Tabla 3. Asignación de roles.	31
Tabla 4. Requerimientos Funcionales.	32
Tabla 5. Requerimientos no Funcionales.	32
Tabla 6.HU1 - Configuración de lugares de trabajo.....	33
Tabla 7. HU2 - Establecimiento de puntos de timbrado.	33
Tabla 8. HU3 - Administración de usuarios.	33
Tabla 9. HU4. Envío de notificaciones.....	34
Tabla 10. HU5 - Configuración de puestos para eventos.....	34
Tabla 11. HU6 - Creación de puntos de encuentro.....	34
Tabla 12. Acceso a la aplicación de asistencia.....	35
Tabla 13. HU8 - Registro diario de asistencia.....	35
Tabla 14. HU9 - Consulta de reportes de asistencia.....	35
Tabla 15. Product Backlog.	36
Tabla 16. Ambiente de desarrollo - Lenguajes de programación.....	44
Tabla 17. Ambiente de desarrollo - Frameworks y entorno de ejecución.....	44
Tabla 18. Ambiente de desarrollo - Base de datos.	44
Tabla 19.Ambiente de desarrollo - Herramientas de desarrollo.....	44
Tabla 20.Ambiente de desarrollo - Herramientas de control de versiones.....	45
Tabla 21. Ambiente de desarrollo - Plataformas y servicios en la nube.	45
Tabla 22. Ambiente de desarrollo - Herramientas de geolocalización.....	45
Tabla 23. Ambiente de desarrollo - Contenedorización.....	45
Tabla 24. Sprint 1 Backlog.....	46
Tabla 25. Revisión Sprint 1.....	47
Tabla 26. Retrospectiva del Sprint 1.	47
Tabla 27. Sprint 2 Backlog.....	50
Tabla 28. Revisión Sprint 2.....	50
Tabla 29. Retrospectiva Sprint 2.	51
Tabla 30.Sprint 3 Backlog.....	53
Tabla 31. Revisión Sprint 3.....	53
Tabla 32. Retrospectiva Sprint 3.	54
Tabla 33. Sprint 4 Backlog.....	56
Tabla 34. Revisión Sprint 4.....	56
Tabla 35. Retrospectiva Sprint 4.....	56
Tabla 36. Pruebas de aceptación - Aplicación móvil de asistencia.....	58
Tabla 37 - Preguntas del Instrumento de recolección de datos	60
Tabla 38 Tabla de frecuencias de respuestas Control de asistencia -P1- Registro sin dificultades	61
Tabla 39 Tabla de frecuencia de respuestas -Control de asistencia -P2- Inivio de sesion complicado	62
Tabla 40 Tabla de frecuencia de respuestas- Control de asistencia -P3- Facilidad de uso	63
Tabla 41 Tabla de frecuencia de resouestas-Usabilidad -P4-Tiempo de carga excesivo	64
Tabla 42 Tabla de frecuencia de respuestas - Usabilidad -P5-Botones claramente identificados	65
Tabla 43 Tabla de frecuencia de respuestas- Usabilidad -P6-Errores frecuentes.....	66
Tabla 44 Tabla de Frecuencia de respuestas-P7- Uso sin ayuda técnica.....	67
Tabla 45 Tabla de frecuencia de respuestas -P8-Respuesta rápida	68

Tabla 46 Tabla de frecuencia de respuestas -Eficiencia y rendimiento -P9- Satisfacción general	69
Tabla 47 Tabla de frecuencia de respuestas-Satisfaccion del usuario-P10 -Facilita el control de calidad.....	70

RESUMEN

Este trabajo nace de una necesidad muy concreta: mejorar la forma en que los empleados del GAD Municipal de Cotacachi registran su asistencia, especialmente aquellos que trabajan fuera de las oficinas. Durante mucho tiempo, el control dependía de métodos tradicionales que no siempre eran prácticos ni eficientes. Por eso, se propuso desarrollar una aplicación móvil que facilitara este proceso y lo hiciera más seguro y confiable.

La solución fue un proceso en sí mismo, utilizando metodologías ágiles y tecnologías actuales. Así fue como se desarrolló una aplicación móvil en Flutter que a su vez estaba conectada a un sistema central (backend) creado en Node.js y que tenía una base de datos PostgreSQL, integrado además la geolocalización para validar el registro de asistencia desde el lugar autorizado y notificaciones para mantener informados a los empleados. En definitiva, todo esto funciona como un engranaje bien engrasado, el usuario marca asistencia desde el celular y el sistema valida, registra y almacena la información instantáneamente.

Más allá de la parte técnica, la cuestión era saber si realmente funcionaba para quién la iba a usar. Para ello, se realizó una encuesta a 55 empleados en función de los criterios de calidad del estándar ISO/IEC 25010. Los resultados estaban muy claros: la mayoría de los encuestados considera que la aplicación es fácil de usar, responde rápido y es capaz de cumplir el propósito propuesto.

De hecho, todos los participantes indicaron que este software facilita el control diario de asistencia. Y esto, en la práctica, significa que esta herramienta representa una ventaja. En definitiva, la aplicación no solamente es una herramienta que digitaliza un proceso, sino que representa un pequeño avance en la gestión por parte del municipio, más ordenada, más transparente y más moderna. Se trata de una solución concreta a un problema real y práctica para aquellas personas que la utilizan día a día.

Palabras Clave: Control de asistencia, Aplicación móvil, geolocalización, Desarrollo Móvil con Flutter, Gestión de empleados, Sistemas de información.

ABSTRACT

This research project arose from a simple yet important need: the desire to improve how Cotacachi Municipal Government employees record their attendance, especially those working far from the main offices. Traditional methods were no longer practical or efficient, so the goal was to create a mobile application that would make this process easier, safer, and more reliable.

The solution was created utilizing agile methodologies and state-of-the-art technologies. A mobile application was developed through Flutter, while the back end was done in Node.js and had the assistance of a PostgreSQL database. Geolocation was introduced to achieve that attendance can be recorded only at the locations that are authorized, and push notifications were incorporated to make sure notified employees. Everything runs in real time: the employees clock in from their mobile phones, and the information is validated and saved at the moment.

To know if the system met user expectations, a survey based on the ISO/IEC 25010 quality standard was launched with 55 employees. The results were very good since most of the participants stated that the application was easy to use, fast and useful to monitor attendance in their daily work. Additionally, all the participants stated that the application helped to track attendance. It can therefore be stated that the tool is not only usable but also very practical when it comes to everyday use.

In conclusion, the developed application is more than just a digital system. It is a step towards a more organized, transparent, and modern administrative process in the municipality: a real solution made for the people.

Keywords: Attendance control, Mobile application, Geolocation, Mobile development with Flutter, Employee management, Information systems.

INTRODUCCIÓN

Tema

DESARROLLO DE UNA APLICACIÓN MÓVIL PARA EL CONTROL Y GESTIÓN DE ASISTENCIA DE LOS EMPLEADOS, EN EL GAD MUNICIPAL DE COTACACHI UTILIZANDO METODOLOGÍAS ÁGILES Y TECNOLOGÍA DE DESARROLLO NATIVO

Problema

El desarrollo de una aplicación móvil para el control y gestión de asistencia de los empleados en el GAD municipal de Cotacachi. La iniciativa de este proyecto es que los empleados que trabajan fuera de la municipalidad necesitan una forma eficiente de registrar sus actividades en lugares remotos, lo cual es esencial para mantener la confianza y la cooperación con la ciudadanía y las empresas locales. Además, la extensa cobertura geográfica de Cotacachi, que incluye oficinas satélites en localidades como Apuela, Cuellaje, Magdalena y Golondrinas, complica la supervisión y el seguimiento del personal. Sin una herramienta tecnológica adecuada, es difícil asegurar que las tareas se realizan de manera oportuna y eficiente en los horarios establecidos. La propuesta de una aplicación móvil con funcionalidades de geolocalización, identificación digital, y registro con cedula de identidad que solo facilitaría la gestión de la asistencia y las actividades de campo, sino que también permitiría una supervisión más efectiva y una mayor transparencia en las operaciones municipales. Esta solución tecnológica fortalecería la capacidad del GAD municipal de Cotacachi para gestionar su personal de manera más eficiente, mejorando la calidad del servicio prestado a la comunidad.

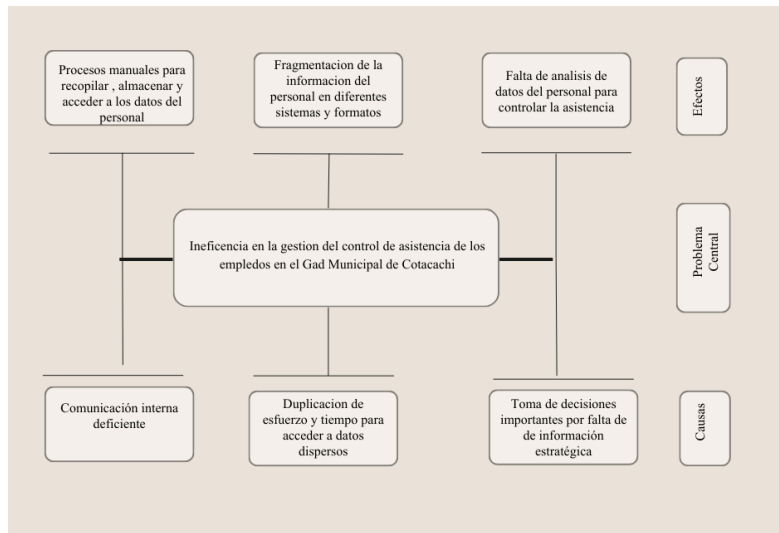


Figura 1. Árbol de Problemas.

Objetivos

Objetivo General

Desarrollar una aplicación móvil para el control y gestión eficiente de la asistencia de los empleados en el GAD municipal de Cotacachi, empleando metodologías ágiles y tecnologías de desarrollo nativo

Objetivos Específicos:

- Realizar un marco teórico sobre herramientas tecnológicas y metodologías de implementación de aplicaciones móviles.
- Desarrollar una aplicación móvil para el GAD Municipal de Cotacachi, utilizando el marco Scrum.
- Evaluar la usabilidad de la aplicación móvil basándose en los estándares de la norma ISO/IEC 25010

Alcance

El objetivo principal de desarrollar una aplicación móvil para la Administración Municipal de Cotacachi es crear una herramienta digital que registre y facilite la gestión de información del personal. El proyecto se centrará en el desarrollo de una aplicación móvil para identificar toda la información de los empleados, al momento de realizar una actividad, implementándola en el módulo de gestión en el municipio de Cotacachi, todo esto con el fin de incrementar la eficiencia y mejorar la toma de decisiones dentro de la administración pública. La aplicación incluirá módulos importantes como por ejemplo: Gestión de Personal (que llevará el registro, control, cumplimiento de actividad de los empleados); y funcionalidades para el control o cumplimiento de asistencia de los empleados; módulos para: Comunicaciones Internas (para el envío de notificaciones y mensajes) y para Seguridad y Acceso (para la gestión de roles). Son módulos pensados para aumentar la eficacia y la calidad de la toma de decisiones en la administración pública. El empleo de frameworks modernos para el desarrollo de aplicaciones móviles trae consigo varias ventajas, como la mejora del rendimiento y la capacidad de funcionar en distintas plataformas. Estos frameworks son beneficiosos porque soportan la compilación en diversos formatos de código, asegurando un funcionamiento suave y uniforme en múltiples dispositivos.[1]

Metodología

Para el primer objetivo de la investigación, se realizó una revisión técnica y documental relacionada con el desarrollo de aplicaciones móviles multiplataforma, geolocalización en tiempo real y sistemas de gestión de información institucional. Se han analizado artículos académicos, documentación oficial y casos de estudio, consultando bases de datos dedicadas, por medio de términos de control de asistencia, validación GPS y arquitectura de software. Con el fin de identificar las tecnologías más apropiadas para la implementación de un sistema que optimizase la administración del personal en el GAD Municipal de Cotacachi

Fue necesario aplicar este proceso; por lo que para cumplir con el segundo objetivo, se estima conveniente aplicar la metodología ágil conocida como Scrum, el cual organiza el desarrollo del sistema en iteraciones conocidas como sprints que permiten implementar poco a poco las funcionalidades del sistema. Se adoptó una arquitectura estructurada compuesta por una aplicación móvil, un backend centralizado y una base de datos relacional, garantizando escalabilidad y mantenimiento del software. Durante el

desarrollo se implementaron módulos como autenticación de usuarios, gestión de empleados, gestión de horarios, gestión de roles y cargos, gestión de eventos, gestión de lugares con coordenadas geográficas, registro de asistencia con validación por GPS, envío de notificaciones y generación de reportes.

Para llevar a cabo el tercer objetivo se tiene que evaluar la calidad del propio sistema conforme a la subcaracterística de la usabilidad considerada en la norma ISO/IEC 25010, se establecieron criterios orientados a medir la facilidad de uso, la eficiencia, la comprensión y la satisfacción del usuario final. En este sentido, se ha podido verificar que la aplicación desarrollada cumple con los niveles de calidad y responde a las necesidades de gestión y control del personal en el GAD Municipal de Cotacachi.

Justificación

El desarrollo de esta aplicación móvil responde a la necesidad de mejorar los procesos de control y gestión del personal en el GAD Municipal de Cotacachi. La digitalización del registro de asistencia permite contar con información organizada y confiable, reduciendo errores derivados de métodos manuales. Esta propuesta se relaciona con el Objetivo de Desarrollo Sostenible 16, que promueve instituciones más eficientes y transparentes, ya que fortalece la gestión interna y facilita el seguimiento de las actividades del personal.

Justificación Tecnológica

El sistema integra una aplicación móvil desarrollada en Flutter con un backend en Node.js y una base de datos PostgreSQL, lo que permite centralizar la información y mejorar su administración. La validación de asistencia mediante geolocalización y el envío de notificaciones en tiempo real optimizan el control institucional y la comunicación interna. La utilización de tecnologías actuales aporta estabilidad, escalabilidad y mayor seguridad en el manejo de datos.

Justificación Ambiental

La implementación del sistema contribuye a disminuir el uso de documentos físicos en los procesos administrativos, reduciendo el consumo de papel y otros recursos. Al automatizar el registro de asistencia y los reportes, se promueve una gestión más sostenible y alineada con prácticas responsables dentro del sector público.

Contexto de la investigación

Tabla 1 Trabajos Relacionados

Investigación	Aporte
Contexto: Desarrollo local de aplicación móvil con Flutter para promover el turismo del GAD de Pedro Moncayo [2]	Para la información se creó una página web para que el departamento de turismo gestione la información turística.
Contexto: Implementación local de una app móvil para mejorar la gestión de notificaciones e incidencias ciudadanas del GAD municipal de San Miguel de Ibarra, basado en las características de adecuación funcional de la ISO/IEC 25010.[3]	El portal ciudadano ofrece varios servicios en línea que facilitarán ciertos trámites y su estado actual, pero no es el medio digital idóneo para abordar las incidencias.
Contexto: Nacional Desarrollo de aplicación móvil para empresa de bienes raíces, versión android [4]	Ayuda a captar nuevos clientes para la empresa. Se utilizaron herramientas de código abierto para desarrollarla, servidor linux centos para alojar los archivos de datos, php, mysql server y json. Además cuenta con un sistema web que permite usar información. Opera en teléfonos con sistema operativo android 4.0 o superior
Contexto: Nacional Creación de una app móvil con gestión en línea para la generación de Gobierno en GAD'S parroquiales[5]	Reducir la brecha existente entre la ciudadanía y gobierno e incentivar la participación de los ciudadanos, apoyado en los principios del gobierno abierto y utilizando las tecnologías de información, para este caso el estudio se realizó en conjunto con el GAD parroquial de Conocoto
Contexto: Internacional Diseño y Desarrollo de una Aplicación de Gestión del Tiempo[6]	El objetivo del trabajo consiste en desarrollar una aplicación móvil con la finalidad de ayudar en el rendimiento la organización del día

CAPITULO I

Marco Teórico

1.1 Tecnologías de desarrollo nativo

1.1.1 Plataformas de desarrollo móvil (Android/iOS)

En la actualidad, el desarrollo de aplicaciones móviles se encuentra frente a desafíos debido a las dificultades técnicas y limitaciones propias del entorno en el que operan. Algunos de los desafíos incluyen la variabilidad en las capacidades de los dispositivos, múltiples estándares de red y tecnologías, compatibilidad multiplataforma necesaria, demandas específicas por parte de los usuarios y la constante presión para cumplir con plazos ajustados en el mercado [7]

Como vemos en la figura 1, los sistemas operativos que son líderes en el mercado actualmente son Android y iOS, según StatCounter Global que es un servicio de análisis web en tiempo real, Android es el sistema operativo líder con un 70.4 % seguido por iOS con un 28.7%. Por lo tanto, debido a la cuota de mercado es importante al empezar un proyecto de desarrollo móvil considerar estos sistemas operativos.

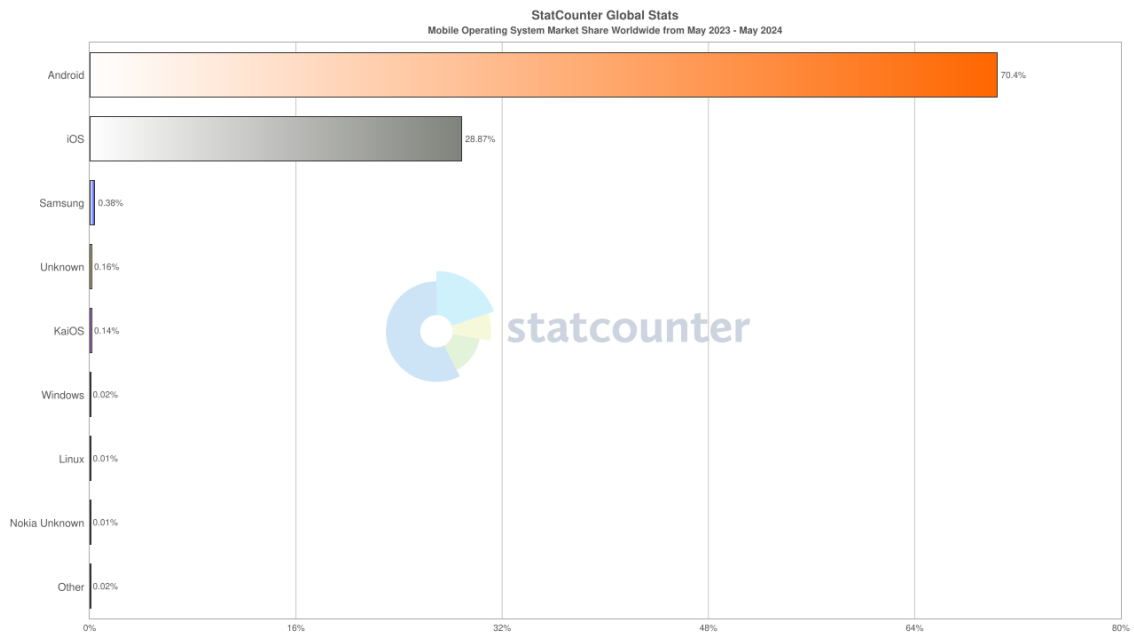


Figura 2. Sistemas operativos para dispositivos móviles más usados[8]

1.1.2 Android

Android fue diseñado desde su creación para ser instalado y utilizado en dispositivos móviles que cuentan con pantallas táctiles, también denominadas interfaces de tipo touch. Mediante el contacto directo con la pantalla es posible interactuar con el dispositivo utilizando esta tecnología. En primer lugar, la empresa Android Inc. fue responsable del desarrollo inicial del sistema operativo y recibió respaldo financiero de parte de Google para llevar a cabo el proyecto[9].

1.1.3 Kotlin o Java

Podemos utilizar lenguajes de programación como Kotlin, Java y C# para desarrollar aplicaciones en Android. A través del uso de herramientas como el toolkit, se compila el código, los datos y los archivos para crear un APK (Android package). Mediante el uso de estos archivos, es posible instalar la aplicación en el sistema[10]

Gracias a su innovadora propuesta "escribe una vez y ejecuta en cualquier lugar", Java logró ingresar al mercado de manera exitosa, logrando expandir su presencia no solo en computadoras personales y teléfonos inteligentes, sino también abriendo puertas hacia navegadores web, televisores, electrodomésticos y muchos otros dispositivos. A diferencia de Java, que fue desarrollado mucho antes, Kotlin aprovecha la experiencia acumulada durante esos años para optimizar su diseño desde el principio[11].

1.1.4 iOS

El sistema operativo iOS, que antes se conocía como iPhone OS, fue lanzado en 2007. iOS 17 trae mejoras significativas en materia de privacidad, inteligencia artificial y la posibilidad de integrarse con otros dispositivos Apple. El sistema operativo utiliza un kernel XNU (X is Not Unix), el cual está compuesto por partes de Mach y FreeBSD. El sistema iOS ofrece una experiencia fluida y segura al centrarse en la perfecta integración de hardware y software para optimizar el rendimiento y asegurar el dispositivo[11].

Xcode, la IDE oficial de Apple, provee el entorno de desarrollo para iOS e incluye un editor de código, simulador de dispositivos iOS, depurador y diseñador gráfico. Swift, el preferido lenguaje de programación, se destaca por su rendimiento y seguridad. Asimismo, se integra eficientemente con las herramientas de desarrollo de Apple para brindar un entorno seguro y eficaz [12]

1.1.5 Arquitectura basada en el patrón Modelo Vista Controlador (MVC)

La estructura del sistema se organizó bajo un enfoque de separación de responsabilidades similar al patrón Modelo Vista Controlador (MVC), lo que permite dividir claramente las funciones del software. En este esquema, el modelo está representado por la base de datos y las entidades que almacenan la información, la vista corresponde a la aplicación móvil que interactúa directamente con el usuario, y el controlador se implementa en el backend, donde se procesan las peticiones y se aplican las reglas del negocio. Esta organización facilita el mantenimiento del sistema, mejora su escalabilidad y permite realizar actualizaciones sin afectar el funcionamiento general de la aplicación.[13]

1.2 Herramientas y entornos de desarrollo (Android Studio, Xcode)

1.2.1 Android Studio

El editor de código inteligente con autocompletado, el soporte para diversas versiones de Android SDK y las herramientas de rendimiento y profiling están incorporados en Android Studio.[14] Además, este entorno de desarrollo cuenta con un sistema de emulación resistente que posibilita la prueba de aplicaciones en diferentes dispositivos y configuraciones. Esto garantiza una cobertura amplia en las pruebas y asegura su compatibilidad.

Android Studio se integra perfectamente con plataformas como Firebase, GitHub y Google Cloud, mejorando la colaboración y el despliegue continuo de aplicaciones. Además, soporta la generación de APKs y AABs (Android App Bundles) optimizados para distintas configuraciones de dispositivos, facilitando un proceso de desarrollo más eficiente y personalizado[14].

1.2.2 Xcode

El editor de código en Xcode cuenta con soporte para autocompletado, refactorización del código y la opción de depurar el programa en tiempo real. El entorno de desarrollo tiene un simulador de dispositivos iOS que es compatible con diversas versiones y configuraciones, lo cual brinda una plataforma sólida para desarrollar y probar aplicaciones [8].

Xcode, además de permitir la integración de frameworks clave como UIKit, SwiftUI y CoreML, simplifica el proceso de desarrollar interfaces de usuario interactivas y agregar características avanzadas como IA y AR. Con estas herramientas, los

desarrolladores logran optimizar las capacidades de los dispositivos iOS y crear experiencias personalizadas y sofisticadas para los usuarios[15] .

1.2.3 Metodologías Ágiles

Las metodologías ágiles se han establecido firmemente como un enfoque eficaz para la gestión de proyectos de desarrollo de software. En este contexto, se explorarán en detalle sus principios fundamentales, las prácticas más comunes y se realizará una comparación entre dos de sus marcos de trabajo más utilizados: Scrum y Kanban.[16]

1.3 Principios y Prácticas de las Metodologías Ágiles

Los métodos ágiles han demostrado ser una forma efectiva de gestionar proyectos de desarrollo de software. En este contexto, se explorarán en detalle sus principios fundamentales, las prácticas más comunes y se realizará una comparación entre dos de sus marcos de trabajo más utilizados: Scrum y también Kanban [17]:

1.3.1 Valores del Manifiesto Ágil

- Individuos e Interacciones sobre Procesos y Herramientas: Priorizamos las relaciones humanas por encima de los procedimientos y herramientas formales, para fomentar la comunicación y colaboración directa entre los miembros del equipo[18] .
- Software Funcionando sobre Documentación Extensa: El enfoque principal es entregar software funcional y utilizable, sin enfocarse demasiado en producir documentación superflua [19] .
- Colaboración con el Cliente sobre Negociación de Contratos: La colaboración continua con el cliente es fundamental para garantizar que el producto final satisfaga sus expectativas en constante cambio y necesidades[20] .
- Responder al Cambio sobre Seguir un Plan: Se toma en cuenta la habilidad para adaptarse y ser flexible ante los cambios en los requerimientos del proyecto, en vez de seguir rigurosamente un plan establecido.[20] .

1.3.2 Principios del Manifiesto Ágil

- Cumplir con el cliente mediante una entrega constante y anticipada de software: Al enfocarse en la entrega temprana y continua de software, los clientes podrán ver pronto los resultados de su inversión y ofrecer retroalimentación instantánea. Al aplicar esta metodología, se evita la creación de funcionalidades innecesarias o sin utilidad [21].

- Dar preferencia a los sistemas operativos sobre la documentación: La funcionalidad operativa del software es lo que realmente proporciona beneficios al cliente. A pesar de que la documentación es importante, no debe retrasar el cumplimiento del cronograma para desarrollar funcionalidades útiles[22].
- Colaborar con el cliente a lo largo del proyecto: Al mantener una colaboración constante con el cliente, se asegura que el desarrollo esté alineado con las cambiantes necesidades y expectativas del mismo, lo cual reduce la probabilidad de malentendidos y trabajo extra[23] .
- Responder al cambio más que seguir un plan: Los entornos de desarrollo de software son sumamente variables y es posible que las necesidades del cliente cambien rápidamente. Es esencial contar con la capacidad de adaptarse y responder a estos cambios para lograr el éxito del proyecto[24] .
- Mejorar constantemente la eficiencia del equipo :El equipo debe mantenerse constantemente en busca de formas de mejorar su trabajo y eliminar ineficiencias para lograr una mejora continua, lo cual se traduce en un mejor rendimiento y mayor satisfacción del cliente[25] .

1.3.3 Prácticas comunes

- Planificación Iterativa: Los sprints, también llamados ciclos cortos de desarrollo, posibilitan la realización rápida y frecuente de ajustes. Dividir el proyecto en segmentos manejables ayuda a adaptarse a los cambios y mejorar la capacidad de entrega, según este enfoque. En cada sprint, que generalmente tiene una duración de entre una y cuatro semanas, se ofrece la oportunidad de evaluar el progreso, incorporar comentarios y ajustar las prioridades sin alterar el cronograma global.[26]
- Daily Stand-ups: Es esencial asegurar que el equipo esté sincronizado a través de las reuniones diarias, también conocidas como daily stand-ups. Los miembros del equipo comparten sus progresos, planes y obstáculos en estas reuniones breves que suelen durar 15 minutos. Evitar retrasos y garantizar que todos estén al tanto del progreso del proyecto se logra al identificar y resolver los impedimentos de manera rápida.[27] .
- Retrospectivas: El equipo puede reflexionar sobre lo que ha funcionado bien y lo que necesita mejorar gracias a las reuniones periódicas de retrospcción. Estas

sesiones favorecen la creación de un ambiente en donde cada individuo pueda evaluar su evolución y a su vez, permitir que el equipo, contribuya al aprendizaje de la retrospectiva y tenga la posibilidad de compartir tanto lo positivo logrado como sus obstáculos / problemas. En el caso del equipo que ha encontrado áreas para mejorar y sugiere mejoras, puede hacer mejoras en los siguientes sprints para hacerlo más eficiente y para poder ser más efectivo [28] .

- Revisión de Sprint: Tiene importancia hacer cada revisión al final del sprint para enseñar al cliente lo que se ha realizado y escuchar su opinión. De este modo el desarrollo sabe que va cumpliendo sus expectativas porque ya pueden realizar pequeños ajustes o correcciones en el mismo. Las revisiones de sprint también garantizan que los avances del proyecto son transparentes, y eso genera confianza entre el equipo y el cliente [29]

1.4 Scrum y Kanban: Características y Comparaciones

Scrum es una técnica rápida empleada en la creación de software para gestionar eficientemente proyectos complejos. Esta metodología se fundamenta en los principios de transparencia, inspección y adaptación, fomentando así la colaboración, autoorganización y entrega constante de valor al cliente. En Scrum, los proyectos se fragmentan en períodos de tiempo llamados "sprints", que suelen ser de dos a cuatro semanas. Durante estos sprints, se trabaja en el desarrollo incremental del producto para hacerlo funcional y potencialmente entregable.[30]

- Product Owner: El papel de (PO) es crucial para representar los intereses de los stakeholders en un proyecto. El desempeño de este rol consiste en establecer y dar importancia a las cualidades del producto que se desarrollará, asegurando que el equipo de desarrollo comprenda sin ambigüedades estos requerimientos y maximizando así el valor del producto. El Product Owner trabaja de manera cercana con el equipo Scrum, los interesados y el Scrum Master para gestionar efectivamente el backlog del producto y garantizar un progreso eficaz del proyecto[31].
- Scrum Master: El Scrum Master juega un rol fundamental como facilitador y entrenador del equipo, garantizando el cumplimiento de las prácticas ágiles de Scrum y eliminando cualquier obstáculo que pueda dificultar el avance del equipo. Además, ayuda al equipo en su constante mejora al facilitar las reuniones y eventos de Scrum [17] .

- **Equipo de Desarrollo:** Los profesionales del equipo de desarrollo trabajan juntos para crear el incremento del producto. Ellos poseen autonomía y capacidad para realizar múltiples funciones, encargándose de planificar, ejecutar y entregar el trabajo en cada uno de los sprints. El éxito del proyecto es compartido por todo el equipo, ya que colaboran estrechamente[32] .

Scrum utiliza artefactos como el Product Backlog, el Sprint Backlog y el aumento de Producto para gestionar y seguir el progreso del trabajo.

- **Product Backlog:** Se trata de una lista que presenta en orden de importancia todas las tareas necesarias para llevar a cabo el proyecto. Es responsabilidad del Product Owner gestionar y priorizar este backlog, garantizando que esté siempre en línea con las necesidades empresariales y las expectativas de los clientes[33] .
- **Sprint Backlog:** Durante el sprint, el equipo de desarrollo se compromete a completar un subconjunto del Product Backlog. Asegúrate de incorporar los elementos elegidos para el sprint actual junto con un plan detallado que describa cómo el equipo alcanzará los objetivos del mismo [33].
- **Incremento de Producto:** Después de cada sprint, se espera que el equipo genere un incremento del producto que sea potencialmente entregable. Este aumento es una versión del producto que ha sido probada y verificada, siendo funcional y lista para ser lanzada si se decide hacerlo[33].

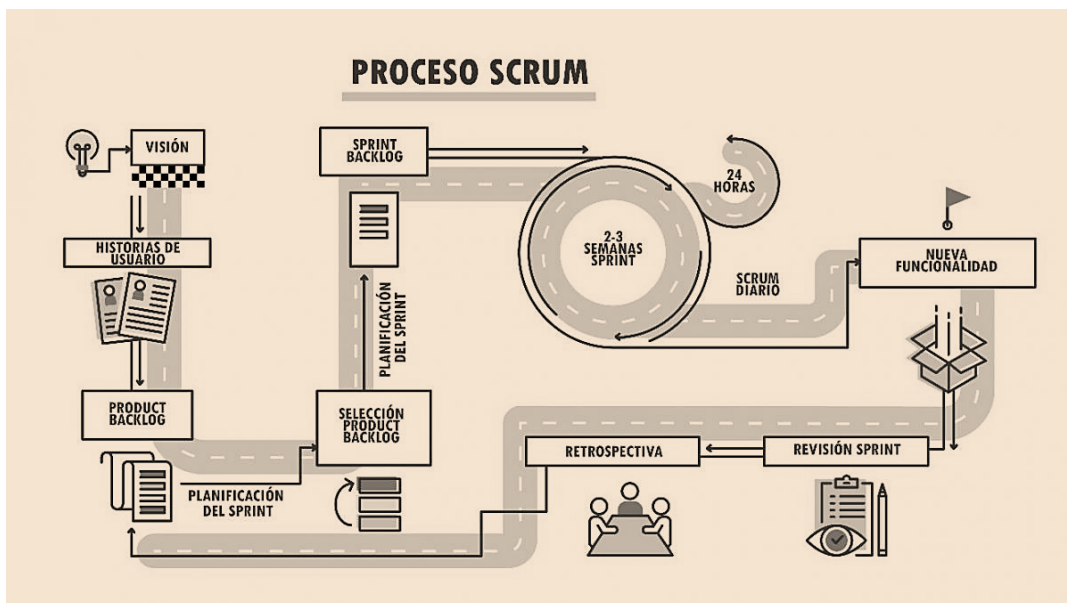


Figura 3. Proceso de Scrum.

1.4.1 Kanban

Se trata de una metodología visual que se enfoca en mejorar la eficiencia del trabajo y reducir el trabajo sin terminar (WIP). La metodología utiliza tableros visuales que muestran las tareas a realizar, separadas en columnas según su estado actual. Kanban tiene como propósito principal maximizar la eficiencia, detectar los puntos de congestión y promover una mejora constante en los procedimientos laborales[34].

Flujo de Trabajo Visual

- **Tableros Kanban:** Son las herramientas gráficas que se emplean para administrar y observar el desarrollo de tareas en un proyecto. "Kanban", un término que se origina en Japón, combina las palabras "Kan" y "Ban". La primera significa letrero, mientras que la segunda se refiere a tablero. Estos tableros proporcionan una representación visual y clara del avance de las tareas y metas a medida que se va llevando a cabo el trabajo planificado[35].
- **Visualización del trabajo:** El uso de un tablero Kanban posibilita que todo el equipo pueda comprender al instante y mantenerse actualizado sobre el estado del proyecto, lo cual favorece la comunicación, colaboración y toma de decisiones fundamentadas. Promoviendo la transparencia, organización y eficiencia en la gestión del proyecto se logra al poder visualizar de manera clara las tareas realizadas, quién las lleva a cabo y el estado en que se encuentran[35].
- **Limitación del trabajo en curso (WIP):** Establecer un límite máximo en la cantidad de tareas o elementos que un equipo puede tener en progreso simultáneamente es fundamental dentro de la metodología Kanban. Uno de los objetivos de esta restricción es evitar la sobrecarga laboral, disminuir la realización de múltiples tareas al mismo tiempo y optimizar el flujo eficiente del trabajo[35].

1.4.2 Flexibilidad

- La capacidad de modificar y reorganizar las tareas, asignar recursos de manera dinámica, cambiar las prioridades según sea necesario y responder rápidamente a nuevas demandas o situaciones imprevistas es indicativo de flexibilidad. El uso de tableros Kanban ofrece una manera visual y transparente de administrar el trabajo, lo que facilita la detección de áreas para mejorar, tomar decisiones informadas y adaptarse constantemente a medida que se progresa en el proyecto[35].
- **Sin Iteraciones Fijas:** En Kanban, no tener iteraciones fijas ayuda a entregar valor de forma continua, optimizar el flujo de trabajo y priorizar las tareas según su

importancia y urgencia en tiempo real. La flexibilidad de Kanban permite una gestión del trabajo más dinámica y adaptable, lo que fomenta la eficiencia, transparencia y colaboración en el equipo de desarrollo [35].

- Implementación adaptativa: Los resultados se pueden obtener mediante la visualización del trabajo, limitando el trabajo en proceso, midiendo y analizando las métricas de desempeño y fomentando una colaboración activa entre los miembros del equipo. Con una representación visual clara del estado de las tareas en el tablero Kanban, es posible que el equipo identifique rápidamente áreas de mejora, cuellos de botella o desviaciones en el proceso y tome medidas correctivas oportunas[35].

1.4.3 Mejora Continua

- Ciclo de Mejora Continua (Kaizen): Se trata de un proceso recurrente que se centra en la constante y gradual mejora de los procesos, productos y servicios de una empresa. Para lograr mejoras sostenidas en el tiempo, este ciclo se compone de cuatro etapas principales que se repiten de manera continua[36].
- Ajuste de Límites WIP: Los límites de trabajo en curso son flexibles y se adaptan según sea necesario para mejorar el flujo de trabajo. Mediante la exploración de distintos límites y la observación de los resultados, los equipos pueden descubrir el punto ideal que permita maximizar la productividad y minimizar las restricciones[36].
- Reuniones de Revisión: Son un componente fundamental en el proceso de mejora continua de una organización, ya que permiten evaluar el progreso, detectar oportunidades de mejora y realizar decisiones estratégicas fundamentadas en datos y resultados concretos. Normalmente, estas reuniones se hacen periódicamente y pueden tener la participación de personas de diferentes niveles en la organización [36].
- Indicadores de Rendimiento: Dado que existen indicadores para facilitar la mejora continua de una organización, ya que permiten medir progresos, detectar oportunidades de mejora y tomar decisiones estratégicas y fundamentadas por datos y resultados tangibles. Normalmente, estas reuniones se realizan periódicamente, con personas de diferentes niveles en la propia compañía en la reunión [36].

1.5 Sistemas de Gestión de Asistencia

Con el avance de la tecnología digital, los sistemas de gestión de asistencia han experimentado grandes cambios, sustituyendo los métodos manuales tradicionales por soluciones automatizadas y en la nube. Además de simplificar el proceso de registro de asistencia, estos sistemas modernos también cuentan con características adicionales como análisis de datos, integración de comunicaciones y mejoras en la seguridad y eficiencia operacional[37].

1.5.1 Tecnologías Existentes y sus Limitaciones

Dentro de las limitaciones de las tecnologías existentes en los sistemas de gestión de asistencia podemos destacar las siguientes:

- Complejidad de integración: La implementación y mantenimiento de ciertas tecnologías pueden ser complicados para usuarios menos experimentados debido a la pronunciada curva de aprendizaje que requieren[37].
- Costos: La incorporación de ciertas tecnologías avanzadas puede suponer gastos considerables, sobre todo para empresas o instituciones educativas con recursos económicos limitados[37].
- Seguridad: Aunque se han tomado medidas de seguridad en las tecnologías actuales, siempre existe la posibilidad de que haya vulnerabilidades y fallos de seguridad que pongan en riesgo la integridad de los datos de asistencia[37].
- Escalabilidad: Determinadas tecnologías pueden presentar dificultades en cuanto a su capacidad de escalabilidad, sobre todo cuando se trata del manejo de grandes cantidades de datos relacionados con asistencia en entornos que cuentan con varios usuarios [37].

1.5.2 Importancia de la Automatización en la Gestión de Asistencia

La automatización de la administración de asistencia no solo facilita los trámites administrativos, sino que también mejora la precisión de los registros, promueve la sostenibilidad ambiental y favorece la toma de decisiones informadas en contextos educativos y laborales. [38]

1.5.3 Precisión y Confiabilidad:

- Hace referencia a la habilidad de un sistema, instrumento o método para ofrecer resultados que sean consistentes y se acerquen al valor real o verdadero de una medida. Dentro del marco de controlar la asistencia escolar mediante tecnología RFID y GSM, la precisión está vinculada a registrar correctamente la presencia

de los estudiantes. Para evaluar la precisión del sistema, se contrastan los registros obtenidos con el sistema propuesto y lo que realmente está sucediendo en términos de asistencia estudiantil. Por lo general, la exactitud se representa en forma de porcentaje para mostrar qué tan cercanos están los registros del sistema a la realidad[39].

1.5.4 Eficiencia Operativa:

- La eficiencia operativa es un aspecto fundamental dentro los sistemas digitalizados, ya que puede marcar la diferencia en la rentabilidad y competitividad de las empresas [40].
- Procesos Simplificados: Es posible llevar a cabo diferentes acciones, tales como la automatización de tareas repetitivas, la integración de sistemas para evitar duplicar esfuerzos, digitalizar documentos para facilitar su acceso y gestión, y estandarizar procedimientos para asegurar consistencia en las operaciones[41].

1.5.5 Mejora en la Gestión de Recursos Humanos:

- La digitalización ha tenido un impacto positivo en la administración del talento humano, ya que las empresas pueden mejorar sus operaciones y fomentar el crecimiento de su equipo[42].
- El uso de sistemas de gestión de recursos humanos (HRMS) y herramientas digitales facilita la automatización de procesos, lo cual agiliza la gestión administrativa y deja más tiempo libre para actividades estratégicas [42].
- Toma de decisiones basada en datos: La analítica de recursos humanos provee valiosa información acerca del rendimiento, la retención de talento y el grado de satisfacción de los empleados. Esto permite que las empresas tomen decisiones estratégicas e informadas [42].
- Mejora en la experiencia del empleado: El uso de herramientas digitales como programas de capacitación en línea, plataformas de retroalimentación continua y análisis de participación, ayuda a mejorar la satisfacción y el compromiso del personal al personalizar su experiencia laboral[42].
- Colaboración y comunicación efectiva: El uso de aplicaciones y herramientas digitales para la comunicación interna y colaboración en línea simplifica la interacción entre los equipos, lo cual resulta en una mejora de la comunicación interna y promueve un ambiente laboral favorable al trabajo conjunto[42].

- Toma de Decisiones Basada en Datos: Ofrece información analítica que se pueda emplear para tomar decisiones informadas acerca de la administración del personal, la programación de horarios y el uso eficiente de los recursos[42].

CAPÍTULO II

DESARROLLO

2.1. Fase 1: Prejuego

2.1.1. Sprint 0

El Sprint 0 corresponde a la fase inicial de planificación en la que se establecen las bases del proyecto, se definen los objetivos, el alcance y la arquitectura tecnológica del sistema. Aunque no es un evento formal dentro de Scrum, en la práctica se utiliza para preparar el entorno de desarrollo y organizar el Product Backlog.[33]

Tabla 2. Planificación Sprint 0.

Elemento	Descripción
Asistentes	Product Owner (Estudiante), Scrum Master (Tutor), Equipo de desarrollo (Estudiante desarrollador).
Objetivo	Establecer las bases técnicas, funcionales y organizativas necesarias para el desarrollo del sistema de control y gestión de asistencia.

Fase de desarrollo	Tareas realizadas
Planificación	- Definición de roles del equipo Scrum.- Levantamiento de requerimientos funcionales y no funcionales.- Redacción de historias de usuario.- Elaboración y priorización del Product Backlog.
Diseño	- Diseño inicial de la base de datos en PostgreSQL.- Definición de la arquitectura del sistema (Frontend Flutter – Backend Node.js).- Elaboración de diagramas UML (casos de uso, clases y arquitectura).

Estructura del Equipo de Trabajo Scrum

Dado que este es un proyecto individual de titulación, se adapta la estructura de Scrum de la siguiente manera:

2.1.2. Asignación de roles

Tabla 3. Asignación de roles.

Rol	Adaptación en el Proyecto
Product Owner	Estudiante (Carlos Pozo) con supervisión del tutor académico.
Scrum Master	Tutor (MSc. MacArthur Ortega), orientador del proceso.
Equipo Scrum	Estudiante desarrollador, responsable de la implementación.

2.1.3 Definición de requisitos

Historias de usuario

Se aplicó la técnica ágil de historias de usuario (HU) para documentar necesidades y procesos desde la visión del usuario final. Seguidamente, en las tablas se establecen las condiciones para la creación de la aplicación.

Requerimientos Funcionales

Tabla 4. Requerimientos Funcionales.

Requerimiento	Descripción
Iniciar sesión	El administrador debe ingresar al sistema con el usuario y contraseña correspondiente.
Gestión de lugares de trabajo	El administrador gestionará la creación, modificación y eliminación de los lugares de trabajo, registrando datos como nombre, dirección o coordenadas geográficas.
Configuración de puntos de control	Establecer puntos de timbrado con geolocalización precisa y un radio ajustable de 10 metros mediante un mapa selectivo.
Administración de usuarios	Registrar empleados y usuarios para los lugares de trabajo y los eventos
Notificaciones y comunicación	El sistema debe permitir al administrador enviar notificaciones personalizadas a los empleados con información sobre los eventos, cambios de horarios o lugar de trabajo y reuniones.
Registro de asistencia	El sistema debe registrar la entrada y salida de los empleados, validando su ubicación GPS para comprobar que se encuentran dentro el radio permitido.
Reportes de asistencia	El sistema debe incluir reportes de asistencia para los empleados creando la posibilidad de visualizar y comprobar los registros de asistencia de una manera sencilla.

Requerimientos no Funcionales

Tabla 5. Requerimientos no Funcionales.

Requerimiento	Descripción
Usabilidad	Diseñar una interfaz intuitiva y fácil de usar, que permita realizar las funciones principales
Rendimiento	Garantizar tiempos de respuesta inferiores a 2 segundos para las interacciones comunes y generación de reportes en menos de 5 segundos.
Compatibilidad	Asegurar el funcionamiento en dispositivos Android
Portabilidad	Encapsular la aplicación en Docker, asegurando su despliegue y su implementación en cualquier servidor virtual.
Infraestructura	Utilizar PostgreSQL como base de datos dinámica

Historias de Usuario

Tabla 6.HU1 - Configuración de lugares de trabajo.

Código: HU1	Título: Configuración de lugares de trabajo	Prioridad : Alta
Descripción:	Como administrador, quiero poder crear y definir lugares donde se realizarán las tareas, para que los empleados puedan registrar su asistencia en estos lugares específicos	
Criterios de aceptación:	CA1: Debe existir una opción en el menú principal para "Gestión de Lugares".	
	CA2: Al seleccionar esta opción, se debe presentar un formulario para ingresar los datos del lugar..	
	CA3: Cada lugar debe almacenarse con un identificador único y las coordenadas de geolocalización.	
	CA4: El sistema debe permitir editar y eliminar cualquier lugar previamente registrado.	

Tabla 7. HU2 - Establecimiento de puntos de timbrado.

Código: HU2	Título: Establecimiento de puntos de timbrado	Prioridad : Alta
Descripción:	Como administrador, quiero establecer puntos de timbrado con geolocalización, para controlar la asistencia en áreas específicas dentro de un radio de 10 metros.	
Criterios de aceptación:	CA1: La interfaz debe permitir seleccionar un lugar y configurar un punto de timbrado.	
	CA2: Debe permitirse ajustar el radio de acción del punto de timbrado	
	CA3: Cada empleado debe ser agregado al sistema con un identificador único y recibir automáticamente sus credenciales de acceso por correo electrónico.	

Tabla 8. HU3 - Administración de usuarios.

Código: HU3	Título: Administración de usuarios	Prioridad : Alta
Descripción:	Como administrador, quiero establecer puntos de timbrado con geolocalización, para controlar la asistencia en áreas específicas dentro de un radio de 10 metros.	
Criterios de aceptación:	CA1: La interfaz debe permitir seleccionar un lugar y configurar un punto de timbrado.	
	CA2: Debe permitirse ajustar el radio de acción del punto de timbrado	
	CA3: Cada empleado debe ser agregado al sistema con un identificador único y recibir automáticamente sus credenciales de acceso por correo electrónico.	

Tabla 9. HU4. Envío de notificaciones.

Código: HU4	Título: Envío de notificaciones	Prioridad : Meida
Descripción:	Como administrador de recursos humanos, quiero poder seleccionar los empleados y enviarles notificaciones sobre la fecha, hora, motivo y lugar de concentración, para que conozcan fechas relevantes de los eventos y reuniones importantes.	
Criterios de aceptación:	CA1: Debe existir una funcionalidad para que se envíen las notificaciones utilizando Firebase	
	CA2: El administrador puede seleccionar los empleados que recibirán la notificación.	
	CA3: Las notificaciones deben incluir la fecha, la hora y el motivo del evento o reunión.	

Tabla 10. HU5 - Configuración de puestos para eventos.

Código: HU5	Título: Configuración de puestos para eventos	Prioridad : Baja
Descripción:	Como Administrador quiero poder configurar los puestos para los eventos especiales para que los empleados sepan dónde colocarse y a qué hora.	
Criterios de aceptación:	CA1: El sistema debe permitir la creación de eventos y asignar los puestos de trabajo requeridos.	
	CA2: La información relacionada con el evento será suministrada de forma automática a todos los empleados asignados.	

Tabla 11. HU6 - Creación de puntos de encuentro.

Código: HU6	Título: Creación de puntos de encuentro	Prioridad : Baja
Descripción:	Como administrador, quiero poder crear puntos de encuentro, y definir puestos de trabajo móviles, para que los empleados puedan realizar sus tareas en distintas ubicaciones, cuando se realice un evento o actividad especial.	
Criterios de aceptación:	CA1: La creación de puntos de encuentro tendrá que ser configurable a través de la interfaz de administrador.	
	CA2: Cada punto de encuentro tendrá que guardarse con suficiente información (lugar y hora del evento, por ejemplo).	

Tabla 12. Acceso a la aplicación de asistencia.

Código: HU7	Título: Acceso a la aplicación de asistencia	Prioridad : Alta
Descripción:	Como empleado, quiero contar con la posibilidad de descargar la aplicación de asistencia e ingresar en ella, de modo que pueda comenzar a gestionar la asistencia diaria.	
Criterios de aceptación:	CA1: La aplicación tiene que poder descargarse en las plataformas existentes, iOS o Android.	
	CA2: Tiene que existir una autenticación segura para la aplicación de asistencia.	

Tabla 13. HU8 - Registro diario de asistencia.

Código: HU8	Título: Registro diario de asistencia	Prioridad : Alta
Descripción:	Como empleado, quiero registrar mi entrada y salida diariamente dentro de la aplicación para que se me contabilicen mis horas de trabajo	
Criterios de aceptación:	CA1: La aplicación debe permitir registrar la hora de entrada y la hora de salida utilizando geolocalización.	
	CA2: La aplicación debe validar el registro sólo si se realiza dentro del área geográfica preasignada.	

Tabla 14. HU9 - Consulta de reportes de asistencia.

Código: HU9	Título: Consulta de reportes de asistencia	Prioridad : Media
Descripción:	Como empleado, quiero consultar reportes de mi asistencia diaria, para que pueda realizar una verificación y confirmar mis registros.	
Criterios de aceptación:	CA1: Hay que incluir una opción en la aplicación para que se pueda mostrar el historial de asistencia	
	CA2: Los reportes deben incluir las horas de entrada y de salida, así como cualquier incidencia guardada.	

2.1.4. Product Backlog

Tabla 15. Product Backlog.

ID	Historia de Usuario	Prioridad	Módulo o Funcionalidad	Sprint sugerido	Observaciones
HU1	Como administrador, quiero definir lugares de trabajo con coordenadas GPS	Alta	Gestión de lugares	Sprint 1	Incluye formularios, validación y mapa interactivo.
HU2	Como administrador, quiero establecer puntos de timbrado dentro de un radio	Alta	Geolocalización	Sprint 2	Necesario para el control de asistencia.
HU3	Como administrador, quiero registrar empleados y usuarios	Alta	Administración de usuarios	Sprint 2	Requiere formularios y validaciones de datos.
HU4	Como administrador, quiero enviar notificaciones a empleados sobre eventos	Media	Notificaciones push	Sprint 3	Usa Firebase; permite segmentación por usuario o evento.
HU5	Como administrador, quiero configurar puestos de trabajo para eventos especiales	Baja	Eventos y asignaciones	Sprint 4	Datos dinámicos; opción de notificar a usuarios asignados.
HU6	Como administrador, quiero definir puntos de encuentro móviles	Baja	Eventos móviles	Sprint 4	Función de uso en campo; requiere gestión por GPS.
HU7	Como empleado, quiero ingresar a la aplicación de asistencia	Alta	Autenticación e ingreso	Sprint 1	Registro y login, posiblemente con token JWT.
HU8	Como empleado, quiero registrar	Alta	Registro de asistencia	Sprint 2	Validación por GPS; almacena en PostgreSQL.

	mi entrada y salida diaria				
HU9	Como empleado, quiero ver mis reportes de asistencia	Media	Reportes individuales	Sprint 3	Interfaz con filtros y gráficos simples.

2.1.5. Diagramas de casos de uso

El diagrama de casos de uso es un diagrama uml (lenguaje de modelado unificado) que, a través de una representación gráfica del sistema, permite modelar los requerimientos funcionales de una aplicación, mostrándolos desde el punto de vista del usuario.[43]

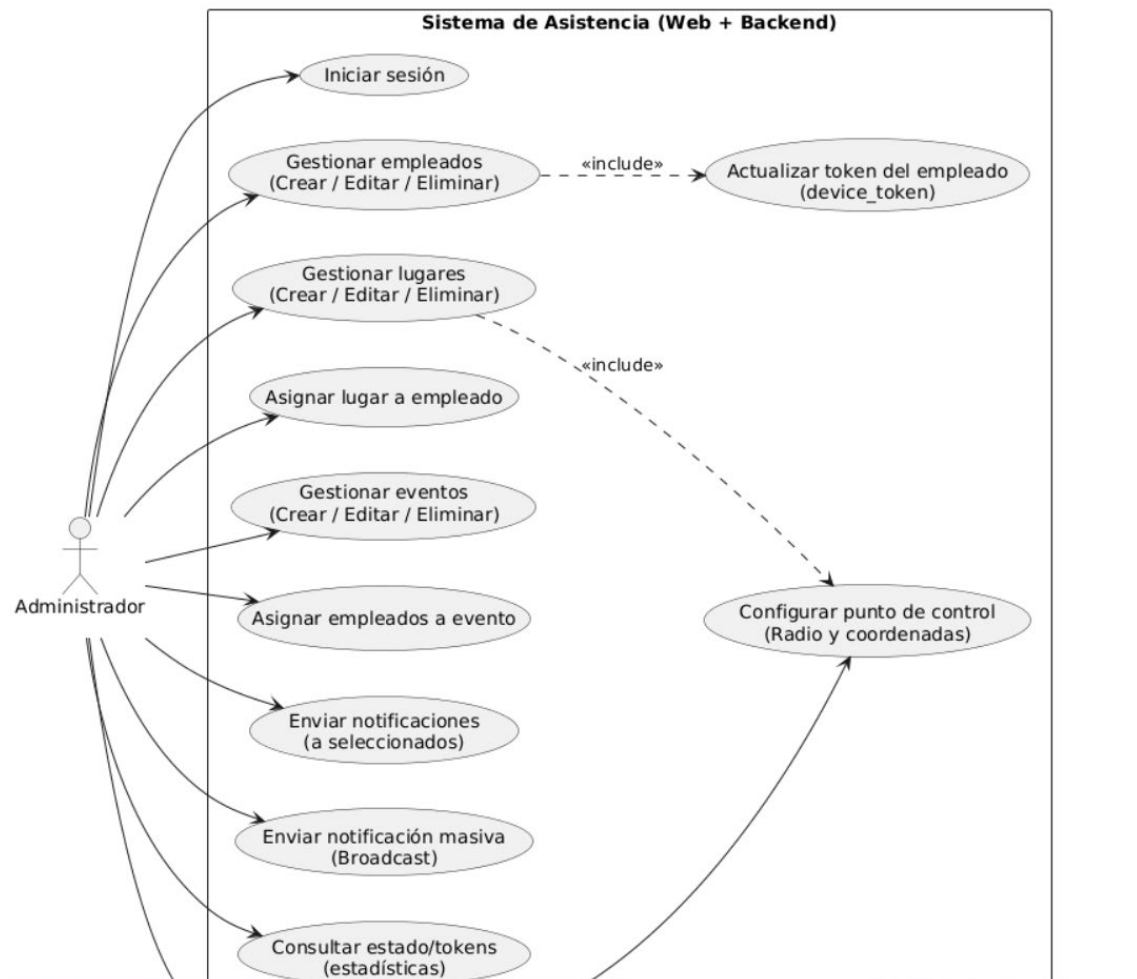


Figura 4. Diagrama de caso de Uso - Sistema de Asistencia (Web + Backend).

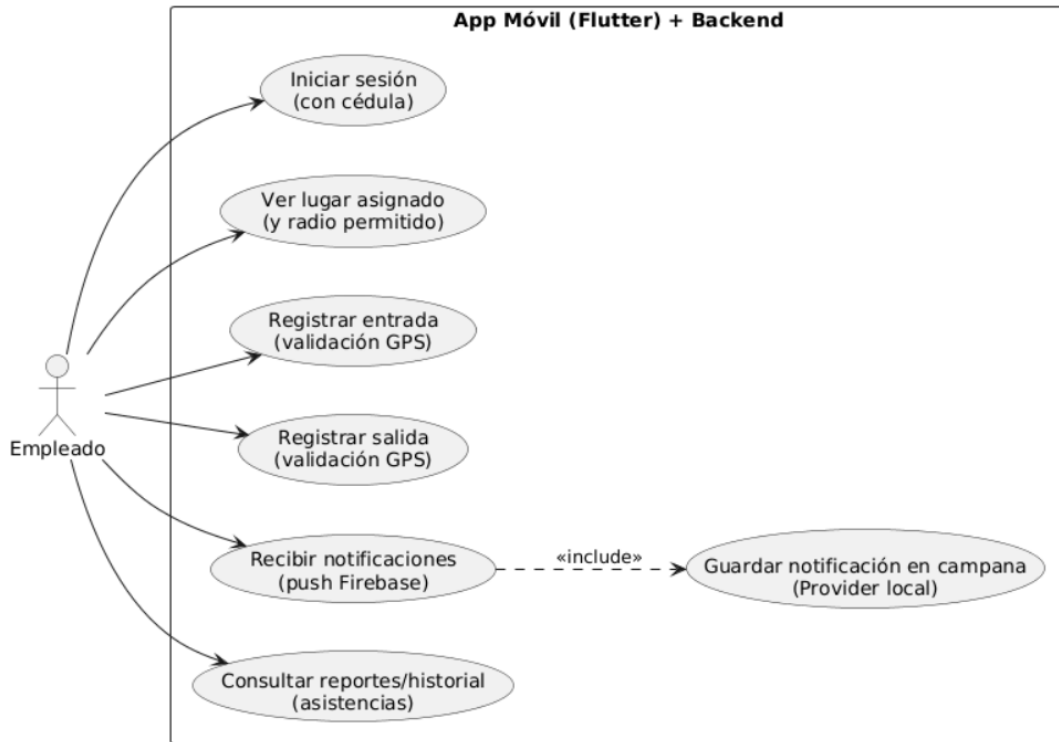


Figura 5. Diagrama de caso de Uso - App Móvil (Flutter) + Backend.

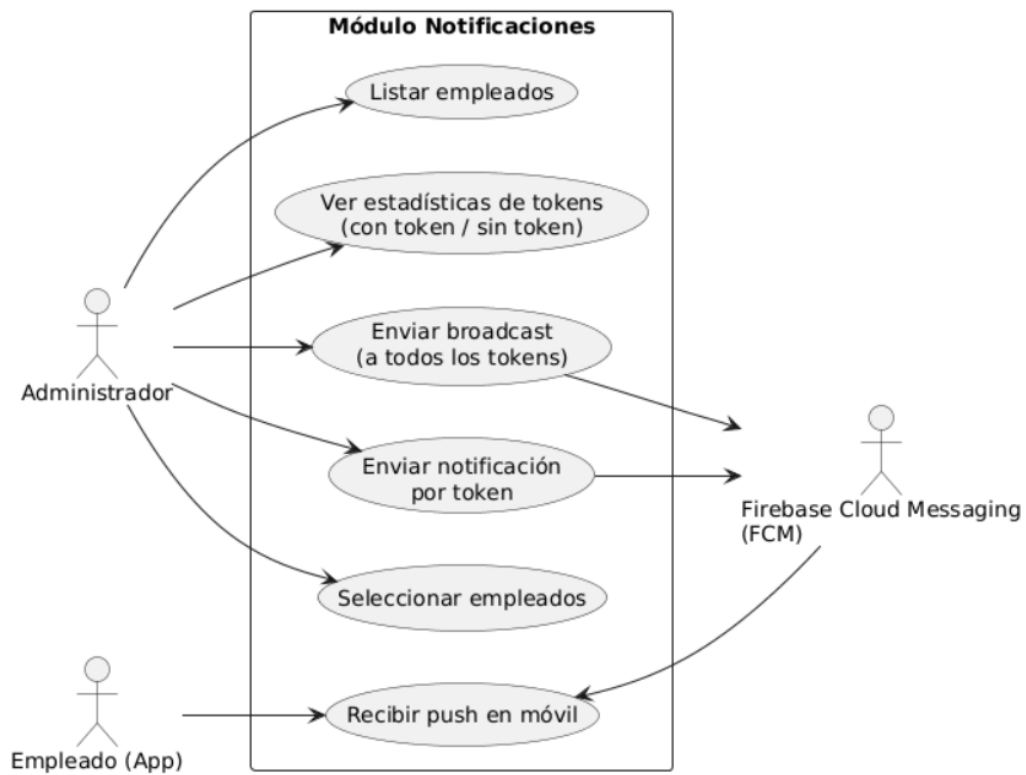


Figura 6. Diagrama de caso de Uso - Módulo Notificaciones.

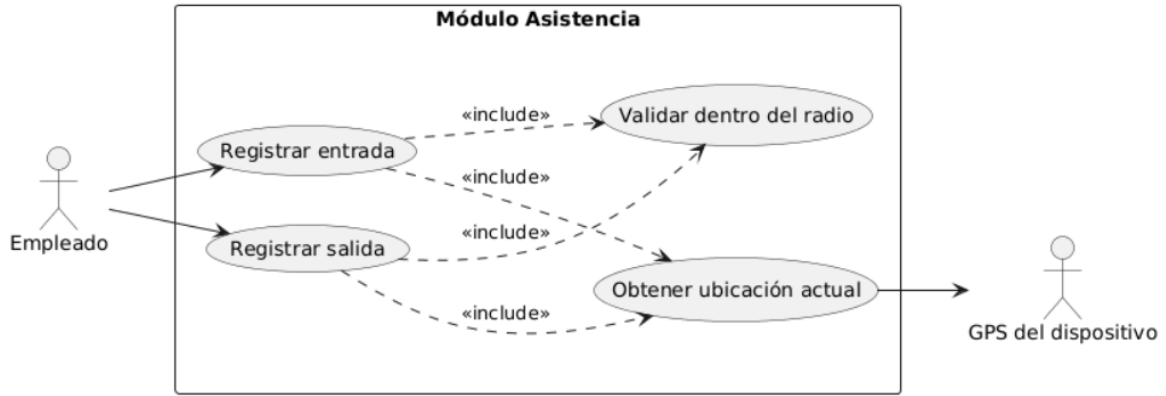


Figura 7. Diagrama de caso de Uso - Módulo Asistencia.

2.1.6. Diagrama de secuencias aplicación móvil

Los diagramas de estado facilitan la representación de los estados clave y acontecimientos que provocan modificaciones en una instancia de clase o en un sistema completo, mientras que los diagramas de secuencia permiten visualizar interacciones entre actores u objetos de clases mediante mensajes.[44]

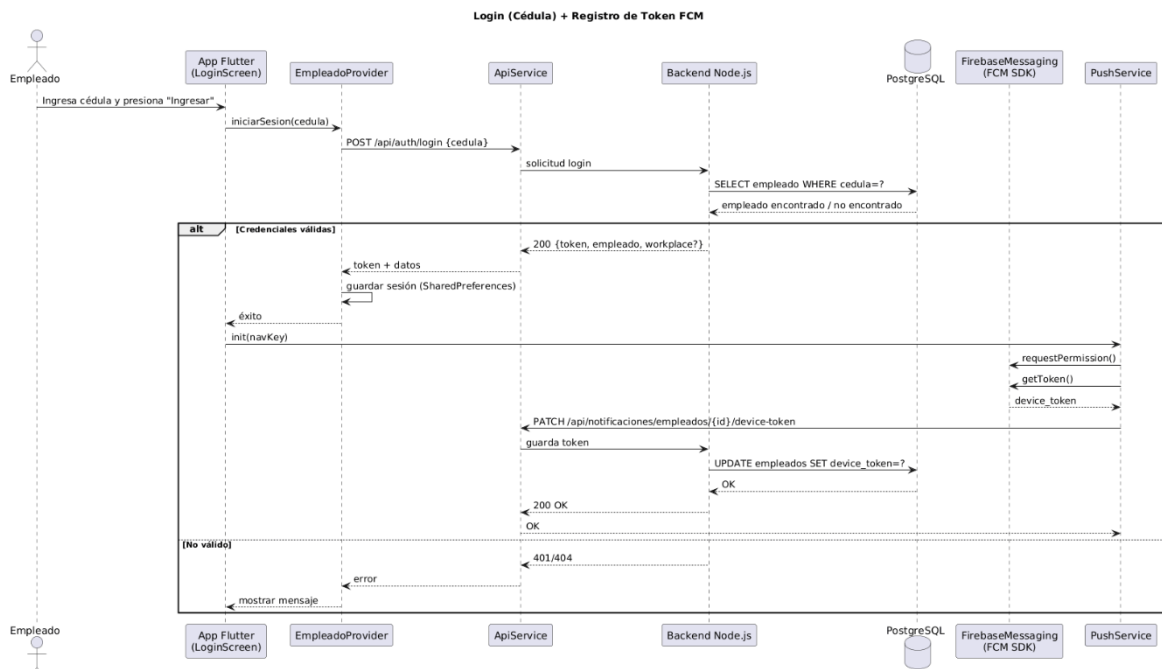


Figura 8. Diagrama de secuencias - Login + Registro de Token FCM.

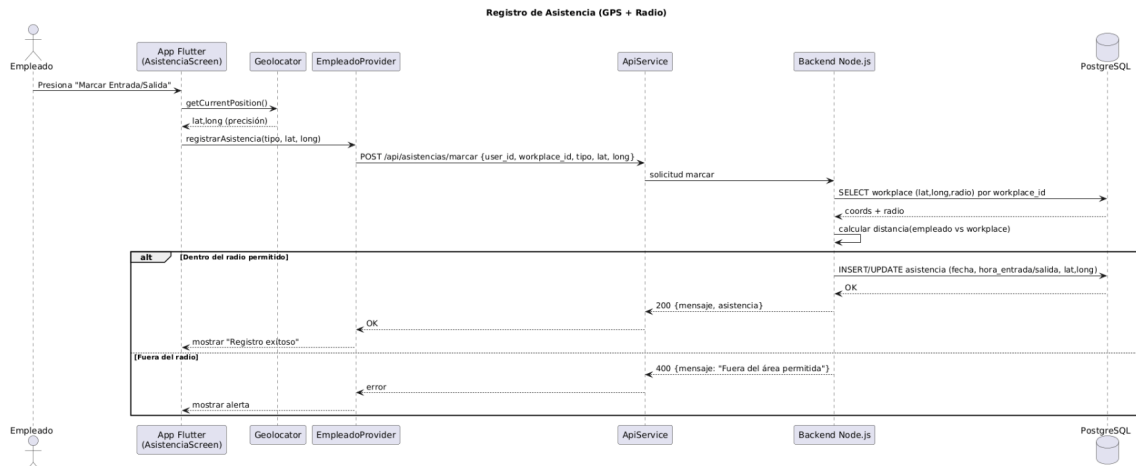


Figura 9. Diagrama de secuencias - Registro de Asistencia (GPS + Radio).

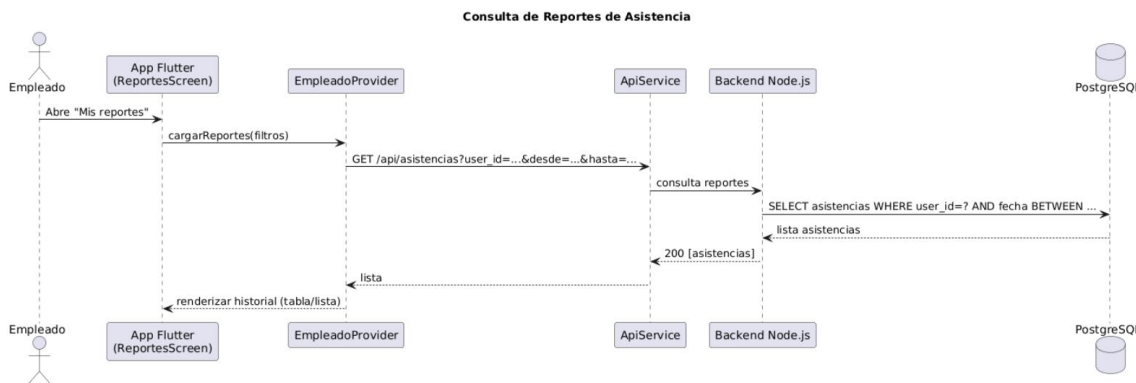


Figura 10. Diagrama de secuencias - Consulta de Reportes de Asistencia

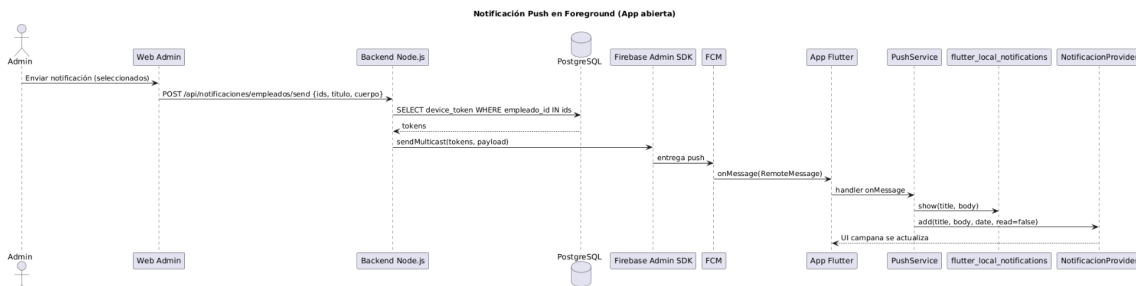


Figura 11. Diagrama de secuencias - Notificación Push en Foreground.

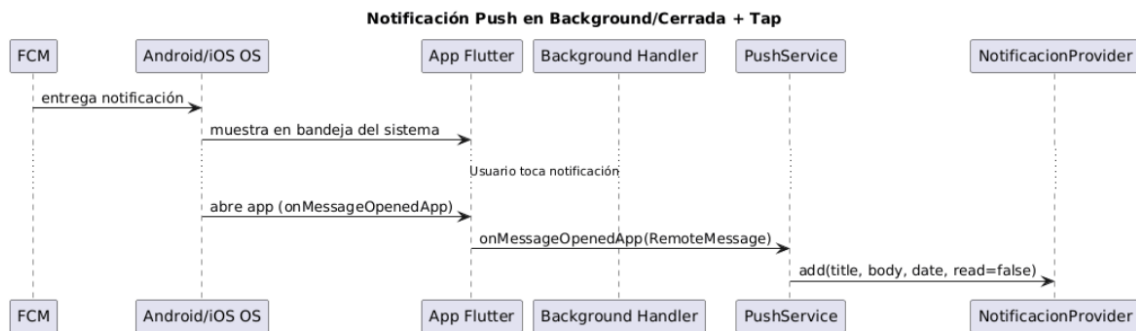


Figura 12. Diagrama de secuencias - Notificación Push.

Refresh de Token FCM

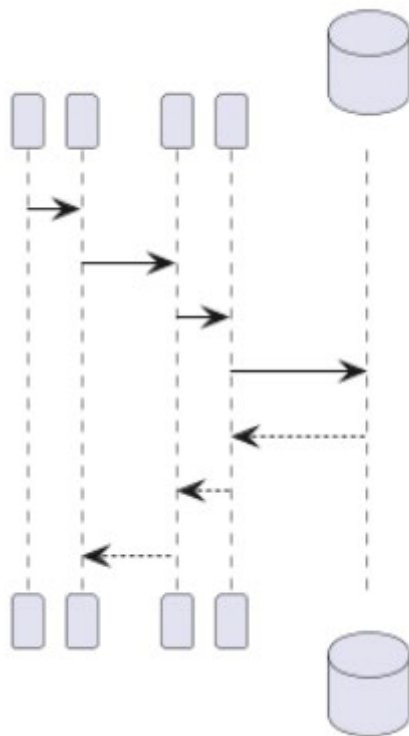


Figura 13. Diagrama de secuencias - Refresh de Token FCM.

2.1.7 Arquitectura del software

Utilizamos un modelo cliente-servidor. Básicamente, hay dos formas de interactuar con él: una app móvil realizada en Flutter para los empleados y una interfaz web para los administradores. Juntas se conectan a una API REST que se desarrolló con Node.js y Express. Esta API es el "cerebro" del sistema: maneja toda la lógica importante, guarda la información en una base de datos PostgreSQL y envía notificaciones push a través de Firebase Cloud Messaging (FCM).

En el backend se implementó el código siguiendo el patrón MVC (Modelo-Controlador-Vista) para que todo quede ordenado y fácil de mantener. En la aplicación móvil, se utilizó una estructura por capas y módulos para que sea limpia y escalable.

En detalle, las partes principales son:

- **App móvil (Flutter):** Es la que instalan los empleados en su celular. Se conecta a la API con peticiones HTTP normales, usa Provider para manejar el estado de la app, valida la asistencia comprobando la ubicación con geolocalización y recibe las notificaciones push cuando corresponde.
- **Interfaz web (Administrador):** Es la sección del software que utilizan los jefes o los administradores. Desde este punto pueden dar de alta o modificar empleados, definir espacios de trabajo, enviar notificaciones y controlar todo lo que hace falta. Además es la interfaz que realiza las peticiones HTTP con la API para realizar las operaciones.
- **Backend (Node.js / API REST):** Es el núcleo del sistema. Es el responsable de recibir todas las peticiones que hagamos desde la app y desde la web y decide (validación, aplicar las reglas del negocio...) qué hacer con ellas, coordinando la comunicación con la base de datos y con Firebase.
- **Lógica de negocio / Servicios:** Se encuentran aquí las reglas más relevantes que comprenden el sistema, por ejemplo, cómo comprobar que un trabajador se encuentra dentro del radio permitido para marcar presencia, cómo gestionar los tokens de notificación y cómo enviar los mensajes push cuando proceda.
- **Base de datos (PostgreSQL):** Almacena de forma ordenada y segura cualquier tipo de información: información de los empleados, de los lugares de trabajo, registros de las asistencias, tokens de dispositivos para las notificaciones, etcétera.
- **Firebase Cloud Messaging (FCM):** Es el servicio que usé para enviar notificaciones push directamente a los celulares de los empleados (por ejemplo, recordatorios o alertas).

- **Docker:** Lo utilicé para empaquetar todo el backend en contenedores. Esto hace que sea mucho más fácil desplegarlo en cualquier servidor virtual, sin preocuparme por diferencias de entorno o dependencias.

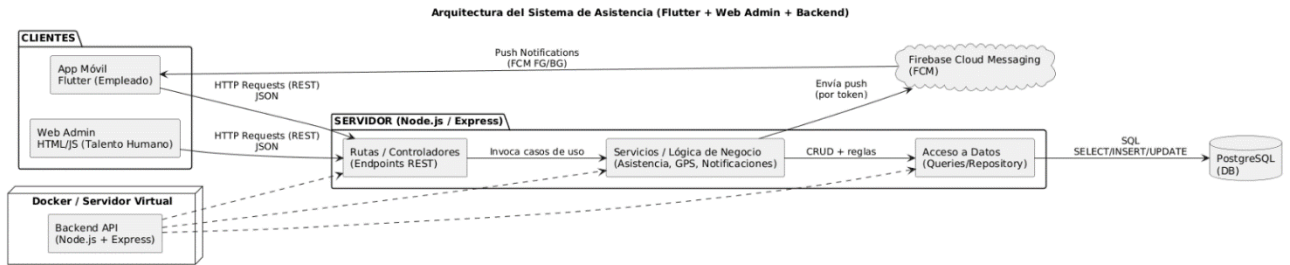


Figura 14. Arquitectura del Software.

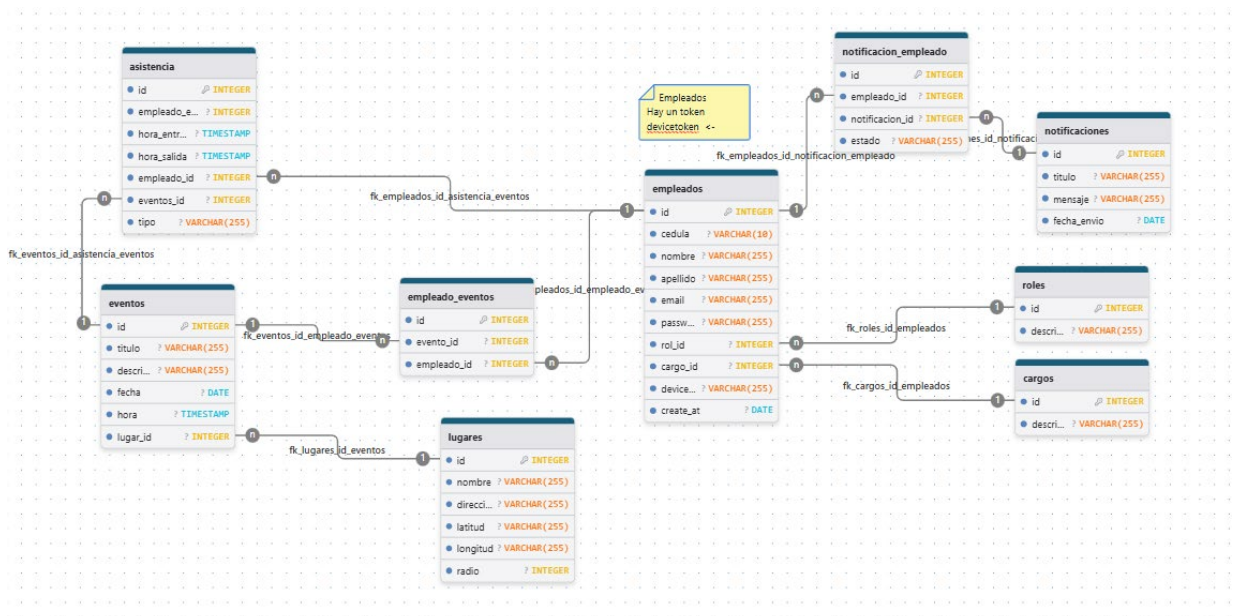


Figura 15. Modelo de datos Lógico.

2.1.8. Ambiente de desarrollo

Lenguajes de programación

Tabla 16. Ambiente de desarrollo - Lenguajes de programación.

Nombre	Descripción
JavaScript	Lenguaje de programación que se utiliza en el desarrollo del backend mediante Node.js y Express. Posibilita la creación de APIs REST y la gestión de las peticiones HTTP entre cliente y servidor.
Dart	Lenguaje de programación que se utiliza en el desarrollo de la aplicación móvil mediante Flutter. Permite generar interfaces dinámicas y con un alto rendimiento.
HTML	Lenguaje de etiquetas utilizado para dar formato el contenido del panel web administrativo.
CSS	Lenguaje de hojas de estilo utilizado para determinar el diseño visual y la presentación del panel web.

Frameworks y entorno de ejecución

Tabla 17. Ambiente de desarrollo - Frameworks y entorno de ejecución.

Nombre	Descripción
Node.js	Entorno de ejecución que permite ejecutar JavaScript del lado del servidor, se utiliza para desarrollar la API del sistema.
Express.js	Express.js minimalista que gestiona el uso de rutas.
Flutter	Framework creado por Google para hacer aplicaciones móviles multiplataforma a partir de una única base de código.

Base de datos

Tabla 18. Ambiente de desarrollo - Base de datos.

Nombre	Descripción
PostgreSQL	Gestor de base de datos de tipo relacional, se utiliza para almacenar empleados, lugares, asistencias, roles y tokens. Asegura la integridad y seguridad de la información.

Herramientas de desarrollo

Tabla 19. Ambiente de desarrollo - Herramientas de desarrollo.

Nombre	Descripción
Visual Studio Code	Editor de código empleado para el desarrollo del backend y de la aplicación móvil.
Postman	Herramienta usada para comprobar y validar los endpoints para la API REST.
Android Studio	Entorno utilizado para compilar y probar la aplicación móvil en dispositivos Android.

Herramientas de control de versiones

Tabla 20. Ambiente de desarrollo - Herramientas de control de versiones.

Nombre	Descripción
Git	Sistema de control de versiones distribuido que permite gestionar cambios en el código fuente.
GitHub	Plataforma en la nube que permite alojar el repositorio del proyecto y facilitar la colaboración.

Plataformas y servicios en la nube

Tabla 21. Ambiente de desarrollo - Plataformas y servicios en la nube.

Nombre	Descripción
Firebase Cloud Messaging (FCM)	Servicio de mensajería en la nube utilizado para el envío de notificaciones push a los dispositivos móviles.

Herramientas de geolocalización

Tabla 22. Ambiente de desarrollo - Herramientas de geolocalización.

Nombre	Descripción
OpenStreetMap	Servicio de mapas que sirve para mostrar ubicaciones y para definir los puntos de control con coordenadas geográficas.

Contenedorización

Tabla 23. Ambiente de desarrollo - Contenedorización.

Nombre	Descripción
Docker	Herramienta empleada para encapsular el back end y la base de datos en contenedores, facilitando el despliegue y la portabilidad del sistema.

2.2. Fase 2: Juego

2.2.1. Sprint 1 — Autenticación y base administrativa

Planificación – Sprint 1

Este sprint implementa el núcleo del sistema: estructura del backend, conexión a PostgreSQL y autenticación (admin y empleado). Se habilitan módulos base en Web Admin y App Flutter para garantizar acceso seguro.

Objetivo del sprint:

Construir el backend base + login admin (web) + login empleado (Flutter) + estructura del proyecto.

Duración del sprint: 72 horas (1 semana)

Tabla – Sprint Backlog (Sprint 1)

Tabla 24. Sprint 1 Backlog.

ID	Historia/Tarea	Esfuerzo (h)
S1-01	Crear proyecto backend Node.js/Express y middlewares	10
S1-02	Configurar PostgreSQL y variables de entorno	4
S1-03	Crear modelos/consultas base (empleados, lugares)	10
S1-04	Endpoint login empleado (por cédula + token)	12
S1-05	Endpoint login admin (web)	10
S1-06	Web Admin: pantalla login y protección de rutas	10
S1-07	Flutter: pantalla login + provider de sesión	10
S1-08	Pruebas de endpoints con Postman + documentación	6
TOTAL		72

Revisión – Sprint 1

Tabla 25. Revisión Sprint 1.

Fecha	(10/03/2025)
Asistentes	Product Owner, Scrum Master, Equipo de desarrollo
Descripción	Se completó la configuración inicial del backend en Node.js y Express, incluyendo la conexión con PostgreSQL. Se implementó el sistema de autenticación para administrador y empleados, así como la estructura base del proyecto siguiendo el patrón MVC. Se validaron los endpoints mediante Postman.

Retrospectiva del Sprint 1

Tabla 26. Retrospectiva del Sprint 1.

Fecha	(20/03/2025)
Asistentes	Product Owner, Scrum Master, Equipo de desarrollo
Título	Descripción
¿Qué salió bien del sprint?	Se completó la creación del backend en Node.js y Express, incluyendo los endpoints de autenticación y la conexión con PostgreSQL. Se estructuró el proyecto bajo el patrón MVC.
¿Qué no salió como se esperaba?	Se presentaron inconvenientes en la configuración inicial del entorno y validación de rutas protegidas.
¿Qué mejoras se implementarán?	Optimizar la organización de los endpoints y fortalecer la validación de errores para mejorar la estabilidad del sistema

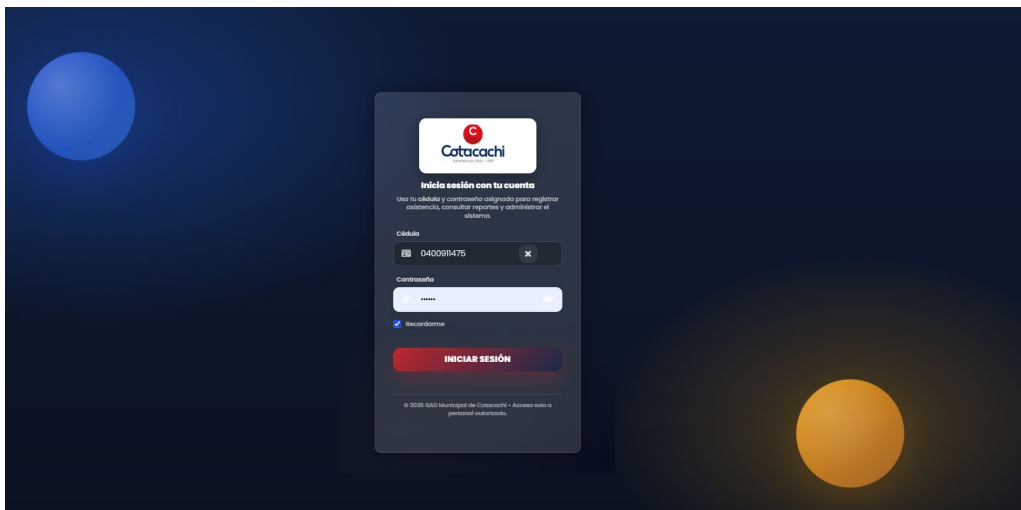


Figura 16. Login Web.



Figura 17. Login móvil.



Figura 18.Home login empleado.

2.2.3 Sprint 2 — Gestión de lugares + geolocalización + asistencia (HU1, HU2, HU8)

Planificación Sprint 2

Se implementa la gestión de lugares y el punto de control (radio + coordenadas), además del registro de asistencia con validación GPS dentro del radio permitido. Esto cubre el corazón funcional del sistema.

Objetivo del sprint:

CRUD lugares + radio + validación GPS + registro entrada/salida.

Duración del sprint: 80 horas (3 semanas)

Tabla – Sprint Backlog (Sprint 2)

Tabla 27. Sprint 2 Backlog.

ID	Historia/Tarea	Esfuerzo (h)
S2-01	Backend: CRUD lugares (crear/editar/eliminar/listar)	12
S2-02	Backend: endpoint para configurar radio/coords del lugar	8
S2-03	Web Admin: vista gestión de lugares + mapa + radio	16
S2-04	Flutter: obtener lugar asignado y radio	10
S2-05	Flutter: permisos + lectura GPS (Geolocator)	10
S2-06	Backend: registrar asistencia (entrada/salida) + validación	16
S2-07	Flutter: UI “Registrar Asistencia” + feedback de validación	8
TOTAL		80

Revisión Sprint 2

Tabla 28. Revisión Sprint 2.

Campo	Información
Fecha	(25/03/2025)
Asistentes	Product Owner, Scrum Master, Equipo de desarrollo
Descripción	Se implementó el módulo de gestión de lugares de trabajo y la configuración de puntos de control con geolocalización y radio de validación. Se desarrolló el registro de asistencia con verificación GPS dentro del perímetro permitido. Se realizaron pruebas funcionales desde la aplicación móvil.

Retrospectiva Sprint 2

Tabla 29. Retrospectiva Sprint 2.

Fecha	(05/04/2025)
Participantes	Product Owner, Scrum Master, Equipo de desarrollo
Título	Descripción
¿Qué ha ido bien del sprint?	La gestión de lugares y la validación de asistencia en geolocalización con radio definido ha funcionado de forma correcta; el registro de entrada-salida ha funcionado conforme lo esperado.
¿Qué no ha funcionado como se esperaba?	La precisión del GPS en algunos dispositivos móviles.
¿Qué mejoras tienen que implementarse?	El radio configurable y el mensaje informativo al usuario cuando se encuentra fuera del área permitida.

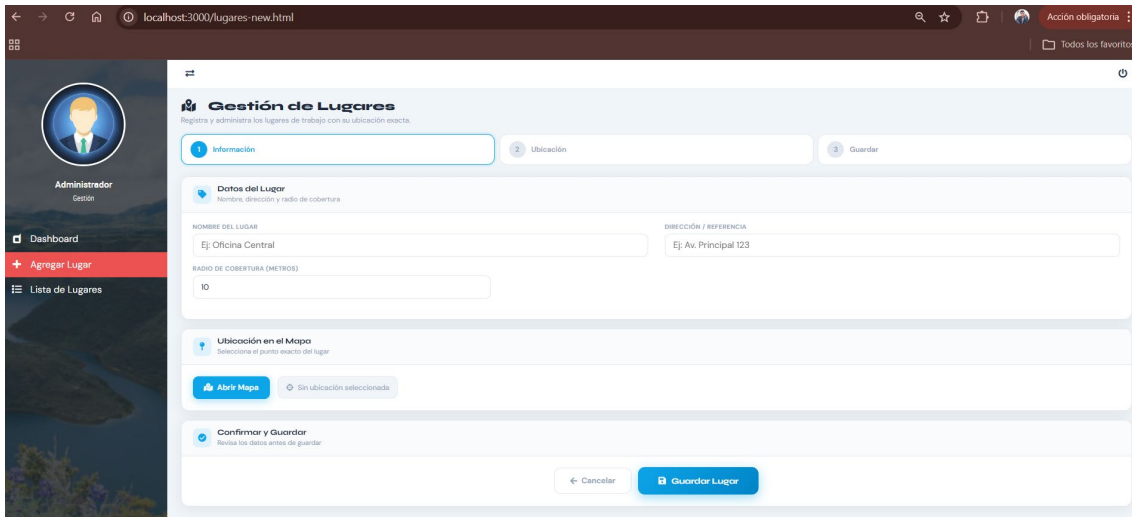


Figura 19. Gestión de Lugares.



Figura 20. Eventos móviles.

2.2.4 Sprint 3 — Empleados + Reportes + Notificaciones (HU3, HU4, HU9)

Planificación – Sprint 3

Se habilita la administración de empleados (incluye asignación de lugar) y el consumo de reportes. También se integra completamente Firebase Cloud Messaging: el móvil guarda token y el administrador envía notificaciones a empleados seleccionados o broadcast.

Objetivo del sprint:

Gestión de empleados + reportes + notificaciones push end-to-end.

Duración del sprint: 90 horas (3 semanas)

Tabla – Sprint Backlog (Sprint 3)

Tabla 30. Sprint 3 Backlog.

ID	Historia/Tarea	Esfuerzo (h)
S3-01	Backend: CRUD empleados + asignación lugar id	14
S3-02	Web Admin: pantalla empleados (CRUD + asignar lugar)	16
S3-03	Backend: endpoints reportes asistencia (por empleado/fecha)	14
S3-04	Flutter: pantalla reportes (historial asistencia)	12
S3-05	Firebase: configuración proyecto + google-services.json	6
S3-06	Flutter: PushService (FG/BG) + guardar token backend	12
S3-07	Backend: guardar device token por empleado	6
S3-08	Web Admin: módulo notificaciones (seleccionados/broadcast)	10
TOTAL		90

Revisión – Sprint 3

Tabla 31. Revisión Sprint 3.

Fecha	(05/06/2025)
Asistentes	Product Owner, Scrum Master, Equipo de desarrollo
Descripción	Se desarrolló el módulo de gestión de empleados y la asignación de lugares de trabajo. Se implementaron los reportes de asistencia para los empleados y se integró Firebase Cloud Messaging para el envío de notificaciones push desde el panel web hacia la aplicación móvil.

Retrospectiva – Sprint 3

Tabla 32. Retrospectiva Sprint 3.

Fecha	(05/08/2025)
Participantes	Product Owner, Scrum Master, Equipo de desarrollo
Título	Descripción
¿Qué ha ido bien del sprint?	Se implementó el módulo de gestión de empleados y se pudo integrar sin problemas Firebase Cloud Messaging para el envío de notificaciones push. Los reportes de asistencia se implementaron correctamente.
¿Qué no ha salido como se esperaba?	Se encontraron incoherencias en la registración de algunos tokens de notificación y errores pequeños en la asignación de empleados a los lugares.
¿Qué mejoras implementaremos?	Implementar más validaciones en el backend para asegurar el registro correcto de los tokens y mejorar la gestión de las asignaciones.

Figura 21. Gestión de empleados.

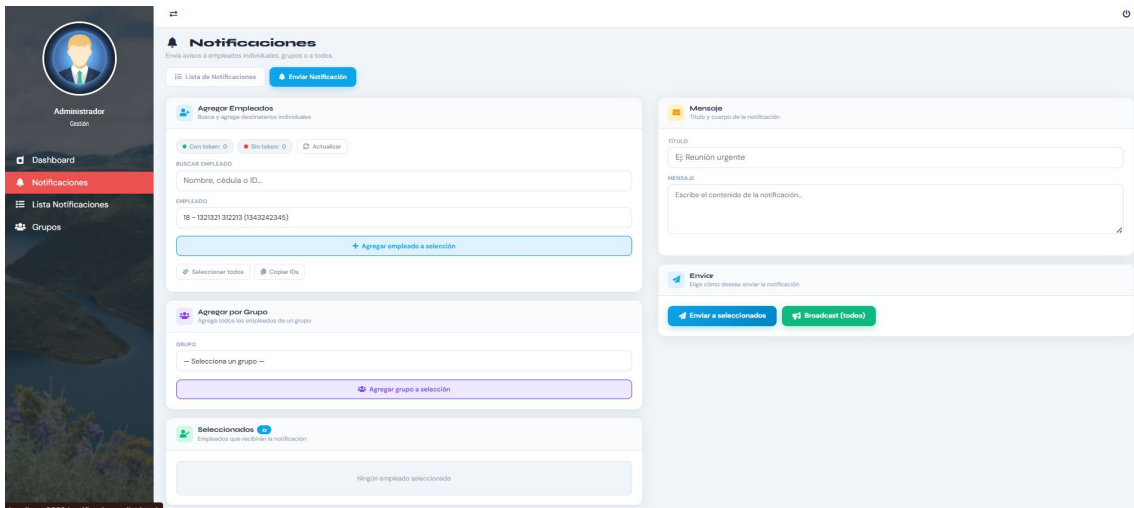


Figura 22. Notificaciones web.

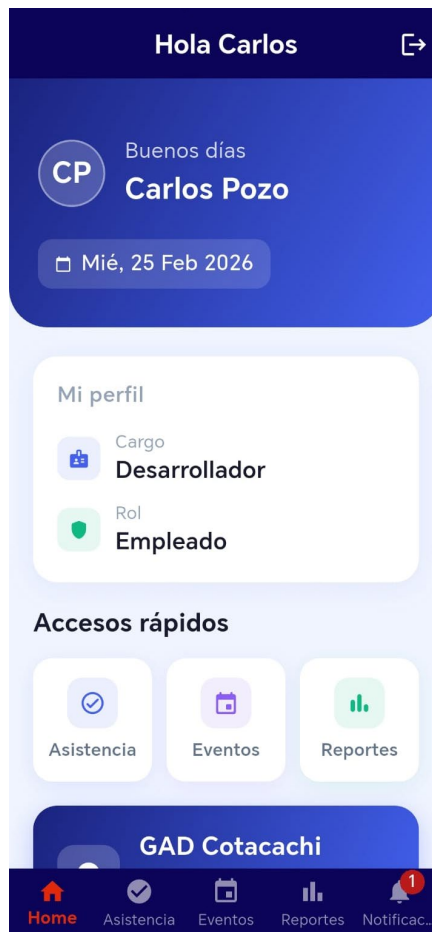


Figura 23. Identificación de notificaciones móviles.

2.2.5 Sprint 4 — Eventos, puntos móviles y cierre (HU5, HU6)

Planificación – Sprint 4

Se agregan eventos y asignaciones (puestos/puntos de encuentro), y se realiza cierre técnico: validaciones, rendimiento, seguridad de acceso por roles, y estabilización general para entrega.

Objetivo del sprint:

Eventos + puntos de encuentro + mejoras finales de calidad y seguridad.

Duración del sprint: 70 horas (2–3 semanas)

Tabla – Sprint Backlog (Sprint 4)

Tabla 33. Sprint 4 Backlog.

ID	Historia/Tarea	Esfuerzo (h)
S4-01	Backend: CRUD eventos	12
S4-02	Backend: asignar empleados a evento + notificar asignados	10
S4-03	Web Admin: módulo eventos + asignaciones	16
S4-04	Flutter: vista eventos/avisos (si aplica)	10
S4-05	Seguridad: roles admin/empleados + protección endpoints	10
S4-06	Optimización y pruebas (rendimiento, errores comunes)	8
S4-07	Documentación final (manual técnico + capturas)	4
TOTAL		70

Revisión Sprint 4

Tabla 34. Revisión Sprint 4.

Campo	Información
Fecha	(10/09/2025)
Asistentes	Product Owner, Scrum Master, Equipo de desarrollo
Descripción	Se desarrolló el módulo de eventos y puntos de encuentro móviles. Se reforzó la seguridad mediante validación de roles y protección de endpoints. Se realizaron pruebas finales de rendimiento, validación de notificaciones y documentación técnica del sistema para su entrega.

Retrospectiva Sprint 4

Tabla 35. Retrospectiva Sprint 4.

Fecha	(20/09/2025)
Asistentes	Product Owner, Scrum Master, Equipo de desarrollo
Título	Descripción
¿Qué salió bien del sprint?	Se completó el módulo de eventos y puntos de encuentro móviles. Se reforzó la seguridad mediante control de roles y protección de endpoints. El sistema quedó estable para su entrega final.
¿Qué no salió como se esperaba?	Se requirieron ajustes adicionales en la optimización del rendimiento y en pruebas finales de integración.

¿Qué mejoras se implementarán?

Mejorar la documentación técnica, optimizar consultas SQL y fortalecer pruebas de validación antes del despliegue definitivo.

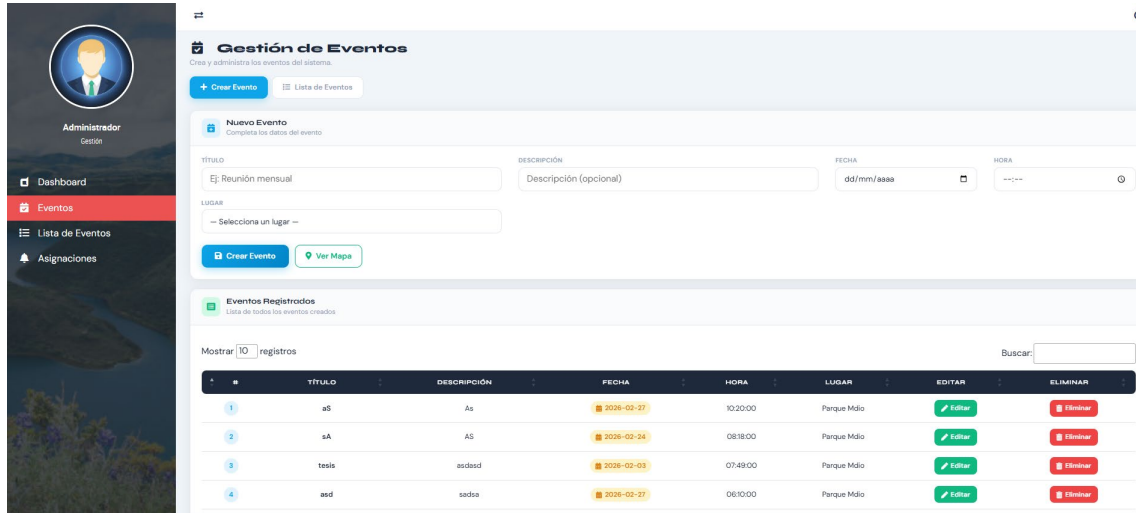


Figura 24. Gestión de Eventos web.

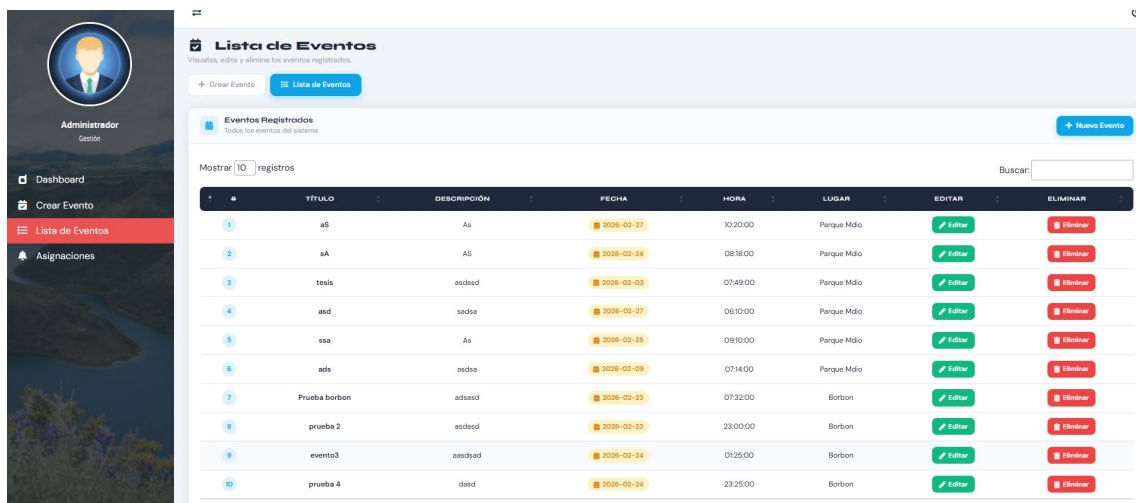


Figura 25. Lista de Eventos web.

2.3. Fase 3: Postjuego

2.3.1. Pruebas de aceptación

La Tabla 36 presenta las pruebas de aceptación realizadas a la aplicación móvil de control de asistencia. En ella se detallan las historias de usuario evaluadas, las funcionalidades verificadas y el estado de aceptación correspondiente. Estas pruebas permitieron validar que el sistema cumple con los requerimientos funcionales definidos en el Product Backlog.

Tabla Pruebas de aceptación – Aplicación móvil de asistencia

Tabla 36. Pruebas de aceptación - Aplicación móvil de asistencia.

ID HU	Nombre	Funcionalidad	Aceptación
			SI
HU-EMP1	Inicio de sesión empleado	Mostrar mensaje de error cuando la cédula no esté registrada.	x
HU-EMP1	Inicio de sesión empleado	Permitir acceso cuando la cédula exista en el sistema.	x
HU-EMP1	Inicio de sesión empleado	Permitir cerrar sesión correctamente.	x
HU-LUG1	Visualizar lugar asignado	Mostrar nombre, ubicación y radio del lugar asignado.	x
HU-LUG1	Visualizar lugar asignado	Mostrar ubicación en mapa (OpenStreetMap).	x
HU-ASIS1	Registrar asistencia	Permitir registrar asistencia solo dentro del radio permitido.	x
HU-ASIS1	Registrar asistencia	Mostrar mensaje de error si el usuario está fuera del rango.	x
HU-ASIS1	Registrar asistencia	Guardar fecha, hora, latitud y longitud en la base de datos.	x
HU-ASIS2	Validación de geolocalización	Solicitar permiso de ubicación al usuario.	x
HU-ASIS2	Validación de geolocalización	Mostrar ubicación actual en el mapa.	x
HU-ASIS2	Validación de geolocalización	Validar distancia mediante cálculo de radio configurado.	x
HU-REP1	Consulta de reportes	Permitir visualizar historial de asistencias registradas.	x
HU-REP1	Consulta de reportes	Mostrar fecha, hora de entrada y salida.	x

HU-NOT1	Notificaciones	Recibir notificaciones push enviadas desde el administrador.	x
HU-NOT1	Notificaciones	Mostrar alerta en primer plano y segundo plano (Firebase FCM).	x
HU-ADM1	Gestión de empleados (Web)	Crear, editar y eliminar empleados.	x
HU-ADM2	Gestión de lugares (Web)	Configurar coordenadas y radio de control.	x
HU-ADM3	Gestión de eventos (Web)	Crear eventos y asignar empleados.	x

III. VALIDACIÓN DE RESULTADOS

3.1. Metodología

3.1.1. Población y muestra

Con el propósito de evaluar la calidad en uso del sistema, se diseñó y aplicó un instrumento de tipo encuesta tomando como referente la norma ISO/IEC 25010 incluido en la familia ISO/IEC 25000, que fueron nos servimos para analizar sus dimensiones de eficacia, eficiencia y satisfacción, utilizando una escala tipo Likert con cinco categorías. Los datos se recogieron a través de una encuesta realizada a un grupo de 55 empleados con el objetivo de establecer el grado de aceptación del sistema en su entorno real de uso.

3.1.2 Instrumento de recolección de datos

Descripción del cuestionario

Tabla 37 - Preguntas del Instrumento de recolección de datos

Dimensión	Preguntas	Tipo de pregunta
Control de asistencia	1. ¿Considera usted que la aplicación móvil le permite registrar su asistencia sin dificultades?	Escala de Likert
	2. ¿Considera usted que la aplicación móvil facilita el control de su asistencia diaria en el Municipio de Cotacachi?	Escala de Likert
	3. ¿Considera usted que la aplicación móvil presenta errores frecuentes al momento de registrar su asistencia?	Escala de Likert
Usabilidad (Facilidad de uso)	4. ¿Considera usted que la aplicación móvil es fácil de usar desde el primer momento?	Escala de Likert
	5. ¿Considera usted que las opciones y botones de la aplicación están claramente identificados?	Escala de Likert
	6. ¿Considera usted que puede utilizar la aplicación móvil sin necesidad de ayuda técnica?	Escala de Likert
	7. ¿Considera usted que el proceso de inicio de sesión en la aplicación móvil es complicado?	Escala de Likert
Eficiencia y rendimiento	8. ¿Considera usted que la aplicación responde rápidamente cuando realiza una acción?	Escala de Likert
	9. ¿Considera usted que la aplicación móvil tarda demasiado en cargar la información?	Escala de Likert
Satisfacción del usuario	10. ¿Considera usted que se siente satisfecho(a) utilizando la aplicación móvil para sus actividades laborales?	Escala de Likert

3.1.3 Escala de Likert

Es una herramienta para medir o recoger datos cuantitativos en la investigación. Es una escala aditiva que se sitúa en un nivel de medición ordinal; incluye una serie de ítems o juicios como afirmaciones para obtener la reacción del sujeto[45]

3.2 Análisis de Datos

3.2.1 Control de asistencia- P1 - Registro sin dificultades

Tabla 38 Tabla de frecuencias de respuestas Control de asistencia -P1- Registro sin dificultades

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	30	54.55%
De acuerdo	17	30.91%
Neutral	5	9.09%
En desacuerdo	3	5.45%
Totalmente en desacuerdo	0	0%
Total	55	100%

Interpretación de resultados

Los resultados muestran que el 85.46% de los empleados (Totalmente de acuerdo + De acuerdo) considera que la aplicación móvil permite registrar su asistencia sin dificultades. Solo el 5.45% manifestó desacuerdo. Estos resultados evidencian que la funcionalidad principal del sistema cumple adecuadamente con el objetivo de control de asistencia institucional.

3.2.2 Control de asistencia -P2- Inicio de sesión complicado

Tabla 39 Tabla de frecuencia de respuestas -Control de asistencia -P2- Inicio de sesión complicado

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	1	1.82%
De acuerdo	0	0%
Neutral	4	7.27%
En desacuerdo	15	27.27%
Totalmente en desacuerdo	35	63.64%
Total	55	100%

Interpretación de resultados

Los resultados muestran que el **90.91%** de los empleados (En desacuerdo + Totalmente en desacuerdo) considera que el proceso de inicio de sesión no es complicado. Esto evidencia que el sistema presenta un mecanismo de autenticación claro y sencillo, facilitando el acceso a la aplicación móvil.

3.2.3 Control de asistencia -P3- Facilidad de uso

Tabla 40 Tabla de frecuencia de respuestas- Control de asistencia -P3- Facilidad de uso

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	29	52.73%
De acuerdo	13	23.64%
Neutral	12	21.82%
En desacuerdo	1	1.82%
Totalmente en desacuerdo	0	0%
Total	55	100%

Interpretación de resultados

El 76.37% de los empleados considera que la aplicación es fácil de usar desde el primer momento. Este resultado confirma que el diseño de la interfaz es intuitivo y comprensible para los usuarios.

3.2.4 Usabilidad -P4- Tiempo de carga excesivo

Tabla 41 Tabla de frecuencia de resouestas-Usabilidad -P4-Tiempo de carga excesivo

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	1	1.82%
De acuerdo	2	3.64%
Neutral	12	21.82%
En desacuerdo	22	40.00%
Totalmente en desacuerdo	18	32.73%
Total	55	100%

Interpretación de resultados

El 72.73% de los empleados está en desacuerdo con que la aplicación tarde demasiado en cargar información. Esto indica que la aplicación presenta un rendimiento adecuado y tiempos de respuesta satisfactorios.

3.2.5 Usabilidad -P5- Botones claramente identificados

Tabla 42 Tabla de frecuencia de respuestas - Usabilidad -P5-Botones claramente identificados

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	37	67.27%
De acuerdo	16	29.09%
Neutral	1	1.82%
En desacuerdo	1	1.82%
Totalmente en desacuerdo	0	0%
Total	55	100%

Interpretación de resultados

El 96.36% de los empleados considera que los botones y opciones están claramente identificados. Esto demuestra una adecuada organización visual y facilidad de navegación en la aplicación

3.2.6 Usabilidad -P6 - Errores frecuentes

Tabla 43 Tabla de frecuencia de respuestas- Usabilidad -P6-Errores frecuentes

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	2	3.64%
De acuerdo	0	0%
Neutral	4	7.27%
En desacuerdo	15	27.27%
Totalmente en desacuerdo	34	61.82%
Total	55	100%

Interpretación de resultados

El 89.09% de los empleados está en desacuerdo con que la aplicación presente errores frecuentes. Esto evidencia estabilidad y confiabilidad en el registro de asistencia

3.2.7 Usabilidad -P7- Uso sin ayuda técnica

Tabla 44 Tabla de Frecuencia de respuestas-P7- Uso sin ayuda técnica

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	19	34.55%
De acuerdo	25	45.45%
Neutral	7	12.73%
En desacuerdo	1	1.82%
Totalmente en desacuerdo	3	5.45%
Total	55	100%

Interpretación de resultados

El 80% de los empleados considera que puede utilizar la aplicación sin necesidad de ayuda técnica, lo que refleja autonomía y facilidad de aprendizaje.

3.2.8 Eficiencia y rendimiento -P8- Respuesta rápida

Tabla 45 Tabla de frecuencia de respuestas -P8-Respuesta rápida

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	34	61.82%
De acuerdo	19	34.55%
Neutral	2	3.64%
En desacuerdo	0	0%
Totalmente en desacuerdo	0	0%
Total	55	100%

Interpretación de resultados

El 96.36% de los empleados considera que la aplicación responde rápidamente al realizar una acción, confirmando eficiencia operativa.

3.2.9 Eficiencia y rendimiento -P9- Satisfacción general

Tabla 46 Tabla de frecuencia de respuestas -Eficiencia y rendimiento -P9- Satisfacción general

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	24	43.64%
De acuerdo	29	52.73%
Neutral	2	3.64%
En desacuerdo	0	0%
Totalmente en desacuerdo	0	0%
Total	55	100%

Interpretación de resultados

El 96.36% de los empleados se siente satisfecho con el uso de la aplicación móvil, evidenciando un alto nivel de aceptación institucional.

3.2.10 Satisfacción del usuario -P10- Facilita el control de asistencia

Tabla 47 Tabla de frecuencia de respuestas-Satisfacción del usuario-P10 -Facilita el control de calidad

Respuesta	Frecuencia	Porcentaje
Totalmente de acuerdo	42	76.36%
De acuerdo	13	23.64%
Neutral	0	0%
En desacuerdo	0	0%
Totalmente en desacuerdo	0	0%
Total	55	100%

Interpretación de resultados

El 100% de los encuestados considera que la aplicación facilita el control diario de asistencia, confirmando el cumplimiento total del objetivo general del proyecto.

Conclusiones

La aplicación móvil desarrollada para el control de asistencia en el GAD Municipal de Cotacachi cumplió su objetivo principal: optimizar el registro y seguimiento de la jornada laboral del personal. Las 55 encuestas aplicadas muestran una alta aceptación: el 85,46 % de los empleados registra su asistencia sin dificultades y el 100 % considera que la aplicación facilita el control diario de su horario.

En usabilidad, el 90,91 % indicó que el inicio de sesión es sencillo, más del 76 % la encontró intuitiva desde el primer uso y el 96,36 % destacó que los elementos de la interfaz son claros y bien identificados.

El sistema tiene buen rendimiento: responde rápidamente, tiene pocos errores y alta estabilidad en el uso real. El 96,36 % de los usuarios manifestó tener una buena satisfacción general, por lo que se considera que hay un tipo de aceptación buena en la institución.

La combinación de Flutter, Node.js con Express, PostgreSQL, Firebase Cloud Messaging y Docker permite construir una solución moderna, escalable y acorde con la norma ISO/IEC 25010, garantizando eficacia, eficiencia y satisfacción de los usuarios.

Recomendaciones

Optimizar el rendimiento del sistema mediante la mejora de las consultas en la base de datos PostgreSQL y la implementación de validaciones más eficientes en el backend, con el fin de garantizar tiempos de respuesta adecuados cuando aumente el número de usuarios dentro del GAD Municipal de Cotacachi.

Fortalecer la seguridad del aplicativo incorporando mecanismos de autenticación más robustos y controles adicionales en la validación de geolocalización, para asegurar la protección de la información institucional y la confiabilidad en el registro de asistencia.

Desarrollar una versión del aplicativo del sistema operativo de iOS con el propósito de tener mayor accesibilidad de la herramienta en la institución y por lo tanto su utilización en un mayor número de dispositivos móviles.

Publicar la aplicación en Google Play Store, con la finalidad de facilitar su distribución oficial, mejorar su visibilidad, permitir acudir a actualizaciones incesantes ya que la entrega mediante archivo APK potencia limitar la entrega de la aplicación y su control de versiones.

BIBLIOGRAFÍA

- [1] A. Danilo, C. Navarrete, S. Tatiana, and P. Rodriguez, “UNIVERSIDAD TÉCNICA DE COTOPAXI EXTENSIÓN LA MANÁ.”
- [2] I. A. Lescano Vásquez, “UNIVERSIDAD TÉCNICA DEL NORTE.” Accessed: May 28, 2024. [Online]. Available: <https://repositorio.utn.edu.ec/handle/123456789/13228>
- [3] R. Andrés and P. Moreno, “UNIVERSIDAD TÉCNICA DEL NORTE.”
- [4] V. M. Murillo, “ESCUELA SUPERIOR POLITECNICA DEL LITORAL ‘DESARROLLO DE APLICACIÓN MÓVIL PARA EMPRESA DE BIENES RAÍCES, VERSIÓN ANDROID’ EXAMEN DE GRADO (COMPLEXIVO),” 2015.
- [5] C. De Ingeniería *et al.*, “UNACH FACULTAD DE INGENIERÍA.”
- [6] Vicente Martinez Orga, “Diseño y Desarrollo de una Aplicación de Gestión del Tiempo,” 2022.
- [7] L. René Quisaguano and J. W. Toca, “Análisis comparativo de entornos de desarrollo móvil,” *Ciencia Latina Revista Científica Multidisciplinar*, vol. 6, no. 4, pp. 4478–4498.
- [8] A. Cabrera Moisés, “Uso básico de Xcode & Interface Builder en MAC OS X usando la librería Cocoa de Objective-C”.
- [9] D. Gavalas and D. Economou, “Development platforms for mobile applications: Status and trends,” *IEEE Softw.*, vol. 28, no. 1, pp. 77–86.
- [10] D. André and D. Rosario, “Comparación de métodos de desarrollo de aplicaciones: Android e iOS”.
- [11] F. Sánchez Rueda, “Comparativa Kotlin y Java en desarrollo Android,” Jan. 2020, Accessed: Jul. 16, 2024. [Online]. Available: <https://hdl.handle.net/10630/19201>
- [12] K. Rahkema and D. Pfahl, “Quality Analysis of iOS Applications with Focus on Maintainability and Security,” in *Proceedings - 2022 IEEE International Conference on Software Maintenance and Evolution*.
- [13] J. Gamaliel, M. Reyes, and M. De Lourdes López García, “Automation of the Codification of the Model-View-Controller Pattern (mvc Pattern) in Projects Oriented to the Web,” 2012.
- [14] F. Pecorelli and A. De Lucia, “Developer-Driven Code Smell Prioritization,” in *Proceedings - 2020, MSR 2020*, Association for Computing Machinery, Inc.
- [15] “PFC_AznarAlberto”.
- [16] A. N. Cadavid, J. and J. Morales Vélez, “Análisis de enfoques ágiles para la creación de software”
- [17] E. Bautista-Villegas, “XP y Scrum, metodologías ágiles utilizadas en el desarrollo de sitios web, bajo MVC, con lenguaje PHP y framework Laravel,” *Revista Amazonía Digital*, vol. 1, no. 1.
- [18] “84934064”.
- [19] “UNIVERSIDAD NACIONAL DE SAN CRISTÓBAL DE HUAMANGA “GESTIÓN DE DESARROLLO ÁGIL DE SOFTWARE USANDO SCRUM Y KANBAN EN PROGRAMACIÓN”.
- [20] A. : Liher, A. Rodríguez, and M. J. Calzado, “ANÁLISIS DEL ALCANCE DE LAS HERRAMIENTAS AGILE EN LA INDUSTRIA 4.0”
- [21] V. Grau, J. Luis, C. Rizo, and J. Luis Vila Grau, “AGILE PROJECT MANAGEMENT ACCORDING TO THE PRINCE2 AND PMBOK MODELS,” pp. 6–9, 2021.

- [22] C. De *et al.*, “UTC FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS AUTORES”.
- [23] A. : Àlex and R. Laborda, “Título: Metodología de gestión de proyectos”.
- [24] Á. : Negocios, “UNIVERSIDAD DEL CEMA”, [Online]. Available: www.cema.edu.ar/publicaciones/doc_trabajo.html
- [25] “República del Ecuador Universidad Tecnológica Empresarial de Guayaquil Trabajo de Titulación Para la Obtención del Título de”.
- [26] I. Bustínduy, “La gestión smart working del tiempo : rendimiento y bienestar en entornos de teletrabajo ,” 2024.
- [27] L. A. M. Zaina, H. Sharp, and L. Barroca, “UX information in the daily work of an agile team: A distributed cognition analysis,” *International Journal of Human Computer Studies*.
- [28] “783”.
- [29] “I”.
- [30] H. J. Patilla *et al.*, “Scrum y Kanban en Programación Extrema para Desarrollo de Software Ágil”.
- [31] H. S. Sverrisdottir, H. T. Ingason, and H. I. Jonasson, “The Role of the Product Owner in Scrum-comparison between Theory and Practices,”
- [32] C. Técnicas Aplicadas, M. I., J. I. Alexandra Núñez-Villacis, and W. I. Clemente Cunuhay-Cuchipe, “Revisión Sistemática de la Metodología Scrum para el Desarrollo de Software,”
- [33] K. Schwaber and J. Sutherland, “The Scrum Guide The Definitive Guide to Scrum: The Rules of the Game,” 2020.
- [34] Henrik. Kniberg and Mattias. Skarin, “Kanban and Scrum : making the most of both,” 2010.
- [35] “jlleon,+Gestor_a+de+la+revista,+A23”.
- [36] M. De *et al.*, “UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA FACULTAD DE CIENCIAS ADMINISTRATIVAS CARRERA ADMINISTRACIÓN DE EMPRESAS ADMINISTRATIVA DE LA EMPRESA HERCHI DE LA PARROQUIA POSORJA, PROVINCIA DEL GUAYAS, AÑO 2020. COMPONENTE PRÁCTICO DEL EXAMEN COMPLEXIVO PREVIO A LA OBTENCIÓN DEL TÍTULO DE LICENCIADO EN”.
- [37] T. Fin De Grado, “UNIVERSITAT POLITÈCNICA DE VALÈNCIA Escuela Politécnica Superior de Gandia Sistema de control de asistencia para alumnos de la UPV”.
- [38] G. Fernández, M. Directores José, I. Gómez, P. Iván, and M. Ortiz, “Seguimiento docente automatizado mediante tarjetas inteligentes sin contacto Automation of teacher attendance tracking using contactless smartcards Trabajo de Fin de Máster Curso 20222023 Autor.”
- [39] P. por and C. Yinmel Castro Buleje, “UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS FACULTAD DE INGENIERÍA ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS LUZ MARILÍN TOMIÑO QUISPE.”
- [40] J. A. Galloso *et al.*, “PLAN DIGITAL DE TRANSFORMACIÓN PARA EL SECTOR DE HIDROCARBUROS Caso: Campo Digital Olympic Perú.”
- [41] E. De Postgrado, “UNIVERSIDAD CATÓLICA SEDES SAPIENTIAE”.
- [42] E. Pérez, “Revista Global Negotium.” [Online]. Available: <https://orcid.org/0009-0004-4623-9823>
- [43] F. Y. Patiño Martínez, “Creación de diagramas de casos de uso”.

- [44] C. L. Vidal and R. H. Villarroel, “Extensión del diagrama de secuencias UML (lenguaje de modelado unificado) para modelado basado en aspectos,”
- [45] S. Margarita and P. Moreno, “Guía práctica para diseñar la Escala Likert,” 2007.

Anexos

Anexo A: Repositorio del Backend

GitHub: <https://github.com/CFabricioP/BackendAsistenciaGad/tree/main>

Anexo B: Repositorio del Frontend Flutter

GitHub: <https://github.com/CFabricioP/FrontendMovilAsistenciaGad>