



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL**

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

**APLICACIÓN MÓVIL CON FLUTTER Y ASP.NET CORE PARA EL
CONTROL DE CUMPLIMIENTO LABORAL EN LA EMPRESA
AUTOREPUESTOS W.M.C.C.**

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la
obtención del título de Ingeniera en Tecnologías de la Información

ÁREA: Base de datos y Sistemas informáticos

LÍNEA DE INVESTIGACIÓN: Software, Tecnología de la Información y
Sistemas de control

AUTOR: Samantha Alexandra Caicedo Coca

TUTOR: Ing. Leonardo David Torres Valverde, Mg.

Ambato - Ecuador

enero – 2026

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: APLICACIÓN MÓVIL CON FLUTTER Y ASP.NET CORE PARA EL CONTROL DE CUMPLIMIENTO LABORAL EN LA EMPRESA AUTOREPUESTOS W.M.C.C., desarrollado bajo la modalidad Proyecto de Investigación por la señorita Samantha Alexandra Caicedo Coca, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que la estudiante ha sido tutorada durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, enero 2026.



Ing. Leonardo David Torres Valverde, Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación con el tema: APLICACIÓN MÓVIL CON FLUTTER Y ASP.NET CORE PARA EL CONTROL DE CUMPLIMIENTO LABORAL EN LA EMPRESA AUTOREPUESTOS W.M.C.C. es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, enero 2026.



Samantha Alexandra Caicedo Coca

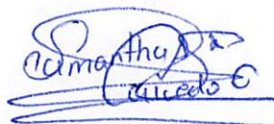
C.C. 1804452009

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, enero 2026.



Samantha Alexandra Caicedo Coca

C.C. 1804452009

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por la señorita Samantha Alexandra Caicedo Coca, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado APLICACIÓN MÓVIL CON FLUTTER Y ASP.NET CORE PARA EL CONTROL DE CUMPLIMIENTO LABORAL EN LA EMPRESA AUTOREPUESTOS W.M.C.C., nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con el señor Presidente del Tribunal.

Ambato, enero 2026.



Ing. Franklin Oswaldo Mayorga Mayorga, Mg.
PRESIDENTE DEL TRIBUNAL



Ing. Marco Guachimboza Villalba, Mg.
PROFESOR CALIFICADOR



Ing. Clay Aldas Flores, Mg.
PROFESOR CALIFICADOR

DEDICATORIA

Le dedico con mucho amor a mis adorados padres, Williams Caicedo y Marisol Coca, por su apoyo incondicional y por nunca dejarme vencer durante todo este proceso.

A mis queridos hermanos Lissette y Josue, por cada momento compartido, cada consejo y por ser mi impulso para salir adelante frente a cualquier adversidad. A mi cuñado por cada palabra de ánimo que me ha brindado. A mi sobrinita Alondra porque su llegada a llenado de alegría y esperanza mi vida.

A mis abuelitas Mercedes Nuñez y Zoila Carrillo, quienes estuvieron apoyándome hasta su último día de vida pues se que su amparo y amor sigue presente en mi corazón.

A mis tíos, por cada palabra de aliento y por demostrarme que todo esfuerzo tiene su recompensa con todo el trabajo duro que cada uno realiza.

A todos ustedes que se encuentran presentes pues han sido parte fundamental de mi proceso, les dedico este logro con todo mi esfuerzo y cariño

Samantha Alexandra Caicedo Coca

AGRADECIMIENTO

Agradezco primero a Dios por permitirme culminar esta etapa de mi vida. A mis amados padres Williams y Marisol, por ser el sostén más grande durante el transcurso de mi carrera, por cada esfuerzo que me enseñaron a valorar las cosas, por inculcarme la fortaleza para nunca rendirme, por cada abrazo y cada gesto que ha guiado mi camino. A mis hermanos Lissette y Josue que siempre me apoyaron y me dieron un abrazo en momentos de dificultad aliviando mis miedos, esto no sería posible sin el amor de ustedes mi querida familia. A mis tíos Joner y Yolanda por ayudarme a cumplir mi sueño, por cada consejo brindado y sobre todo por estar pendiente siempre de mí, su esfuerzo le es recompensado. A mi tía Ruth por todo su apoyo incondicional y por ser mi inspiración que me impulsa a plantearme más propósitos. A mis tíos Rolando y Jessy su apoyo y su dedicación me permite avanzar a paso firme por la vida. A mis amigos Stephy, Isaac, Aly, Xander, por su apoyo emocional en momentos difíciles, gracias por hacer de mi vida universitaria una etapa inolvidable su amistad ha sido fundamental en este proceso. A Bryan Joel, por estar siempre pendiente durante mi vida universitaria, por animarme en cada paso y recordarme que soy capaz, inteligente y fuerte. Tu apoyo ha significado muchísimo para mí.

A mi tutor Ing. Leonardo Torres por su paciencia y su orientación durante el desarrollo de este proyecto. Finalmente, a mis docentes que guiaron mi formación su experiencia profesional fue fundamental para alcanzar mis objetivos.

ÍNDICE GENERAL DE CONTENIDOS

PORTADA.....	i
APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
DERECHOS DE AUTOR	iv
APROBACIÓN DEL TRIBUNAL DE GRADO.....	v
DEDICATORIA.....	vi
AGRADECIMIENTO	vii
ÍNDICE GENERAL DE CONTENIDOS	viii
ÍNDICE DE TABLAS.....	xii
ÍNDICE DE GRÁFICOS.....	xiv
ÍNDICE DE FIGURAS.....	xv
RESUMEN EJECUTIVO	xvii
ABSTRACT	xviii
CAPÍTULO I. MARCO TEÓRICO.....	1
1.1 Tema de investigación.....	1
1.1.1 Planteamiento del problema.....	1
1.2 Antecedentes investigativos	2
1.3 Fundamentación teórica	4

1.3.1 Términos generales	4
1.3.2 Arquitectura.....	6
1.3.3 Metodología	6
1.3.4 IDEs, APIs y Frameworks	7
1.3.5 Lenguajes de Programación	8
1.3.6 Bases de Datos	8
1.4 Objetivos	9
1.4.1 Objetivo general	9
1.4.2 Objetivos específicos	9
CAPÍTULO II. METODOLOGÍA.....	10
2.1 Materiales.....	10
2.2 Métodos.....	10
2.2.1 Modalidad de la investigación	10
2.2.2 Población y muestra	11
2.2.3 Recolección de información.....	12
2.2.4 Procesamiento y análisis de datos	12
CAPÍTULO III. RESULTADOS Y DISCUSIÓN.....	30
3.1 Análisis del nivel de cumplimiento laboral.....	30
3.1.1 Identificación de falencias en los métodos de control existentes.....	32
3.1.2 Impacto en la Organización	33

3.1.3 Propuesta de control de asistencia.....	33
3.2 Análisis de tecnologías y herramientas	34
3.3 Flutter en el desarrollo de aplicaciones móviles	36
3.3.1 Historia y evolución de Flutter.....	37
3.3.2 Requisitos del sistema para desarrollar con Flutter.....	38
3.3.3 Lenguaje de Programación Dart.....	39
3.3.4 Capacidades de Integración con Servicios Externos.....	39
3.3.5 Rendimiento en Aplicaciones de Geolocalización.....	40
3.3.6 Soporte Multiplataforma Real.....	41
3.4 Análisis de metodologías de desarrollo ágil.....	42
3.4.1 Metodología seleccionada	42
3.5 Desarrollo de la propuesta.....	43
3.5.1 Requerimientos	44
3.5.2 Definición del trabajo.....	46
3.5.3 Visualizar fases	46
3.5.4 Desarrollo	47
3.5.5 Pruebas	65
3.5.6 Finalizar antes de empezar	69
3.5.7 Despliegue.....	70
CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES	72
4.1 Conclusiones	72

4.2 Recomendaciones.....	73
REFERENCIAS BIBLIOGRÁFICAS	75
ANEXOS	79

ÍNDICE DE TABLAS

Tabla 1. Población y Muestra.....	11
Tabla 2. Resultados de la pregunta 1.....	13
Tabla 3. Resultados de la pregunta 2.....	14
Tabla 4. Resultados de la pregunta 3.....	15
Tabla 5. Resultados de la pregunta 4.....	16
Tabla 6. Resultados de la pregunta 5.....	17
Tabla 7. Resultados de la pregunta 6.....	18
Tabla 8. Resultados de la pregunta 7.....	19
Tabla 9. Resultados de la pregunta 8.....	20
Tabla 10. Resultados de la pregunta 9.....	21
Tabla 11. Resultados de la pregunta 10.....	22
Tabla 12. Métricas del cumplimiento laboral	30
Tabla 13. Categorización del personal según cumplimiento.....	31
Tabla 14. Comparativa de frameworks móvil.....	34
Tabla 15. Comparativa de frameworks backend.....	35
Tabla 16. Requisitos.....	39
Tabla 17. Tabla multiplataforma	41
Tabla 18. Comparativa de Metodologías de investigación	42
Tabla 19. Etapas de la metodología Kanban	43
Tabla 20. Requerimientos funcionales	44

Tabla 21. Requerimientos No funcionales	45
Tabla 22. Etapa 1 Requerimientos	45
Tabla 23. Etapa 4 Desarrollo.....	47
Tabla 24. Componentes del sistema.....	48
Tabla 25. Apis Admin	53
Tabla 26. Apis Asistencia.....	54
Tabla 27. Apis Auth	54
Tabla 28. Api Registro rápido	55
Tabla 29. Etapa 5 Pruebas	65
Tabla 30. Limitantes.....	69
Tabla 31. Reglas de transición de tareas	69
Tabla 32. Pasos despliegue API	70
Tabla 33. Pasos despliegue aplicación móvil.....	70

ÍNDICE DE GRÁFICOS

Gráfico 1. Resultados de la pregunta 1	13
Gráfico 2. Resultados de la pregunta 2	14
Gráfico 3. Resultados de la pregunta 3	15
Gráfico 4. Resultados de la pregunta 4	17
Gráfico 5. Resultados de la pregunta 5	18
Gráfico 6. Resultados de la pregunta 6	19
Gráfico 7. Resultados de la pregunta 7	20
Gráfico 8. Resultados de la pregunta 8	21
Gráfico 9. Resultados de la pregunta 9	22
Gráfico 10. Resultados de la pregunta 10	23
Gráfico 11. Distribución de conocimiento del control de horarios	31
Gráfico 12. Distribución del personal por nivel de cumplimiento	32

ÍNDICE DE FIGURAS

Figura 1. Logo de Flutter	36
Figura 2. Versiones de Flutter.....	38
Figura 3. Flutter multiplataforma.....	41
Figura 4. Flujo de trabajo en ClickUp.....	46
Figura 5. Tareas del desarrollo del proyecto.....	46
Figura 6. Arquitectura del sistema	48
Figura 7. Relaciones entre tablas	51
Figura 8. Backend ASP.NET Core	52
Figura 9. Estructura base del proyecto Flutter	55
Figura 10. Inicio de sesión de la aplicación	58
Figura 11. Menú Administrador.....	59
Figura 12. Registro de asistencia	59
Figura 13. Registro a otro usuario.....	60
Figura 14. Listar Usuarios.....	60
Figura 15. Crear usuarios	61
Figura 16. Listar horarios.....	61
Figura 17. Crear horarios	62
Figura 18. Verificación de ubicación	62
Figura 19. Registro con cédula	63
Figura 20. Registro de empleados.....	63

Figura 21. Registro exitoso	64
Figura 22. Registro incompleto.....	64

RESUMEN EJECUTIVO

La digitalización de procesos administrativos en las pequeñas y medianas empresas latinoamericanas representa un cambio fundamental hacia la optimización de operaciones mediante herramientas tecnológicas. Esta evolución busca incrementar la precisión en el manejo de información, reducir errores humanos y fortalecer las capacidades de supervisión organizacional. En Ecuador, particularmente en empresas comerciales, mantienen métodos tradicionales que dificulta la eficiencia administrativa. Autorepuestos W.M.C.C., comercializadora de repuestos automotrices en Ambato, muestra esta problemática al mantener un sistema de registro manual de asistencia que genera múltiples inconvenientes con el personal presentan impuntualidad, se registran pérdidas constantes de información y se detectaron modificaciones no autorizadas en los horarios anotados. Esta situación complica la atención al cliente durante la primera hora operativa y limita la toma de decisiones gerenciales con fundamentos.

Por tal motivo esta investigación tiene como objetivo desarrollar e implantar una aplicación móvil capaz de automatizar el registro de asistencia, validar la ubicación geográfica del personal mediante GPS con precisión de 5 metros, y generar reportes en tiempo real que faciliten el control administrativo. El desarrollo tecnológico siguió principios Kanban, implementando Flutter para dispositivos móviles, ASP.NET Core como servidor de servicios y SQL Server para almacenamiento, integrando Google Maps API para verificación. Los resultados revelan aceptación del 93% entre empleados y confianza total en la efectividad del sistema, demostrando que la solución elimina deficiencias del método manual, garantiza integridad de registros y proporciona información verificable para decisiones administrativas.

Palabras clave: Aplicación móvil, control de asistencia, Flutter, Asp.net core, geolocalización, validación gps, Metodología kanban.

ABSTRACT

The digitization of administrative processes in Latin American small and medium-sized enterprises represents a fundamental shift toward optimizing operations through technological tools. This evolution seeks to increase accuracy in information management, reduce human error, and strengthen organizational oversight capabilities. In Ecuador, particularly in commercial enterprises, traditional methods are still in use, hindering administrative efficiency. Autorepuestos W.M.C.C., an automotive parts distributor in Ambato, exemplifies this problem by maintaining a manual attendance record system that generates multiple issues with staff tardiness, constant information loss, and unauthorized modifications to recorded schedules. This situation complicates customer service during the first hour of operation and limits informed management decision-making.

For this reason, this research aims to develop and implement a mobile application capable of automating attendance records, validating the geographical location of personnel using GPS with an accuracy of 5 meters, and generating real-time reports that facilitate administrative control. The technological development followed Kanban principles, implementing Flutter for mobile devices, ASP.NET Core as a service server, and SQL Server for storage, integrating Google Maps API for verification. The results reveal 93% acceptance among employees and total confidence in the effectiveness of the system, demonstrating that the solution eliminates deficiencies in the manual method, guarantees the integrity of records, and provides verifiable information for administrative decisions.

Keywords: Mobile application, attendance control, Flutter, Asp.net core, geolocation, GPS validation, Kanban methodology.

CAPÍTULO I. MARCO TEÓRICO

1.1 Tema de investigación

APLICACIÓN MÓVIL CON FLUTTER Y ASP.NET CORE PARA EL CONTROL DE CUMPLIMIENTO LABORAL EN LA EMPRESA AUTOREPUESTOS W.M.C.C.

1.1.1 Planteamiento del problema

En el contexto actual de transformación digital en América Latina, muchas empresas han comenzado a modernizar sus procesos administrativos mediante herramientas tecnológicas que mejoran la eficiencia y la productividad. Sin embargo, en países como Ecuador, y específicamente en pequeñas y medianas empresas, todavía persisten prácticas tradicionales que dificultan la gestión eficaz del talento humano.

En el ámbito organizacional y sectorial, las Tecnologías de la Información (TI) han desempeñado un papel fundamental en la transformación de los procesos de registro y control de personal dentro de las empresas latinoamericanas. Estas tecnologías han permitido a las organizaciones implementar sistemas automatizados de asistencia, control horario y gestión de recursos humanos, que sustituyen los métodos manuales y reducen significativamente los errores administrativos. En este contexto, herramientas como los sistemas biométricos, aplicaciones móviles, plataformas en la nube y softwares de gestión de talento humano han optimizado el seguimiento del cumplimiento laboral, la puntualidad y la productividad del personal. Además, la integración de las TI ha posibilitado una mayor transparencia y trazabilidad en los procesos de control, facilitando la toma de decisiones basada en datos reales

Este es el caso de la empresa Autorepuestos W.M.C.C., ubicada en Ambato, donde el incumplimiento del horario laboral representa una problemática significativa que impacta directamente en el rendimiento organizacional.

Una de las principales causas de este problema es el uso de registros manuales de asistencia, los cuales resultan poco confiables y propensos a errores. A esto se suma

una débil supervisión en los horarios de entrada y salida, lo que impide un control riguroso del cumplimiento laboral. Además, existe desconocimiento o desinterés por parte de los empleados sobre la importancia de la puntualidad, junto con la ausencia de políticas claras de incentivos o sanciones. Esta combinación de factores genera un ambiente permisivo que dificulta la disciplina horaria.

Como resultado, se producen diversos efectos negativos a nivel operativo: retrasos en los procesos de comercialización, mala atención al cliente por falta de personal a tiempo, desorganización en las labores diarias, y problemas en la planificación de tareas y proyectos. También surgen conflictos internos debido a la desigualdad en el cumplimiento de horarios, lo cual afecta la armonía y el ambiente laboral.

1.2 Antecedentes investigativos

Se realizó una revisión exhaustiva de la literatura científica y académica con el objetivo de identificar estudios relevantes relacionados con el desarrollo de aplicaciones móviles para el control de asistencia laboral, especialmente en microempresas, utilizando tecnologías como Flutter y ASP.NET Core. La búsqueda abarcó el periodo comprendido entre 2021 y 2025, y se enfocó en bases de datos académicas reconocidas, así como en repositorios universitarios nacionales.

Se utilizaron combinaciones de palabras clave como "attendance system", "Flutter", "ASP.NET Core", "mobile application", y "work compliance control" en las bases de datos Scopus e IEEE Xplore. Adicionalmente, se exploraron repositorios académicos de universidades ecuatorianas, como la Universidad Técnica de Ambato (UTA) y la Universidad de las Fuerzas Armadas ESPE. Para garantizar la relevancia y calidad de los estudios seleccionados, se aplicaron criterios de inclusión enfocados en estudios sobre desarrollo de aplicaciones móviles para control de asistencia, aplicaciones desarrolladas con Flutter y ASP.NET Core, y proyectos aplicados en microempresas o pequeñas organizaciones.

En su estudio, Mamoun, Nasor y Abulikailik desarrollaron un prototipo de aplicación móvil para el cuidado de la salud utilizando Flutter. La aplicación permitió a los usuarios programar citas médicas, acceder a perfiles de doctores y gestionar

información personal de salud. La investigación destacó la eficiencia de Flutter para crear interfaces intuitivas y su capacidad para facilitar el acceso a servicios médicos, mejorando la experiencia del usuario en el sector salud [6].

Zefrinaldi, Selviandro y Wulandari analizaron y desarrollaron una aplicación de scouting de fútbol basada en Flutter. La aplicación permitió a los usuarios registrar y buscar información sobre jugadores, facilitando la gestión de datos en tiempo real. El estudio evidenció cómo Flutter puede ser utilizado para desarrollar aplicaciones eficientes en la recopilación y análisis de datos en el ámbito deportivo [7].

Tiwari presentó "Utsarjan", una aplicación móvil desarrollada con Flutter para brindar asistencia en tiempo real a niños con síndrome nefrótico. La aplicación integró funcionalidades como recordatorios de medicación, seguimiento de parámetros de salud y comunicación directa con profesionales médicos. La investigación demostró la eficacia de Flutter en el desarrollo de aplicaciones de salud que requieren interacción constante entre pacientes y médicos [8].

En la investigación desarrollada por Saxena, se diseñó e implementó una aplicación multiplataforma basada en Flutter para el control remoto de recursos Docker. La aplicación proporcionó una interfaz amigable para ejecutar comandos relacionados con Docker en servidores remotos, utilizando protocolos seguros como SSH. Este estudio resaltó la versatilidad de Flutter en la gestión de recursos tecnológicos y su aplicabilidad en entornos de desarrollo y operaciones [9].

Alves y Cipriano exploraron el uso de Flutter en cursos de programación móvil, implementando evaluaciones automatizadas mediante GitHub Classroom. La investigación mostró cómo la integración de Flutter y herramientas de automatización puede mejorar los resultados de los estudiantes, facilitando la retroalimentación y seguimiento del progreso académico [10].

1.3 Fundamentación teórica

El marco teórico proporciona la base conceptual que sustenta la investigación, permitiendo comprender el contexto en el que se desarrolla. A través de la revisión de estudios previos, identifica los aportes existentes y las limitaciones que el proyecto busca abordar. Su elaboración permite establecer las directrices del estudio y aclarar los conceptos fundamentales, garantizando que tanto el investigador como el lector compartan una comprensión común del tema.

1.3.1 Términos generales

a. Tecnologías de la información: comprenden el conjunto de recursos, herramientas y metodologías utilizadas para la gestión y procesamiento de la información. Esto incluye la adquisición, almacenamiento, procesamiento, distribución y protección de datos mediante sistemas informáticos y redes de comunicación. Las TI son fundamentales en la transformación digital de las organizaciones, permitiendo la automatización de procesos, mejora en la toma de decisiones y creación de ventajas competitivas. [11]

b. Informática: es la disciplina que estudia el tratamiento automático y racional de la información mediante el uso de sistemas computacionales. Abarca el diseño, desarrollo y aplicación de hardware y software para la captura, almacenamiento, procesamiento y transmisión de datos. Su campo de acción incluye desde la programación y análisis de algoritmos hasta la implementación de sistemas de información y redes de comunicación, siendo esencial en múltiples áreas del conocimiento y sectores productivos.[12]

c. Desarrollo de software: es un proceso estructurado que involucra la concepción, especificación, diseño, programación, prueba y mantenimiento de aplicaciones y sistemas informáticos. Este proceso se basa en principios de ingeniería para garantizar la calidad, fiabilidad y eficiencia del software producido. Incluye diversas metodologías, como el desarrollo ágil, y prácticas como la integración y la

entrega continuas, que buscan optimizar la producción y adaptación del software a las necesidades cambiantes de los usuarios y del entorno tecnológico. [13]

*d. **Aplicaciones móviles usando flutter:*** flutter es un framework de código abierto desarrollado por Google que permite la creación de aplicaciones móviles, web y de escritorio desde una única base de código. Utiliza el lenguaje de programación Dart y se basa en una arquitectura de widgets altamente personalizables, lo que facilita el desarrollo de interfaces de usuario atractivas y coherentes en múltiples plataformas. Flutter ofrece herramientas como "Hot Reload" para una rápida iteración y pruebas, y su motor de renderizado propio asegura un rendimiento nativo en las aplicaciones desarrolladas.[14]

*e. **Industria automotriz:*** es un sector clave de la economía dedicado al diseño, fabricación y comercialización de vehículos. Impulsa el empleo, la innovación tecnológica y el desarrollo industrial, siendo fundamental en cadenas productivas como la metalurgia, electrónica y logística. [15]

*f. **Control de personal:*** se refiere a los sistemas y procesos implementados por una organización para supervisar y gestionar la asistencia, horarios, desempeño y cumplimiento de las políticas laborales por parte de los empleados. Incluye herramientas como relojes de fichar, software de gestión de recursos humanos y sistemas de evaluación del desempeño. Un control de personal efectivo contribuye a la optimización de los recursos humanos, mejora la productividad y asegura el cumplimiento de las normativas laborales vigentes. [16]

*g. **Reducción de errores:*** en el contexto organizacional implica la implementación de estrategias y prácticas destinadas a minimizar las fallas en procesos, productos o servicios. Esto se logra mediante la estandarización de procedimientos, capacitación continua del personal, uso de tecnologías de automatización y establecimiento de sistemas de control de calidad. La reducción de errores es fundamental para mejorar la eficiencia operativa, aumentar la satisfacción del cliente y reducir costos asociados a retrabajos o reclamaciones. [17]

*h. **Eficiencia operativa:*** es la capacidad de una organización para optimizar el uso de sus recursos humanos, materiales y tecnológicos con el fin de maximizar la

productividad y la calidad de sus productos o servicios, minimizando al mismo tiempo los costos y desperdicios. Se logra mediante la implementación de procesos estandarizados, automatización, formación continua del personal y evaluación constante del desempeño. Una alta eficiencia operativa permite a las empresas adaptarse rápidamente a cambios del mercado, mejorar la satisfacción del cliente y obtener una ventaja competitiva sostenible.[18]

1.3.2 Arquitectura

a. ***Arquitectura de tres capas:*** es un modelo de diseño de software que divide una aplicación en tres niveles independientes: la capa de presentación, encargada de la interacción con el usuario; la capa de lógica de negocio, donde se procesan las reglas y funciones principales del sistema; y la capa de datos, responsable del almacenamiento y acceso a la información. Esta separación facilita la modularidad, escalabilidad y mantenimiento del sistema, ya que permite modificar o actualizar una capa sin afectar las demás. Además, mejora la reutilización del código y la integración con nuevas tecnologías, convirtiéndose en una de las estructuras más utilizadas en el desarrollo de aplicaciones modernas.[19]

1.3.3 Metodología

a. ***Metodología Kanban:*** es una herramienta de gestión ágil que busca optimizar el flujo de trabajo mediante la visualización de las tareas y el control del progreso en tiempo real. Se basa en el uso de un tablero dividido en columnas que representan los diferentes estados del proceso (por ejemplo: “Por hacer”, “En progreso” y “Finalizado”), donde las tareas se organizan en tarjetas que avanzan conforme se completan. Kanban combina la flexibilidad necesaria para un proyecto de investigación-desarrollo con la estructura visual que se necesita para gestionar múltiples tecnologías complejas en un plazo definido, mientras se mantiene una comunicación fluida con la empresa beneficiaria.[20]

1.3.4 IDEs, APIs y Frameworks

- a. ***Android Studio:*** es el entorno de desarrollo integrado (IDE) oficial para la creación de aplicaciones Android, diseñado por Google. Está basado en IntelliJ IDEA y proporciona un conjunto completo de herramientas que facilitan el desarrollo, depuración y prueba de aplicaciones para dispositivos móviles. Entre sus características más destacadas se incluyen un editor de código inteligente con autocompletado, un emulador de dispositivos Android para pruebas, integración con Gradle para la automatización de compilaciones y soporte para lenguajes como Java y Kotlin. Además, permite la gestión eficiente de recursos, interfaces gráficas y versiones del sistema operativo, favoreciendo un proceso de desarrollo ágil y productivo. [21]
- b. ***ASP Net Core:*** es un framework de desarrollo web de código abierto creado por Microsoft que permite construir aplicaciones modernas, escalables y multiplataforma. Está diseñado para ejecutar en sistemas operativos como Windows, Linux y macOS, y soporta la creación de aplicaciones web, APIs, servicios y aplicaciones en la nube. ASP.NET Core utiliza una arquitectura modular y de alto rendimiento basada en el .NET Runtime, lo que facilita la integración con contenedores como Docker y servicios en la nube como Servidor Local. Además, ofrece soporte para Razor Pages, MVC (Model-View-Controller) y Blazor, permitiendo desarrollar interfaces dinámicas y seguras con una gestión eficiente de dependencias y configuración. [22]
- c. ***Google Maps API:*** es un conjunto de interfaces de programación de aplicaciones desarrolladas por Google que permiten integrar mapas interactivos y servicios de localización en aplicaciones web o móviles. A través de sus diversas funciones, los desarrolladores pueden mostrar mapas personalizados, trazar rutas, calcular distancias, ubicar lugares y obtener información geoespacial en tiempo real. Esta API facilita la creación de soluciones basadas en ubicación, como sistemas de navegación, seguimiento de vehículos, geolocalización de usuarios o búsqueda de puntos de interés. Además, ofrece compatibilidad con múltiples lenguajes de programación y plataformas, garantizando una integración flexible y escalable en distintos tipos de proyectos. [23]

1.3.5 Lenguajes de Programación

a. **C#:** es un lenguaje de programación orientado a objetos desarrollado por Microsoft dentro del marco de trabajo .NET, diseñado para ser simple, moderno y seguro. Su sintaxis es similar a la de lenguajes como C++ y Java, lo que facilita su aprendizaje y uso. C# permite desarrollar una amplia variedad de aplicaciones, incluyendo programas de escritorio, aplicaciones web, móviles, juegos con Unity y servicios en la nube. Además, ofrece características como tipado fuerte, manejo automático de memoria, programación asíncrona y soporte para LINQ (Language Integrated Query), que optimizan la eficiencia y legibilidad del código. Gracias a su versatilidad y compatibilidad con múltiples plataformas mediante .NET Core, C# se ha consolidado como uno de los lenguajes más utilizados en el desarrollo de software moderno. [24]

b. **Flutter:** es un framework de desarrollo de interfaces multiplataforma creado por Google que permite construir aplicaciones nativas para Android, iOS, web y escritorio a partir de una sola base de código. Utiliza el lenguaje de programación Dart, también desarrollado por Google, y se destaca por su alto rendimiento gracias a su motor gráfico propio (Skia) que renderiza directamente los componentes visuales sin depender de widgets nativos. Flutter facilita el diseño de interfaces atractivas mediante una amplia colección de widgets personalizables, además de ofrecer una experiencia de desarrollo ágil con funciones como Hot Reload, que permite ver los cambios en tiempo real sin reiniciar la aplicación. Su arquitectura moderna y su capacidad para compilar a código nativo lo convierten en una herramienta eficiente y flexible para el desarrollo móvil y multiplataforma. [25]

1.3.6 Bases de Datos

a. **SQL Server:** es un sistema de gestión de bases de datos relacional (RDBMS) desarrollado por Microsoft, diseñado para almacenar, organizar y administrar grandes volúmenes de información de manera segura y eficiente. Utiliza el lenguaje SQL (Structured Query Language) para la manipulación y consulta de datos, permitiendo

la creación de bases de datos, tablas, procedimientos almacenados y funciones. SQL Server ofrece características avanzadas como replicación, transacciones, respaldo y recuperación, seguridad a nivel de usuario y optimización de consultas, lo que lo convierte en una herramienta robusta para aplicaciones empresariales. Además, se integra fácilmente con otras tecnologías del ecosistema Microsoft, como .NET, Servidor Local y Power BI, facilitando el desarrollo de soluciones completas de gestión y análisis de datos. [26]

1.4 Objetivos

1.4.1 Objetivo general

Implantar una aplicación móvil usando Flutter para el control de cumplimiento al horario laboral para la empresa Autorepuestos W.M.C.C.

1.4.2 Objetivos específicos

- Analizar el nivel de cumplimiento laboral actual de los empleados en la empresa Autorepuestos W.M.C.C., con el fin de identificar falencias en los métodos de control existentes.
- Investigar la tecnología Flutter en el desarrollo de aplicaciones móviles multiplataforma.
- Desarrollar una aplicación móvil usando Flutter para el cumplimiento laboral de los empleados o colaboradores en la empresa Autorepuestos W.M.C.C.

CAPÍTULO II. METODOLOGÍA

2.1 Materiales

- Laptop
- Servicio de Internet
- Impresora
- Frameworks de desarrollo

2.2 Métodos

2.2.1 Modalidad de la investigación

La investigación cuantitativa se centra en la recopilación y el examen de información numérica con el propósito de comprender fenómenos, identificar tendencias, cuantificar variables y establecer correlaciones entre ellas. En el presente trabajo, este método facilitó la evaluación precisa del impacto que tiene la tecnología de geolocalización en la supervisión del cumplimiento horario del personal de Autorepuestos W.M.C.C. Para ello, se obtuvieron datos cuantificables relacionados con la experiencia de los usuarios durante su interacción con la aplicación desarrollada.

En este contexto, se implementaron instrumentos de medición como cuestionarios estructurados y escalas de usabilidad, los cuales permitieron capturar la opinión de los empleados respecto al funcionamiento del sistema y verificar su adecuación a las necesidades operativas. Se consideraron variables críticas tales como la simplicidad en el proceso de marcación de asistencia, la exactitud en la verificación del perímetro geográfico, el desempeño del sistema de reconocimiento facial, la claridad de las alertas generadas y el nivel de aceptación por parte de los usuarios.

El análisis de estos datos generó información concreta y verificable que demuestra cómo la solución tecnológica implementada optimizó los procesos de supervisión laboral, al tiempo que proporciona una experiencia de usuario fluida y accesible. Los resultados obtenidos confirman que la aplicación desarrollada es práctica, segura y replicable en organizaciones con características similares. Así, el método cuantitativo

empleado sustenta de manera sólida tanto la pertinencia como la efectividad de la herramienta digital creada.

Investigación de campo

Esta modalidad se aplicó para acceder de manera directa a la dinámica laboral de la empresa. Se emplearon técnicas como la observación directa, entrevistas semiestructuradas y encuestas aplicadas al personal de distintas áreas, lo que permitió detectar limitaciones en los mecanismos tradicionales de control de asistencia y establecer los requisitos funcionales que debía cumplir el sistema desarrollado [27].

Investigación bibliográfica – documental

Se llevó a cabo un análisis detallado de fuentes académicas, publicaciones científicas, documentación técnica especializada, normativa laboral vigente y recursos sobre construcción de aplicaciones móviles y servicios backend. Este proceso de revisión bibliográfica proporcionó la fundamentación teórica y técnica requerida para diseñar e implementar correctamente la solución tecnológica propuesta [27].

2.2.2 Población y muestra

La población objeto de esta investigación estuvo conformada por el personal administrativo y operativo de la empresa Autorepuestos W.M.C.C. La distribución del personal se detalla en la Tabla 1:

Tabla 1. Población y Muestra

Población	Número	Porcentaje
Personal Administrativo	4	28,57 %
Personal Operativo	6	42,86 %
Supervisores	2	14,29 %
Recursos Humanos	2	14,29 %
Total	14	100 %

Dado el tamaño reducido de la población, se trabajó con una muestra no probabilística por conveniencia, seleccionando directamente a los individuos más involucrados en

los procesos de control de asistencia y gestión del personal. Esta muestra estuvo compuesta por el personal clave, como encargados del registro de entrada y salida, responsables de recursos humanos y empleados de distintas áreas, quienes aportaron información relevante sobre las debilidades del sistema actual y las expectativas sobre la solución digital implementada.

2.2.3 Recolección de información

Para la recolección de datos, se realizó encuestas, entrevista y observación directa. Las encuestas fueron realizadas a todos los trabajadores con el fin de obtener información acerca del nivel de cumplimiento laboral actual e identificar las necesidades para la implementación de una aplicación móvil de control de asistencia.

Así mismo, se realizó una entrevista técnica con el Gerente General, el cual se obtuvo información sobre las necesidades de implementar una aplicación móvil para mantener un control de asistencias de todo el personal.

Finalmente, la observación directa permitió identificar el procedimiento actual para registrar la asistencia encontrando muchas irregularidades que afecta el desempeño laboral

2.2.4 Procesamiento y análisis de datos

La información recolectada fue organizada, depurada y categorizada para su posterior análisis. El procesamiento de datos se realizó mediante herramientas ofimáticas como Microsoft Excel y software estadístico básico, lo cual permitió obtener gráficas, porcentajes y comparaciones significativas. A través del análisis de los resultados, se identificaron patrones, problemas comunes y oportunidades de mejora en el proceso de control laboral, con base en la percepción del personal y el funcionamiento del sistema manual previo.

Se interpretó la información obtenida de las encuestas, entrevistas y observaciones, confrontándola con los objetivos específicos del proyecto. Esto permitió evaluar el grado de aceptación de la solución tecnológica, detectar las principales falencias del sistema anterior de control laboral, y validar la necesidad que justificó el desarrollo de

la aplicación móvil. A partir de estos hallazgos, se estableció una base sólida que orientó el diseño e implementación de la solución tecnológica que responde a las necesidades reales de la empresa Autorepuestos W.M.C.C.

Resultados de la encuesta.

La encuesta se realizó de forma presencial debido al número reducido del personal.

Pregunta 1. ¿Qué tiempo va laborando en la empresa?

Tabla 2. Resultados de la pregunta 1

Indicador	Frecuencia	Porcentaje
Menos de 1 año	2	14%
1 a 3 años	3	21%
3 a 5 años	5	36%
Más de 5 años	4	29%
Total	14	100%

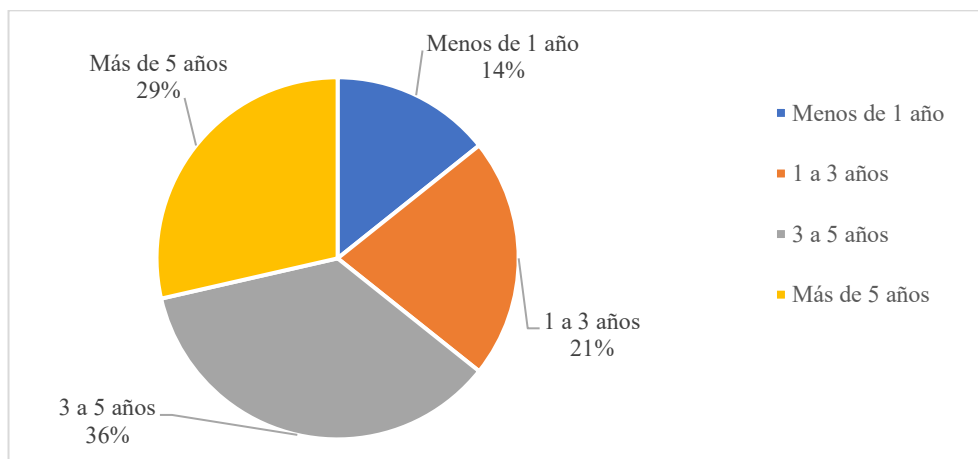


Gráfico 1. Resultados de la pregunta 1

Análisis e interpretación de resultados

Los resultados obtenidos en la Tabla 2 muestran que el 36% de los encuestados tiene una antigüedad de 3 a 5 años, constituyéndose en el grupo más representativo. El 29% indica tener más de 5 años de permanencia, mientras que el 21% cuenta con una

experiencia laboral de 1 a 3 años, y únicamente el 14% labora en la institución desde hace menos de un año.

Estos resultados como se muestra en el Gráfico 1 evidencian que la mayoría del personal (65%) posee más de tres años de antigüedad, lo que refleja un alto nivel de estabilidad laboral y una baja rotación del personal. Este comportamiento puede estar relacionado con la existencia de un ambiente de trabajo favorable, condiciones laborales satisfactorias o estrategias efectivas de retención del talento humano.

Por otra parte, el grupo minoritario con menos de un año de permanencia denota la incorporación de nuevos empleados, lo cual puede asociarse a procesos de expansión, reemplazo de personal o crecimiento institucional.

Pregunta 2. Utiliza actualmente hoja de asistencia manual para registrar su horario.

Tabla 3. Resultados de la pregunta 2

Indicador	Frecuencia	Porcentaje
Siempre	5	36%
A veces	6	43%
Nunca	3	21%
Total	14	100%

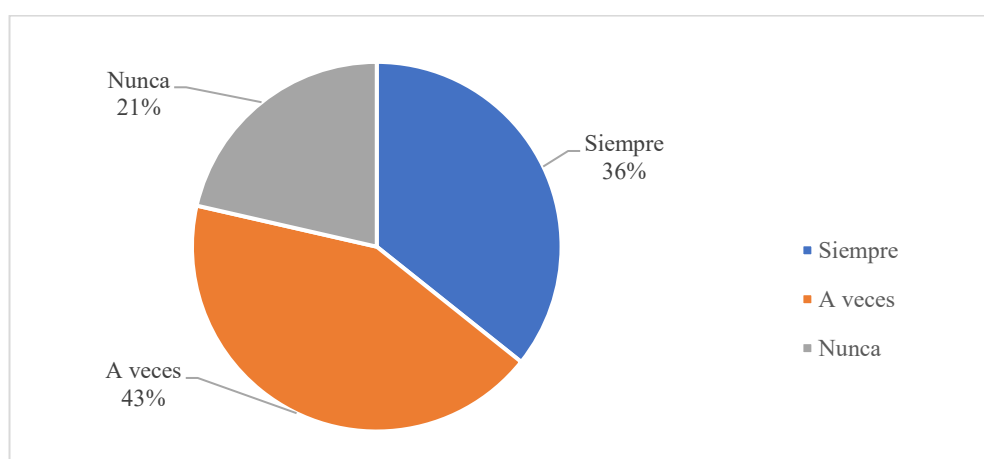


Gráfico 2. Resultados de la pregunta 2

Análisis e interpretación de resultados

De acuerdo con los resultados obtenidos en el Gráfico 2 se observa:

El 43% de los encuestados manifestó que a veces utiliza hojas de asistencia manual, mientras que el 36% indicó que siempre las emplea, y un 21% señaló que nunca recurre a este método.

Estos resultados evidencian que, aunque una parte significativa del personal aún utiliza el registro manual de asistencia, existe una tendencia hacia la disminución de su uso constante, ya que la mayoría lo aplica de forma intermitente o parcial. Esto evidencia una problemática significativa en la consistencia del registro de asistencia, donde la falta de un sistema obligatorio y automatizado permite que los empleados omitan o pospongan el registro según su conveniencia.

Pregunta 3. Los registros de asistencia se pierden con frecuencia.

Tabla 4. Resultados de la pregunta 3

Indicador	Frecuencia	Porcentaje
Siempre	5	36%
A veces	7	50%
Nunca	2	14%
Total	14	100%

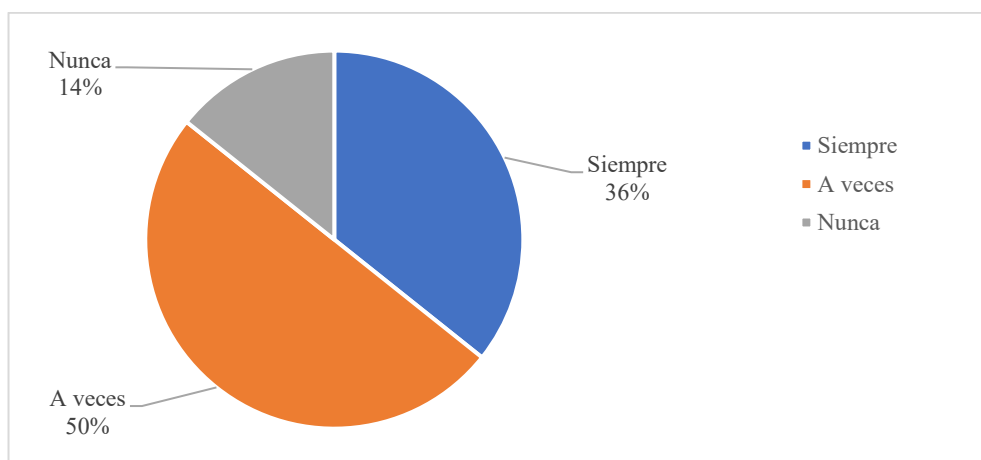


Gráfico 3. Resultados de la pregunta 3

Análisis e interpretación de resultados

Los datos reflejados en el Gráfico 3, manifiesta que el 50% de los encuestados considera que esto ocurre a veces, mientras que el 36% manifestó que siempre se presentan pérdidas en los registros. En contraste, solo el 14% indicó que nunca ha experimentado este tipo de inconvenientes.

Estos resultados evidencian que una mayoría significativa del personal percibe problemas recurrentes o intermitentes en la conservación de los registros de asistencia, lo que sugiere la existencia de fallas en los mecanismos de control y almacenamiento de la información. La prevalencia de respuestas afirmativas (86% entre “siempre” y “a veces”) refleja una debilidad en los procesos administrativos, posiblemente asociada al uso de formatos manuales, errores humanos o ausencia de respaldo digital.

Desde una perspectiva de gestión, estos datos resaltan la necesidad de implementar sistemas tecnológicos automatizados que garanticen la seguridad, trazabilidad y disponibilidad de los registros de asistencia. La digitalización de estos procesos contribuiría a reducir los errores, optimizar el control de personal y fortalecer la eficiencia administrativa de la empresa.

Pregunta 4. Existe un control riguroso del cumplimiento de horarios en la empresa.

Tabla 5. Resultados de la pregunta 4

Indicador	Frecuencia	Porcentaje
Siempre	0	0%
Casi siempre	1	7%
A veces	6	43%
Rara vez	4	29%
Nunca	3	21%
Total	14	100%

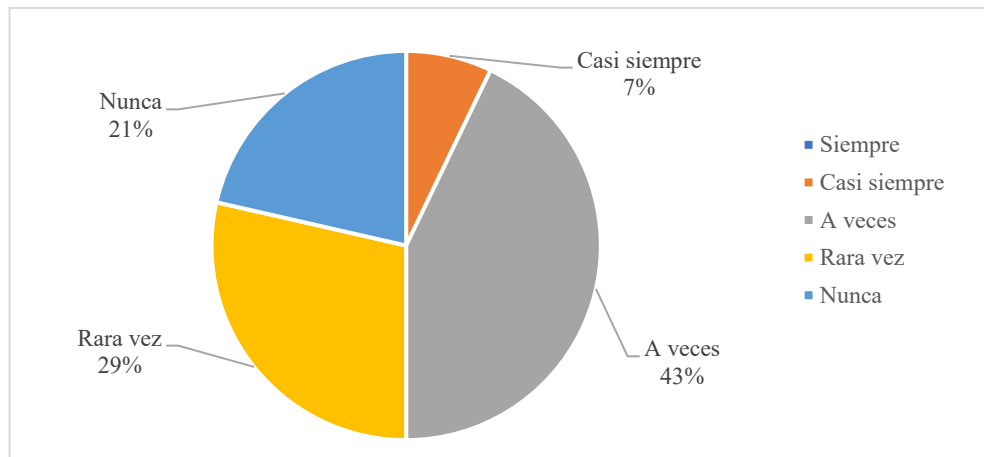


Gráfico 4. Resultados de la pregunta 4

Análisis e interpretación de resultados

En el Gráfico 4, se observa que el 43% de los encuestados considera que a veces se lleva un control estricto de los horarios laborales. Asimismo, el 29% opina que esto ocurre rara vez, mientras que el 21% manifestó que nunca se aplica un control riguroso. Solo el 7% indicó que casi siempre se cumple con esta práctica, y ninguno de los participantes seleccionó la opción siempre.

Esta distribución de respuestas evidencia que en la empresa no existe un control eficiente del cumplimiento de horarios, ya que la mayoría de los trabajadores percibe una supervisión irregular o limitada. La ausencia de un control riguroso puede generar incumplimientos en los horarios establecidos, pérdida de productividad y dificultades en la evaluación del desempeño laboral.

Pregunta 5. ¿Posee un teléfono inteligente (smartphone – teléfono celular)?

Tabla 6. Resultados de la pregunta 5

Indicador	Frecuencia	Porcentaje
Si	14	100%
No	0	0%
Total	14	100%

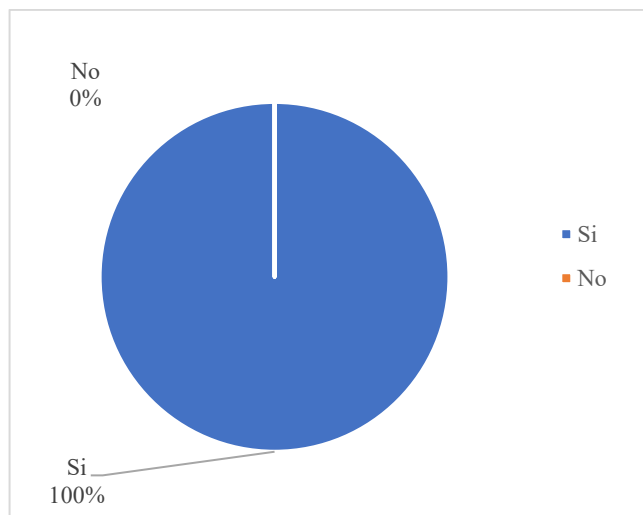


Gráfico 5. Resultados de la pregunta 5

Análisis e interpretación de resultados

Los resultados obtenidos como se muestra la Tabla 6, evidencian que el 100% de los encuestados manifestó sí disponer de un teléfono inteligente, mientras que ninguno indicó no contar con este tipo de dispositivo.

Este resultado demuestra una total disponibilidad tecnológica entre los trabajadores, lo cual representa una condición favorable para la implementación de la aplicación móvil dentro del entorno laboral.

Pregunta 6. ¿Se siente cómodo utilizando aplicaciones móviles?

Tabla 7. Resultados de la pregunta 6

Indicador	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	0	0%
A veces	3	21%
De acuerdo	8	57%
Totalmente de acuerdo	3	21%
Total	14	100%

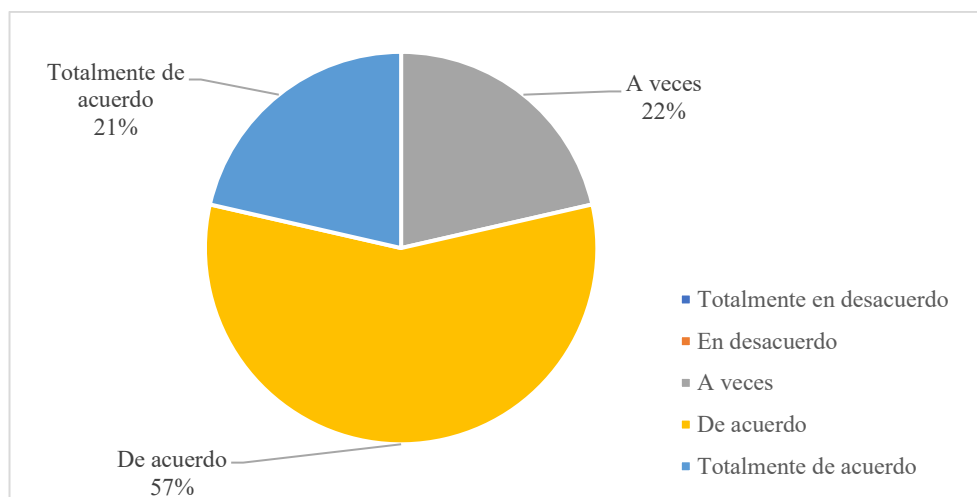


Gráfico 6. Resultados de la pregunta 6

Análisis e interpretación de resultados

En el Gráfico 6, se aprecia que la mayoría de los encuestados, equivalente al 57%, manifestó estar de acuerdo con sentirse cómodo al usar este tipo de herramientas tecnológicas. Asimismo, un 21% expresó estar totalmente de acuerdo, mientras que el 21% restante indicó que a veces se siente cómodo al utilizarlas. Es importante destacar que ninguno de los participantes señaló estar en desacuerdo o totalmente en desacuerdo, lo que refleja una percepción positiva generalizada.

La comodidad manifestada por la mayoría de los empleados indica que las aplicaciones móviles pueden ser implementadas con facilidad en los procesos administrativos y de control de asistencia, sin generar resistencia al cambio.

Pregunta 7. ¿Estaría dispuesto a utilizar una aplicación móvil para registrar su asistencia?

Tabla 8. Resultados de la pregunta 7

Indicador	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	0	0%
A veces	2	14%
De acuerdo	5	36%
Totalmente de acuerdo	7	50%
Total	14	100%

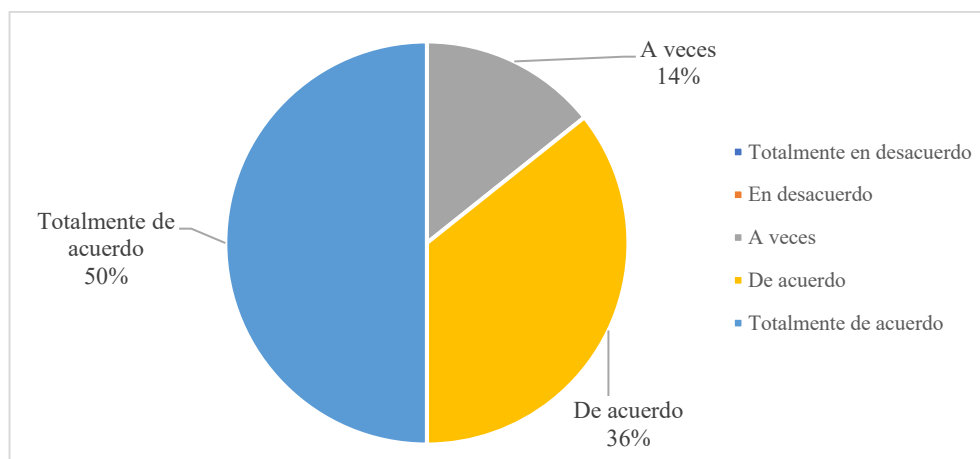


Gráfico 7. Resultados de la pregunta 7

Análisis e interpretación de resultados

Los resultados obtenidos en la pregunta 7 como se observa en el Gráfico 7, muestra que el 50% de los encuestados está totalmente de acuerdo, el 36% de acuerdo, y solo el 14% respondió a veces. Ninguno manifestó desacuerdo.

Estos resultados indican una alta disposición del personal para emplear herramientas tecnológicas en el control de asistencia. La mayoría demuestra confianza y apertura hacia la digitalización de procesos, lo que evidencia un entorno favorable para implementar aplicaciones móviles dentro de la empresa.

Pregunta 8. ¿Considera importante la automatización de entrada y salida?

Tabla 9. Resultados de la pregunta 8

Indicador	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	0	0%
A veces	1	7%
De acuerdo	6	43%
Totalmente de acuerdo	7	50%
Total	14	100%

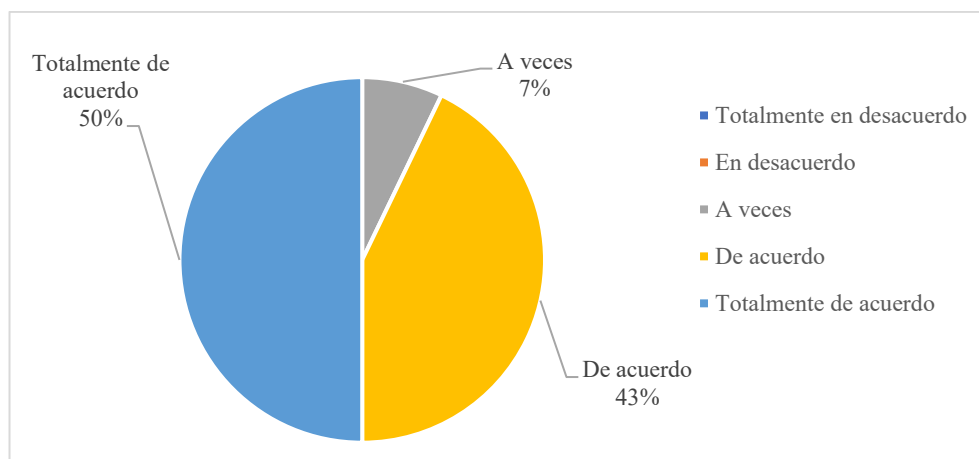


Gráfico 8. Resultados de la pregunta 8

Análisis e interpretación de resultados

Los resultados obtenidos muestran que el 50% de los encuestados está totalmente de acuerdo con que el registro automático de entrada y salida es importante, mientras que el 43% indicó estar de acuerdo y solo el 7% respondió a veces. Ninguno manifestó desacuerdo.

Estos datos como se observa en el Gráfico 8 evidencian que la mayoría del personal valora positivamente la automatización de los registros de asistencia, considerando que este sistema contribuye a una mayor eficiencia y control dentro de la empresa. La tendencia general refleja una actitud favorable hacia la implementación de soluciones tecnológicas que optimicen los procesos administrativos.

Pregunta 9. ¿Estaría de acuerdo con la verificación facial durante el registro de asistencia para evitar suplantaciones?

Tabla 10. Resultados de la pregunta 9

Indicador	Frecuencia	Porcentaje
Totalmente en desacuerdo	6	43%
En desacuerdo	5	36%
A veces	0	0%
De acuerdo	2	14%
Totalmente de acuerdo	1	7%
Total	14	100%

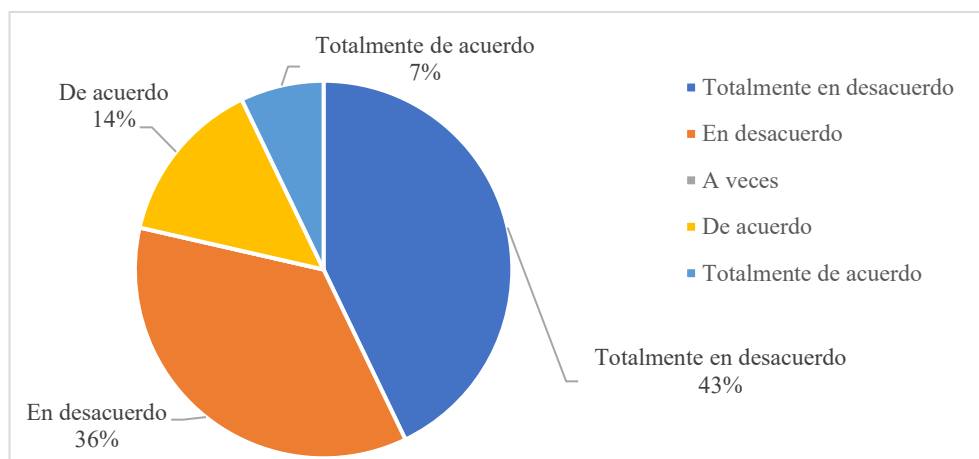


Gráfico 9. Resultados de la pregunta 9

Análisis e interpretación de resultados

Los resultados obtenidos en la Tabla 10 muestran que el 43% de los encuestados está totalmente en desacuerdo con la verificación facial durante el registro de asistencia, mientras que el 36% indicó estar en desacuerdo. Solo el 14% manifestó estar de acuerdo y el 7% totalmente de acuerdo.

Estos datos reflejan una baja aceptación del uso de la verificación facial, posiblemente por preocupaciones relacionadas con la privacidad o la ley de protección de datos. En general, el personal muestra resistencia a los métodos biométricos, lo que se opta por eliminar esa opción.

Pregunta 10. ¿Cree que la empresa será más eficiente en el control de asistencia con este sistema?

Tabla 11. Resultados de la pregunta 10

Indicador	Frecuencia	Porcentaje
Totalmente en desacuerdo	0	0%
En desacuerdo	0	0%
A veces	0	0%
De acuerdo	6	43%
Totalmente de acuerdo	8	57%
Total	14	100%

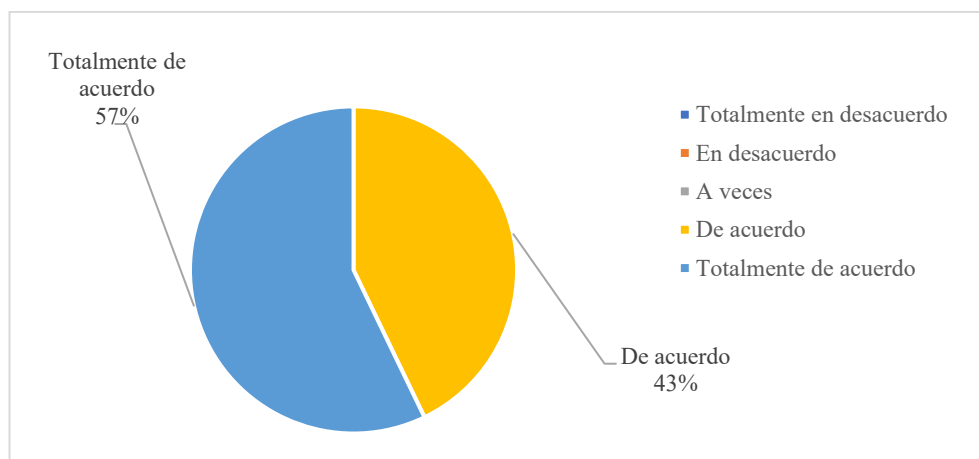


Gráfico 10. Resultados de la pregunta 10

Análisis e interpretación de resultados

Los resultados que se observan en la Tabla 11 muestran que el 57% de los encuestados está totalmente de acuerdo y el 43% de acuerdo con que la empresa sería más eficiente en el control de asistencia al implementar este sistema. Ninguno manifestó desacuerdo.

Estos datos evidencian una percepción altamente positiva respecto a la eficacia del nuevo sistema de control de asistencia, lo que refleja confianza del personal en la tecnología y su impacto favorable en la gestión organizacional. En conjunto, los resultados indican que existe una aceptación generalizada hacia la modernización de los procesos administrativos mediante herramientas digitales.

Resultados de la entrevista

ENTREVISTA

Lugar y fecha: Ambato, 17 de septiembre de 2025.

Entrevistado: Jonathan Alexis Sánchez Alvarado.

Cargo: Gerente general.

Nº	Pregunta	Respuesta
1	¿Podría describir con detalle cómo funciona actualmente el proceso de control de asistencia en la empresa?	Actualmente utilizamos un sistema manual basado en hojas de registro físicas. Cada empleado debe firmar al momento de ingresar y al salir de la empresa. El encargado de recursos humanos recopila estas hojas semanalmente para consolidar los datos en una hoja de cálculo de Excel. Este proceso consume aproximadamente 4 horas semanales y presenta errores frecuentes debido a firmas ilegibles, olvidos de registro o duplicidad de información.
2	¿Qué herramientas	Contamos con una carpeta de asistencia ubicado en la recepción principal donde los empleados anotan su

	específicas utilizan para registrar la entrada y salida del personal?	nombre, hora de entrada y hora de salida. Posteriormente, esta información se transcribe manualmente a una planilla de Excel para llevar el control mensual. No disponemos de ningún sistema automatizado ni software especializado para esta función.
3	¿Puede describir cómo la impuntualidad o inasistencia afecta específicamente las operaciones de venta y atención al cliente?	La impuntualidad genera varios problemas operativos. En primer lugar, retrasa la apertura del local, lo que ocasiona pérdida de ventas en las primeras horas del día, que son críticas para clientes que necesitan repuestos antes de iniciar sus jornadas laborales. Además, cuando el personal de atención no está completo, se genera sobrecarga en los empleados presentes, lo que reduce la calidad del servicio, aumenta los tiempos de espera y puede derivar en quejas de clientes. También afecta la organización del inventario y la recepción de proveedores que llegan en horarios programados.
4	¿Han implementado anteriormente algún tipo de sistema digital o tecnológico en la empresa? ¿Cómo fue esa experiencia?	No se ha implementado ningún sistema digital en la empresa
5	¿Qué características o funcionalidades consideran más importantes para un sistema digital de control de asistencia?	Necesitamos que permita registrar la entrada y salida de forma rápida, idealmente desde dispositivos móviles que los empleados ya poseen. También es fundamental que genere reportes automáticos que nos permitan visualizar la puntualidad, ausencias y horas trabajadas de manera clara. La geolocalización sería útil para verificar que el registro se realice desde las instalaciones de la empresa. Finalmente, el sistema debe ser seguro y confiable, evitando manipulaciones o registros fraudulentos.
6	¿Qué empleados creen que podrían tener más dificultades para adaptarse al nuevo sistema? ¿Por qué?	Principalmente los empleados de mayor edad o aquellos con menos experiencia en el uso de tecnología podrían presentar ciertas dificultades iniciales. Tenemos dos colaboradores mayores de 50 años que utilizan sus teléfonos de manera muy básica. Sin embargo, también tenemos experiencia previa que nos indica que, con capacitación personalizada y acompañamiento constante durante las primeras semanas, logran adaptarse satisfactoriamente.
7	¿Qué recursos técnicos tienen actualmente disponibles (computadoras,	Contamos con 3 computadora de escritorio a nivel de toda la empresa internet local. Todos los empleados poseen teléfono celular. No disponemos de servidores propios, por lo que cualquier solución tecnológica tendría que funcionar en la nube o mediante servicios externos de hosting.

	internet, servidores)?	
8	¿Estarían dispuestos a participar activamente en las pruebas piloto del sistema?	Si, consideramos fundamental participar activamente en todas las fases del proyecto, especialmente en las pruebas piloto. Estamos dispuestos a designar a un grupo de empleados representativos de las diferentes áreas para que prueben el sistema y brinden retroalimentación honesta sobre su funcionamiento. También nos comprometemos a facilitar el acceso a las instalaciones, proporcionar la información necesaria y colaborar en la identificación de mejoras o ajustes que requiera el sistema antes de su implementación definitiva. Entendemos que nuestra participación es clave para el éxito del proyecto.

Resultados de la observación directa

FICHA DE OBSERVACIÓN

Observador: Samantha Alexandra Caicedo Coca

Lugar y fecha: Autorepuestos W.M.C.C., Ambato – Ecuador, 18 de septiembre de 2025

Aspectos para observar

Nº	Aspecto observado	Descripción detallada	Frecuencia /Nivel
1	Proceso de registro de entrada del personal	Se observa que los empleados llegan y buscan el cuaderno de asistencia ubicado en el escritorio de recepción. Algunos empleados firman inmediatamente, mientras que otros ingresan directamente a sus áreas de trabajo y regresan minutos después a registrar su entrada. Se identificaron 3 empleados que olvidaron firmar durante la mañana y lo hicieron hasta el mediodía cuando fueron notificados. El cuaderno presenta firmas poco legibles en aproximadamente el 40% de los registros.	Constante durante todo el día
2	Ubicación y accesibilidad del registro	La carpeta de asistencia está colocada en un escritorio sin supervisión directa. Durante las horas pico (08:00-08:30 a.m.), se genera congestión porque varios empleados intentan firmar simultáneamente. No existe un espacio designado específicamente para este	Alta congestión en horas pico

		proceso, lo que causa interrupciones en la recepción.	
3	Supervisión del cumplimiento de horarios	No se observa una persona designada exclusivamente para supervisar el registro de asistencia. La secretaria ocasionalmente verifica quién ha firmado, pero debe atender otras responsabilidades simultáneamente (teléfono, atención a clientes, mensajería). Durante la observación, no se aplicó ninguna consecuencia inmediata por llegadas tardías.	Supervisión intermitente e informal
4	Horarios de entrada del personal	El horario establecido es 08:00 a.m., sin embargo, se registraron 5 llegadas tardías entre las 08:10 y 08:35 a.m. Los empleados que llegaron tarde firmaron sin justificación aparente ni llamado de atención. La hora de entrada promedio real es aproximadamente 08:12 a.m., lo que indica una cultura de permisividad respecto a la puntualidad.	5 de 14 empleados llegaron tarde (36%)
5	Registro de salida del personal	El proceso de salida presenta mayores irregularidades que el de entrada. Se observó que 4 empleados se retiraron sin firmar su salida. Otros 3 empleados firmaron su salida, pero permanecieron en las instalaciones por 15-20 minutos adicionales atendiendo asuntos personales. No existe un mecanismo para verificar la hora real de salida versus la hora registrada.	Irregular, aproximadamente 50% de incumplimiento
6	Impacto de la impuntualidad en operaciones	A las 08:05 a.m. llegó un cliente solicitando un repuesto urgente, pero el empleado responsable de esa área no había llegado aún. La recepcionista intentó ayudar, pero no pudo localizar el producto específico, generando espera de 20 minutos hasta que llegó el empleado. Esto evidencia cómo la impuntualidad impacta directamente en la atención al cliente y potenciales pérdidas de ventas.	2 casos observados durante el día

7	Actitud del personal hacia el control actual	Se percibe una actitud relajada y poco comprometida con el registro de asistencia. Varios empleados lo realizan de manera mecánica, sin prestar atención a la hora exacta que anotan. Durante conversaciones informales, 3 empleados mencionaron que "nadie revisa realmente las asistencias" y que "no pasa nada si llegas unos minutos tarde". Existe desconocimiento sobre posibles consecuencias del incumplimiento.	Actitud generalizada de permisividad
8	Disponibilidad tecnológica del personal	Durante los recesos y tiempos de descanso, el 100% de los empleados observados utilizan teléfonos inteligentes. Se observó que manejan aplicaciones como WhatsApp, redes sociales y aplicaciones bancarias con familiaridad. Esto indica que existe capacidad tecnológica básica para adoptar una aplicación móvil de control de asistencia.	Todos los empleados tienen smartphones
9	Organización de inicio de jornada	La apertura del local se realiza a las 08:00 a.m., pero la actividad productiva real inicia aproximadamente a las 08:20 a.m. cuando ya está presente la mayoría del personal. Se observa desorganización en la asignación de tareas matutinas, con empleados consultando qué deben hacer o esperando instrucciones. Esta demora inicial reduce la productividad en las primeras horas del día.	Retraso promedio de 20 minutos
10	Manipulación del registro	Se observó un caso donde un empleado que llegó a las 08:25 a.m. anotó como hora de entrada las 08:00 a.m. en el cuaderno. No existe ningún mecanismo de verificación que impida este tipo de alteraciones. El registro manual permite que los empleados anoten cualquier hora sin validación cruzada.	Al menos 1 caso confirmado

a. Análisis e interpretación de resultados

La metodología aplicada mediante entrevista, observación directa y encuesta permitió obtener una comprensión integral de la problemática del control de asistencia en la empresa Autorepuestos W.M.C.C., identificando convergencias significativas que fundamentan la necesidad del presente trabajo de investigación.

Análisis ficha de observación

Durante la observación de una jornada laboral completa se identificaron diversas deficiencias en el control de asistencia del personal. Se registró que el 36% de los empleados presentó retrasos entre 10 y 35 minutos, con un promedio de ingreso real a las 08:12 a.m. Además, se evidenciaron irregularidades en los registros de salida, como omisiones o anotaciones anticipadas, e incluso un caso de manipulación deliberada. La supervisión se realiza de manera intermitente, ya que la recepcionista cumple varias funciones, lo que limita el control efectivo. Estas fallas generan pérdidas de tiempo y afectan la atención al cliente. No obstante, se constató que todo el personal dispone de teléfonos inteligentes y maneja aplicaciones móviles, lo cual demuestra la viabilidad técnica para una solución digital.

Análisis entrevista al gerente general

De acuerdo con la entrevista al gerente general, el sistema manual actual demanda demasiado tiempo y es propenso a errores. La impuntualidad repercute en las ventas, la organización interna y la atención a los clientes. Por ello, se plantea implementar una aplicación móvil que automatice el registro de asistencia, genere reportes en tiempo real, utilice geolocalización y garantice la veracidad de los datos. El gerente expresó su disposición para participar en las pruebas piloto y apoyar la implementación.

Análisis encuesta al personal

Según la encuesta aplicada a la totalidad del personal (14 personas), tres preguntas resultan particularmente significativas. Respecto al control riguroso de horarios (pregunta 4), ningún empleado considera que siempre existe control estricto, solo el 7% manifestó que casi siempre, el 43% indicó que a veces, el 29% rara vez y el 21%

nunca. Esto evidencia que el 93% percibe deficiencias significativas en la supervisión horaria. En cuanto a la disposición para utilizar una aplicación móvil (pregunta 7), el 50% manifestó estar totalmente de acuerdo, el 36% de acuerdo y solo el 14% respondió a veces, sin respuestas negativas. El 86% presenta disposición clara para adoptar la solución tecnológica. Finalmente, respecto a la eficiencia del nuevo sistema (pregunta 10), el 57% está totalmente de acuerdo y el 43% de acuerdo en que la empresa será más eficiente, logrando 100% de respuestas afirmativas.

En conclusión, los tres instrumentos de recolección de datos proporcionan evidencia empírica sólida, convergente y complementaria que justifica plenamente el desarrollo e implementación de la aplicación móvil propuesta en esta investigación. La entrevista aportó la perspectiva estratégica y los requisitos funcionales desde la dirección de la empresa, la observación directa documentó objetivamente las deficiencias operativas y patrones comportamentales del sistema actual, y la encuesta validó la percepción generalizada de ineficiencia del control manual junto con la alta disposición del personal hacia la adopción tecnológica. Los tres instrumentos coinciden en señalar: las graves deficiencias estructurales del sistema manual vigente (errores, pérdidas de información, manipulación), el impacto operativo negativo en la productividad y atención al cliente, la necesidad imperiosa de automatización mediante soluciones tecnológicas, y la existencia de condiciones organizacionales favorables tanto en infraestructura tecnológica (100% del personal con smartphones) como en disposición actitudinal (86% dispuesto a usar la aplicación, 100% confía en su efectividad) para adoptar exitosamente la solución propuesta. Esta convergencia metodológica fortalece la validez interna de la investigación y proporciona fundamentos empíricos robustos para proceder con las siguientes fases del proyecto de desarrollo tecnológico.

CAPÍTULO III. RESULTADOS Y DISCUSIÓN

En el presente capítulo se presentan los principales resultados obtenidos durante el desarrollo de la aplicación móvil basada en Flutter y ASP.NET Core para el control de cumplimiento laboral en la empresa Autorepuestos W.M.C.C. A partir de la metodología planteada en el capítulo anterior, se presentan los hallazgos derivados del análisis de datos, el proceso de diseño e implementación del sistema y la validación funcional de sus componentes.

3.1 Análisis del nivel de cumplimiento laboral

El cumplimiento laboral constituye un eje fundamental para el correcto funcionamiento de cualquier organización. En el contexto de Autorepuestos W.M.C.C., empresa dedicada a la comercialización de repuestos automotrices en la ciudad de Ambato, el control efectivo de la asistencia y puntualidad del personal representa un factor crítico para garantizar la continuidad operativa y la satisfacción de los clientes que requieren atención inmediata.

La presente sección analiza el nivel de cumplimiento laboral actual de los 14 empleados de la empresa, utilizando datos cuantitativos obtenidos mediante tres instrumentos metodológicos complementarios: encuestas estructuradas aplicadas al 100% del personal, entrevista técnica con el Gerente General, y observación directa del proceso de registro de asistencia durante una jornada laboral completa. Este enfoque metodológico permite obtener una visión integral y objetiva de la situación real, identificando tanto patrones comportamentales como deficiencias estructurales del sistema de control manual vigente.

Tabla 12. Métricas del cumplimiento laboral

Indicador	Valor registrado	Valor esperado	Nivel de cumplimiento
Puntualidad en entrada	64% (9/14 empleados)	100%	BAJO
Registro completo de entrada	79% (11/14 empleados)	100%	MEDIO
Registro completo de salida	71% (10/14 empleados)	100%	MEDIO-BAJO

Promedio de retraso	12 minutos	0 minutos	DEFICIENTE
Manipulación de registros	7% (1 caso confirmado)	0%	CRÍTICO

De acuerdo con los indicadores de la Tabla 12 revelan que únicamente el 64% del personal cumple con el horario establecido de entrada a las 08:00 a.m., lo que significa que en promedio 5 de cada 14 empleados llegan tarde diariamente. El retraso promedio de 12 minutos genera un inicio efectivo de operaciones a las 08:12 a.m., impactando negativamente en la atención temprana a clientes. La tasa de registro incompleto (21% en entrada y 29% en salida) evidencia falta de disciplina y ausencia de mecanismos de verificación obligatoria.

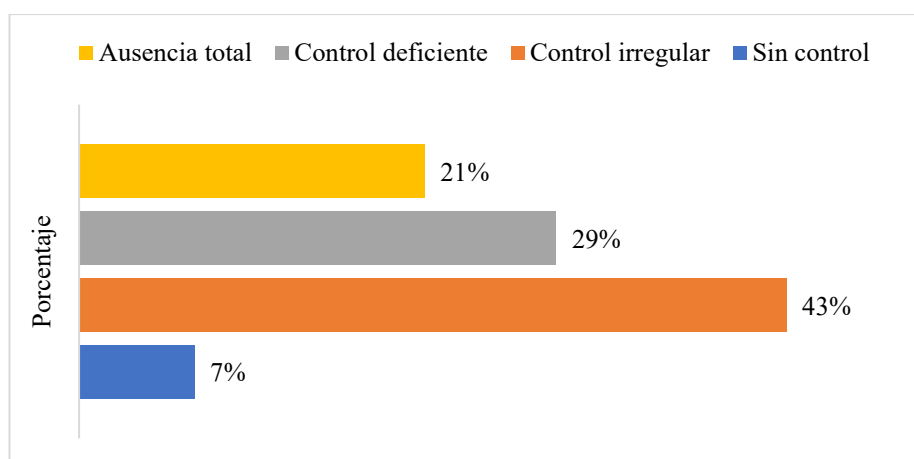


Gráfico 11. Distribución de conocimiento del control de horarios

De acuerdo con el Gráfico 11 el dato más revelador es que ningún empleado considera que siempre existe un control riguroso de horarios. El 93% del personal observa deficiencias significativas o totales en la supervisión, lo que ha generado un ambiente laboral sin control donde no hay consecuencias inmediatas por el incumplimiento.

Tabla 13. Categorización del personal según cumplimiento

Categoría	Nº empleados	Porcentaje	Características
Cumplimiento excelente	2	14%	Siempre puntual, registra correctamente
Cumplimiento bueno	7	50%	Ocasionalmente tarde (≤ 5 min), registra

Cumplimiento regular	3	21%	Frecuentemente tarde (6-15 min)
Cumplimiento deficiente	2	14%	Constantemente tarde (>15 min), omite registros
TOTAL	14	100%	

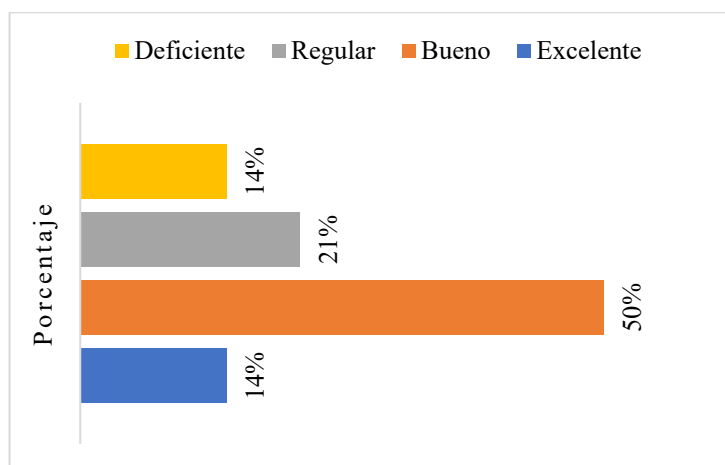


Gráfico 12. Distribución del personal por nivel de cumplimiento

La Tabla 13 muestra que solo el 14% del personal mantiene un cumplimiento excelente, mientras que el 50% se ubica en la categoría "bueno" con incumplimientos ocasionales menores. El 35% restante (regular + deficiente) representa un grupo problemático que requiere intervención inmediata mediante controles más estrictos. Esta distribución sugiere que existen dos culturas laborales diferenciadas: una minoría comprometida y una mayoría que se adapta a la permisividad del sistema actual.

3.1.1 Identificación de falencias en los métodos de control existentes.

Del análisis realizado mediante observación directa, entrevista al gerente y encuesta al personal, se evidencia que el sistema manual de control de asistencia utilizado por la empresa Autorepuestos W.M.C.C. presenta fallas estructurales que afectan la puntualidad, la veracidad de los registros y la eficiencia operativa. Estas debilidades comprometen tanto la gestión interna como la atención al cliente y justifican la necesidad de una solución tecnológica más confiable y automatizada. A continuación, se resumen las principales falencias identificadas.

- Manipulación de horarios registrados: Se evidenció que empleados anotan una hora distinta a la de llegada real, sin ningún mecanismo que evite modificaciones.
- Falta total de verificación o validación de los datos registrados: El sistema manual permite anotar cualquier hora sin validación cruzada y sin supervisión continua.
- Registros incompletos o inexistentes de salida: Se observó que varios empleados olvidan firmar la salida o registran horarios que no coinciden con la hora real de retiro.
- Pérdida de productividad por retrasos frecuentes: El promedio real de entrada es 08:12 a.m., lo que retrasa el inicio de actividades y afecta la atención al cliente.

3.1.2 Impacto en la Organización

El cumplimiento inadecuado del horario laboral crea un círculo vicioso en el que la falta de controles efectivos perpetúa el incumplimiento, afectando al servicio al cliente y la integridad interna. Los empleados comprometidos (14% con cumplimiento excelente) mostrados en el Gráfico 12 perciben injusticia porque no hay diferencia entre los que cumplen y los que no, lo que puede llevar a la desmotivación y aumentar la rotación de empleados de calidad.

3.1.3 Propuesta de control de asistencia

Considerando las problemáticas identificadas en el sistema actual de registro de asistencia, donde se evidenció manipulación de registros, pérdida de información y ausencia de personal para verificación, se propone que el registro de asistencia se realice mediante la implementación de una aplicación móvil con validación de georeferenciación como solución tecnológica integral con el fin de evitar que el personal realice sus registros desde áreas externas a la empresa.

3.2 Análisis de tecnologías y herramientas

Se evaluaron diferentes alternativas tecnológicas para seleccionar las que se adaptaran mejor al entorno de la empresa, garantizando rendimiento, seguridad y facilidad de mantenimiento.

Tabla 14. Comparativa de frameworks móvil

Criterio	Flutter	React Native	Ionic
Lenguaje	Dart	JavaScript	TypeScript
Desempeño	Alto (compila a código nativo)	Medio	Medio-bajo
Compatibilidad multiplataforma	Android, iOS, Web, Desktop	Android, iOS	Android, iOS, Web
Diseño de interfaz	Widgets nativos	Componentes híbridos	Basado en HTML/CSS
Integración con APIs externas	Simple	Requiere librerías	Moderada
Arquitectura interna	Usa Skia para renderizar todo, independiente del sistema	Usa el motor nativo del SO y puente JavaScript	Basado en WebView para renderizar
Mantenimiento a largo plazo	Respaldado por Google	Respaldado por Google	Respaldado por Google

Después de comparar los diferentes frameworks móviles que se muestran en la Tabla 14, se eligió Flutter por varias razones prácticas. Lo primero fue el rendimiento: Flutter compila directamente a código nativo, lo que significa que no necesita intermediarios para comunicarse con el sistema operativo como pasa con React Native. Esto se traduce en una aplicación más rápida y fluida, algo crítico cuando se está validando la ubicación GPS en tiempo real para el registro de asistencia.

Otro punto importante fue la compatibilidad multiplataforma. Con Flutter se puede crear la aplicación una sola vez y funciona tanto en Android como en iOS, pero además en web y escritorio si en el futuro la empresa necesita expandirse a esas plataformas. React Native solo cubre móviles, mientras que Ionic, aunque es multiplataforma, tiene limitaciones de rendimiento porque trabaja con HTML/CSS.

La integración con Google Maps también fue clave en la decisión. Flutter tiene paquetes oficiales que simplifican bastante la implementación de geolocalización [4], [23], mientras que con React Native habría que configurar varias librerías externas y lidiar con más complejidad técnica. Considerando que la validación del perímetro de 5 metros es una función central del sistema, esta facilidad de integración fue determinante.

Por último, el respaldo de Google garantiza que Flutter seguirá recibiendo actualizaciones y soporte a largo plazo, lo cual es fundamental para un sistema que manejará datos sensibles de empleados durante años.

Tabla 15. Comparativa de frameworks backend

Criterio	ASP.NET Core	Node.js (Express)	Laravel (PHP)
Lenguaje	C#	JavaScript	PHP
Rendimiento	Alto	Medio	Medio-bajo
Seguridad	Elevada (JWT, HTTPS, Identity)	Media	Media
Escalabilidad	Alta	Alta	Media
Integración con SQL Server	Nativa	Parcial	Limitada

ASP.NET Core fue seleccionado como framework backend después de evaluar las opciones disponibles que muestra la Tabla 15. La decisión se basó principalmente en que ASP.NET Core ofrece integración nativa con la base de datos SQL Server [5], [22], lo que facilita la conexión y el manejo de datos sin necesidad de configuraciones complejas o librerías adicionales. En comparación, Node.js (Express) tiene integración parcial con SQL Server y Laravel (PHP) presenta limitaciones significativas que habrían complicado el desarrollo.

El rendimiento también fue un factor determinante. ASP.NET Core está clasificado como alto, superando tanto a Node.js con rendimiento medio como a Laravel con medio-bajo [22]. Esto es especialmente importante para un sistema de control de asistencia que debe procesar múltiples solicitudes simultáneas de registro, validación de ubicación y generación de reportes sin comprometer los tiempos de respuesta.

En cuanto a seguridad, ASP.NET Core ofrece un nivel elevado con herramientas integradas como JWT para autenticación, soporte para HTTPS y el sistema Identity para gestión de usuarios y roles [5], [22]. Estas características vienen incorporadas en el framework, mientras que con Node.js o Laravel habría que implementar estas medidas de seguridad manualmente o mediante paquetes externos, aumentando el riesgo de vulnerabilidades.

Finalmente, la escalabilidad de ASP.NET Core garantiza que, si la empresa crece o necesita expandir las funcionalidades del sistema en el futuro, la arquitectura puede adaptarse sin necesidad de reescribir componentes críticos [22]. Esta proyección a largo plazo, sumada a la compatibilidad natural con el ecosistema tecnológico existente en Autorepuestos W.M.C.C. [26], consolidó a ASP.NET Core como la opción más adecuada para el backend del proyecto.

3.3 Flutter en el desarrollo de aplicaciones móviles

Flutter (Figura 1) es un framework de código abierto desarrollado por Google que permite construir aplicaciones nativas para Android, iOS, web y escritorio a partir de una sola base de código [25]. Su arquitectura se distingue por utilizar un motor gráfico propio basado en Skia que renderiza directamente los componentes visuales sin depender de widgets nativos, lo que garantiza un rendimiento consistente y cercano al nativo en todas las plataformas [6], [30].



Figura 1. Logo de Flutter

A diferencia de React Native, que utiliza un puente de JavaScript para comunicarse con componentes nativos, Flutter compila directamente a código ARM nativo,

eliminando la sobrecarga de comunicación entre capas y resultando en un desempeño superior [31], [35]. Esta característica es particularmente importante para aplicaciones que requieren interacciones frecuentes con el sistema, como el caso del registro de asistencia con geolocalización en tiempo real.

3.3.1 Historia y evolución de Flutter

Su historia comienza en 2014 cuando un equipo de ingenieros de Google inició el desarrollo de un proyecto experimental denominado "Sky" con el objetivo de crear aplicaciones que pudieran ejecutarse a 120 fotogramas por segundo de manera consistente [25], [30].

El proyecto Sky representaba un cambio paradigmático al proponer un motor de renderizado propio que compilaría directamente a código ARM nativo, en lugar de depender de WebViews o puentes de comunicación entre JavaScript y código nativo [37]. En mayo de 2017, durante la conferencia Google I/O, el proyecto fue renombrado oficialmente como "Flutter" y se lanzó su primera versión Alpha, enfocada exclusivamente en Android y utilizando el lenguaje Dart [25], [41].

La evolución continuó con la versión Beta en febrero de 2018, incorporando soporte oficial para iOS y demostrando la viabilidad técnica del enfoque multiplataforma [30]. El 4 de diciembre de 2018, Google anunció Flutter 1.0, la primera versión estable lista para producción, validada por empresas como Alibaba, Tencent y Abbey Road Studios [25], [36].

Flutter 1.12 (diciembre 2019) expandió el alcance al introducir soporte experimental para web y desktop, respondiendo a la demanda por soluciones verdaderamente multiplataforma [25], [37]. Flutter 2.0 (marzo 2021) introdujo "sound null safety" en Dart y estableció soporte estable para aplicaciones web, acercándose a la visión multiplataforma completa [30], [40].

Flutter 3.0, presentado en mayo de 2022, consolidó el soporte estable para las seis plataformas principales: Android, iOS, Web, Windows, macOS y Linux, estableciendo a Flutter como el único framework con soporte de producción genuino para todas las plataformas desde una única base de código [25], [34]. Las versiones posteriores han

continuado refinando el framework, con Flutter 3.24 (agosto 2024) implementando mejoras de rendimiento del 20-30% en operaciones gráficas [37].

Según datos de Google publicados en 2024, más de 1 millón de aplicaciones en Google Play Store y Apple App Store han sido desarrolladas con Flutter, adoptado por empresas como Alibaba, eBay, Nubank, Revolut, BMW y Toyota [25], [30], [36]. La comunidad ha crecido proporcionalmente, con más de 150,000 paquetes publicados en pub.dev y contribuciones de desarrolladores en más de 170 países [25], [41].



Figura 2. Versiones de Flutter

La línea de tiempo de la Figura 2 muestra cómo Flutter pasó de ser un proyecto experimental móvil a un framework multiplataforma con soporte de producción para seis sistemas operativos principales.

3.3.2 Requisitos del sistema para desarrollar con Flutter

La siguiente tabla (Tabla 16), desglosa los requisitos importantes para el correcto desarrollo en Flutter de acuerdo a las necesidades presentadas o al tipo de utilización de la persona interesada.

Tabla 16. Requisitos

Componente	Windows	macOS	Linux
Sistema operativo	Windows 10/11 (64-bit)	macOS 10.14 o superior	Ubuntu 20.04 LTS o superior
Espacio en disco	2.5 GB (sin IDE)	2.8 GB (sin IDE)	1.5 GB (sin IDE)
Herramientas requeridas	Git para Windows, PowerShell 5.0+	Xcode (para iOS), CocoaPods	Git, curl, unzip, xz-utils
RAM mínima	8 GB	8 GB	8 GB
RAM recomendada	16 GB	16 GB	16 GB
Android Studio/VS Code	Versión 2022.1+	Versión 2022.1+	Versión 2022.1+
SDK de Android	API level 21+ (Android 5.0)	API level 21+ (Android 5.0)	API level 21+ (Android 5.0)
Compilación iOS	No disponible	Xcode 15.0+ obligatorio	No disponible

3.3.3 Lenguaje de Programación Dart

Flutter utiliza el lenguaje de programación Dart, también desarrollado por Google, que ofrece múltiples ventajas para el desarrollo móvil. Dart soporta tanto compilación AOT (Ahead-of-Time) para producir código nativo optimizado, como compilación JIT (Just-in-Time) que permite la función de Hot Reload durante el desarrollo [25], [41]. Esta capacidad de Hot Reload es fundamental para la productividad del equipo de desarrollo, ya que permite visualizar los cambios en tiempo real sin reiniciar la aplicación ni perder el estado actual, acelerando significativamente el ciclo de desarrollo y pruebas [29].

3.3.4 Capacidades de Integración con Servicios Externos

Una de las ventajas decisivas de Flutter es su capacidad de integración simplificada con APIs externas, particularmente con Google Maps API, componente fundamental para el sistema de validación geográfica del proyecto [23]. Flutter proporciona paquetes oficiales y bien mantenidos para la integración con servicios de Google, lo

que reduce significativamente el tiempo de desarrollo y garantiza compatibilidad a largo plazo [25], [37].

La investigación de Saxena et al. demostró la versatilidad de Flutter en la gestión de recursos tecnológicos mediante el desarrollo de una aplicación multiplataforma para el control remoto de recursos Docker, utilizando protocolos seguros como SSH [9]. Esta capacidad de integración segura con sistemas externos valida la elección de Flutter para aplicaciones empresariales que requieren comunicación con múltiples servicios, como es el caso del sistema de control de asistencia que debe integrarse con APIs de geolocalización, bases de datos en la nube y servicios de autenticación.

3.3.5 Rendimiento en Aplicaciones de Geolocalización

El alto rendimiento de Flutter es particularmente relevante para aplicaciones que hacen uso intensivo de servicios de ubicación. Zefrinaldi desarrollaron una aplicación de scouting de fútbol basada en Flutter que permitió registrar y buscar información en tiempo real, evidenciando cómo Flutter puede ser utilizado eficientemente para desarrollar aplicaciones que requieren procesamiento continuo de datos geoespaciales [7].

En el contexto del sistema de control de asistencia, esta capacidad permite:

- Actualización fluida de la ubicación del empleado en tiempo real
- Respuesta inmediata al verificar si el usuario se encuentra dentro del perímetro permitido
- Procesamiento eficiente de múltiples solicitudes simultáneas de validación de ubicación [32]
- Consumo optimizado de batería durante el seguimiento de ubicación [33]

3.3.6 Soporte Multiplataforma Real



Figura 3. Flutter multiplataforma

La compatibilidad multiplataforma (Figura 3) de Flutter va más allá de iOS y Android, extendiéndose a web y desktop (Windows, macOS, Linux) [4], [25]. Esta característica proporciona flexibilidad futura para expandir el sistema de control de asistencia a una versión web administrativa o de escritorio sin necesidad de reescribir completamente el código.

Tabla 17. Tabla multiplataforma

Móvil	Flutter permite crear experiencias personalizadas para usuarios de iOS y Android sin concesiones y con menos recursos.
Web	La web es una plataforma flexible, pero Flutter es ideal para crear aplicaciones web como PWA o SPA y llevar su aplicación móvil existente a la web.
De Escritorio	Tanto la experiencia móvil como web a dispositivos de escritorio desde la misma base de código para llegar a usuarios que manejan una aplicación de escritorio
Incorporado	La compatibilidad que tiene flutter con integradores personalizados significa que se puede crear nuevas formas de hacer que Flutter funcione en plataformas de interés.

Como se muestra en la Tabla 17, su arquitectura moderna y su capacidad para compilar a código nativo lo convierten en una herramienta eficiente y flexible para el desarrollo móvil y multiplataforma [25], [34]. Esta escalabilidad asegura que la inversión en desarrollo con Flutter no se limita a la solución móvil actual, sino que puede evolucionar según las necesidades futuras de la empresa.

3.4 Análisis de metodologías de desarrollo ágil

Durante la etapa de planificación del proyecto se analizó la conveniencia de aplicar una metodología ágil que permita una gestión flexible, visual y adaptativa del desarrollo del sistema. Las metodologías evaluadas fueron Scrum, Kanban y Extreme Programming (XP), las cuales se comparan en la siguiente tabla (Tabla 18).

Tabla 18. Comparativa de Metodologías de investigación

Criterio	Scrum	Kanban	Extreme Programming (XP)
Estructura de trabajo	Dividido en sprints o iteraciones de tiempo fijo.	Flujo continuo de tareas sin iteraciones definidas.	Ciclos cortos con programación en pareja.
Roles definidos	Product Owner, Scrum Master, Equipo de desarrollo.	No establece roles fijos, el equipo se autogestiona.	Desarrolladores, cliente, coach, tracker.
Planificación	Se planifica al inicio de cada sprint (1-4 semanas).	Se ajusta de forma continua según prioridad.	Planificación diaria basada en retroalimentación.
Gestión visual	Tablero Scrum con historias de usuario.	Tablero Kanban con columnas de flujo de trabajo.	Tablero simple de tareas.
Métricas principales	Velocidad del equipo, burndown chart.	Tiempo de ciclo y flujo de tareas.	Calidad del código y número de entregas.
Ventajas	Control del progreso, definición de roles claros.	Flujo constante, transparencia y reducción de tiempos muertos.	Mejora continua del código y del trabajo en equipo.
Desventajas	Requiere mayor disciplina en roles y tiempos.	Puede generar saturación si no se limita el trabajo en curso.	Menor formalidad y documentación.
Adecuación al proyecto	Media (proyecto corto con pocos miembros).	Alta (flujo continuo, tareas visuales, adaptable a cambios).	Media-baja (no se requiere programación en pareja).

3.4.1 Metodología seleccionada

Luego del análisis comparativo, se seleccionó la metodología Kanban como la más adecuada para el desarrollo de la aplicación móvil, porque combina la flexibilidad necesaria para un proyecto de investigación-desarrollo con la estructura visual que se

necesita para gestionar múltiples tecnologías complejas en un plazo definido, mientras se mantiene una comunicación fluida con la empresa beneficiaria.

El uso de Kanban permite visualizar todas las tareas en proceso, identificar cuellos de botella y optimizar el tiempo de desarrollo. Esta metodología se basa en un tablero compuesto por columnas que representan los estados de avance del trabajo: Por hacer, En progreso, En revisión y Finalizado.

Ventajas Específicas

- Kanban demuestra ser especialmente útil para manejar requisitos que van evolucionando durante el desarrollo.
- La validación continua con los usuarios finales es otra ventaja decisiva. Con Kanban es posible mostrar regularmente el progreso al gerente y al personal de Autorepuestos W.M.C.C.
- El tablero Kanban en ClickUp permite visualizar en todo momento qué tareas están pendientes, cuáles en desarrollo y cuáles completadas.

3.5 Desarrollo de la propuesta

Para el desarrollo del sistema se empleó la metodología Kanban se organizó un flujo de trabajo dividido en etapas (Tablas 19), cada una de ellas garantizan la correcta planificación, implementación y verificación del sistema de control de asistencia.

Tabla 19. Etapas de la metodología Kanban

ETAPA	DESCRIPCIÓN GENERAL	ACCIONES ESPECÍFICAS DEL PROYECTO
1. Requerimientos	Definir funcionalidades, módulos y objetivos del sistema de control laboral.	• Reuniones con gerente de Autorepuestos W.M.C.C. • Levantamiento del proceso manual actual • Identificación de roles (empleado – administrador)
2. Definir trabajo	Configurar el tablero Kanban en Clickup con columnas y reglas del flujo.	• Columnas: Backlog, Por hacer, En progreso, Bloqueado, En revisión, Finalizado • Asignación de responsables

3. Visualizar fases	Dividir el proyecto en fases que representen áreas funcionales.	• Fase Frontend Flutter • Fase Backend ASP.NET Core • Fase Base de Datos SQL Server • Fase pruebas funcionales e integración
4. Desarrollo	Implementación del código del sistema según las fases definidas.	• Diseño de la arquitectura del sistema • Creación de base de Datos en SQL Server • Creación del servidor REST en ASP.NET Core. • Construcción de la estructura base del proyecto en Flutter. • Construcción de pantallas del aplicativo móvil.
5. Pruebas	Garantizar que el sistema funcione correctamente en todas sus funciones.	• Pruebas funcionales • Pruebas de seguridad • Pruebas de geolocalización
6. Finalizar antes de empezar (límites wip)	Controlar el número de tareas activas para evitar saturación del flujo.	• Aplicar regla “Stop starting, start finishing” • Máximo 3 tareas en En Progreso
7. Despliegue	Publicación del sistema para uso interno en la empresa.	• Despliegue del backend • Despliegue del Frontend

3.5.1 Requerimientos

En el transcurso de esta etapa se define un conjunto de tareas claves para una correcta planificación. Para el desarrollo de la aplicación móvil con Flutter y ASP.NET Core destinada al control de cumplimiento laboral en la empresa Autorepuestos W.M.C.C., se realizó un proceso de levantamiento de requerimientos funcionales y no funcionales evidenciados en la Tabla 20 y Tabla 21 con el personal administrativo y operativo. Esta fase permitió determinar las necesidades tecnológicas que debía cubrir el sistema, así como los criterios de eficiencia, seguridad y usabilidad esperados.

Tabla 20. Requerimientos funcionales

Requerimientos Funcionales	Solicita
Permitir el registro y autenticación de empleados mediante usuario y contraseña.	Gerente
Registrar la asistencia diaria con validación de ubicación geográfica.	Gerente
Visualizar el historial de asistencias y reportes en tiempo real.	Talento Humano

Permitir al administrador gestionar usuarios, horarios y ubicaciones.	Gerente
Integrar notificaciones automáticas ante llegadas tardías o salidas anticipadas.	Talento Humano

Tabla 21. Requerimientos No funcionales

Requerimientos No funcionales
La aplicación debe ser multiplataforma (Android e iOS).
El backend debe estar desarrollado en ASP.NET Core, conectado a una base de datos SQL Server.
Se requiere disponibilidad del 99%, almacenamiento seguro de datos y tiempos de respuesta inferiores a 3 segundos por solicitud.
Interfaz intuitiva y adaptable a diferentes tamaños de pantalla.

Esta información evidenció la necesidad de automatizar los procesos de control de asistencia mediante una herramienta que registre la ubicación real del empleado al momento de su ingreso o salida, generando trazabilidad y confiabilidad en la gestión laboral como se observa la Tabla 22.

Tabla 22. Etapa 1 Requerimientos

Tarea	Descripción	Estado
Levantamiento de requerimientos	Identificación de necesidades de la empresa Autorepuestos W.M.C.C.	Completado
Roles del sistema	Definir administrador, empleado e invitado/colaborador.	Completado
Proceso actual	Descripción del flujo manual actual.	Completado
Validación	Aprobación de requerimientos	Completado

3.5.2 Definición del trabajo

En esta etapa como se observa en la Figura 4 se establece el cronograma con las tareas correspondientes, desde su inicio hasta la finalización

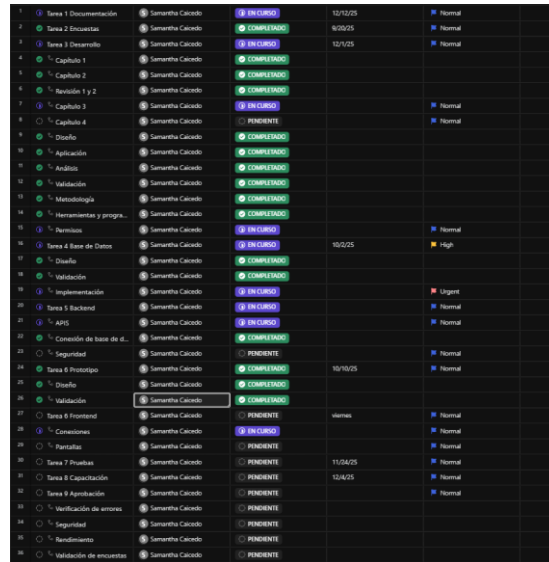


Figura 4. Flujo de trabajo en ClickUp

3.5.3 Visualizar fases

Se muestra el progreso del proyecto (Figura 5), para tener coordinación y comunicación con los interesados. Las tareas se muestran mediante tarjetas gráficas el cual nos permite una clara comprensión y nos muestra el estado real del proyecto.

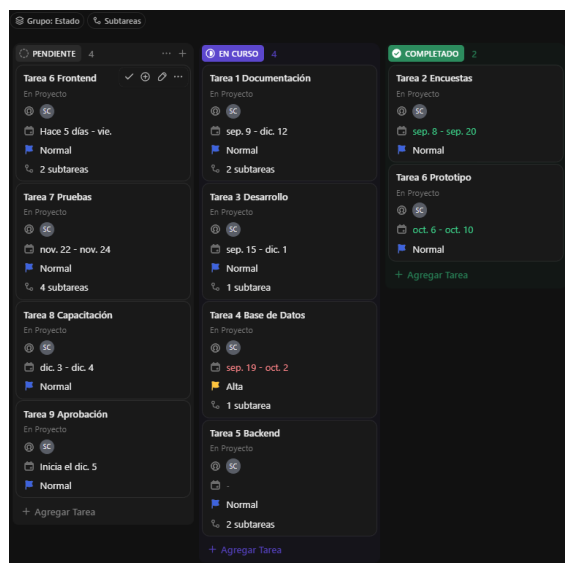


Figura 5. Tareas del desarrollo del proyecto

3.5.4 Desarrollo

Durante la etapa de desarrollo se llevaron a cabo todas las actividades relacionadas con la implementación del sistema como se observa en la Tabla 23. Esta etapa se ejecutó de manera incremental, tomando las tareas desde *Pendiente* hacia *En curso*, y de allí a *Completado*. Las actividades más importantes incluyeron:

- Diseño de la arquitectura del sistema
- Creación de base de Datos en SQL Server
- Creación del servidor REST en ASP.NET Core.
- Construcción de la estructura base del proyecto en Flutter.
- Construcción de pantallas del aplicativo móvil.

Tabla 23. Etapa 4 Desarrollo

Tarea	Descripción	Estado
Diseño de la arquitectura del sistema	Definición de la estructura en tres capas y componentes principales.	Completado
Creación de base de Datos en SQL Server	Diseño e implementación de tablas, relaciones y procedimientos almacenados.	Completado
Creación del servidor REST en ASP.NET Core.	Desarrollo de endpoints para autenticación, asistencia y gestión de usuarios.	Completado
Construcción de la estructura base del proyecto en Flutter.	Organización de carpetas, configuración de providers y servicios.	Completado
Construcción de pantallas del aplicativo móvil.	Implementación de interfaces de iniciar sesión, registro, menús y validaciones.	Completado

a. Diseño de la arquitectura del sistema

La aplicación se diseñó bajo una arquitectura de tres capas, que mejora la organización y la mantenibilidad del código como se muestra en la Figura 6:

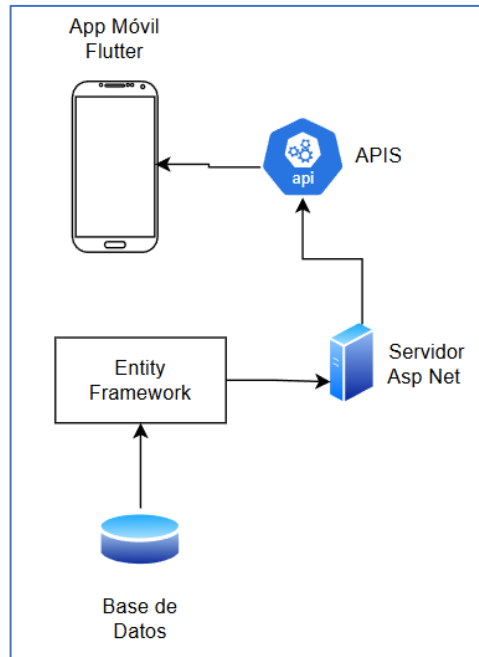


Figura 6. Arquitectura del sistema

- **Capas del sistema**

- **Datos:** Esta capa es importante para el almacenamiento de usuarios, asistencias, horarios. SQL Server.
- **Lógica de negocio:** esta permite un procesamiento de datos, autenticación, validación geográfica. ASP.NET Core, C#.
- **Presentación:** La función principal es su interfaz gráfica e interacción con el usuario. Flutter (Dart).

- **Componentes del sistema**

La aplicación se estructura en cuatro módulos principales que trabajan de manera integrada, como se muestra en la Tabla 24.

Tabla 24. Componentes del sistema

Módulo de Autenticación	Módulo de Geolocalización	Módulo de Registro de Asistencia	Módulo de Reportes
Iniciar sesión con usuario y contraseña	Obtención de coordenadas GPS del dispositivo	Captura de entrada/salida	Historial de asistencias por empleado
Validación de credenciales	Cálculo de distancia con ubicación autorizada	Validación de horario permitido	Reportes de puntualidad

Generación de tokens JWT	Validación de perímetro (radio de 5 metros)	Registro de ubicación geográfica	Estadísticas de cumplimiento
Gestión de sesiones	Registro de ubicación en cada marcación	Control de cédula	Exportación de datos

b. Creación de base de Datos en SQL Server

La base de datos del sistema se diseñó bajo el modelo relacional utilizando SQL Server, considerando los principios de normalización para evitar redundancia de información y garantizar la integridad de los datos. El diseño contempla 7 tablas principales que se relacionan entre sí mediante claves foráneas estableciendo una estructura que permite el flujo correcto de información desde el registro de usuarios hasta la generación de reportes de asistencia.

- ***Diseño del modelo de datos***

El modelo entidad-relación diseñado (como se muestra en el diagrama de la base de datos) establece las siguientes entidades principales:

- **Tabla Usuarios:** Esta tabla constituye el centro del sistema, almacenando la información completa del personal de Autorepuestos W.M.C.C. Utiliza el número de cédula como clave primaria, garantizando que cada empleado tenga un identificador único. Los campos principales incluyen datos personales (nombre, apellido, email, teléfono), credenciales de acceso (contraseña hashada mediante algoritmos seguros), y campos de control como la imagen de perfil y el estado activo/inactivo del usuario.
- **Tabla Roles:** Define los niveles de acceso dentro del sistema. Se tiene una estructura simple pero funcional que incluye el identificador del rol (RoleId), el nombre del rol (RoleNombre) y una descripción detallada de los permisos asociados. Para el caso de Autorepuestos W.M.C.C., se establecieron tres roles principales: “Administrador”, con permisos completos para gestionar usuarios, horarios y reportes; “Empleado”, con permisos limitados al registro de su propia asistencia y por último “Invitado/colaborador”, que solo puede registrar mediante la validación de cédula.

- **Tabla Horarios:** Almacena la configuración de los distintos horarios laborales que pueden existir en la empresa. Esta tabla permite una gestión flexible de jornadas de trabajo, incluyendo campos como la hora de entrada (HoraEntrada), hora de salida (HoraSalida), tiempo de almuerzo (HoraInicioAlmuerzo y HoraFinAlmuerzo), y una tolerancia en minutos para llegadas tardías (ToleranciaMinutos, establecida por defecto en 7 minutos).
- **Tabla Registro:** Esta es la tabla central del sistema de control de asistencia, donde se almacenan todos los eventos de entrada y salida del personal. Cada registro incluye la fecha y hora exacta del check-in (CheckInTime) y check-out (CheckOutTime), junto con las coordenadas GPS correspondientes (CheckInLatitude, CheckInLongitude, CheckOutLatitude, CheckOutLongitude) que validan que el empleado se encuentra físicamente en las instalaciones de la empresa al momento del registro. La tabla incorpora campos para el control del tiempo de almuerzo (AlmuerzoInicio y AlmuerzoFin), el cálculo automático de horas trabajadas (HorasTrabaja), y el estado del registro (Estado), que puede ser "A tiempo", "Tardío", "Ausente", entre otros. El campo RegistroRapido permite diferenciar los registros realizados por colaboradores externos que no tienen cuenta completa en el sistema, característica que resultó fundamental para la flexibilidad operativa de la empresa.
- **Tabla UbicacionesPermitidas:** Define las coordenadas geográficas autorizadas para el registro de asistencia. Cada ubicación se caracteriza por un nombre descriptivo (LocationName), coordenadas de latitud y longitud, y un radio de tolerancia en metros (Radio) que determina el área válida para el registro. Para Autorepuestos W.M.C.C., se estableció un radio de 5 metros alrededor de las coordenadas exactas de la empresa, garantizando que solo los empleados presentes físicamente en las instalaciones puedan registrar su asistencia.
- **Tabla DiasFeriados:** Gestiona los días no laborables, incluyendo feriados nacionales, provinciales (específicos de Tungurahua) y locales (exclusivos de Ambato). Esta tabla permite al sistema ajustar automáticamente las validaciones de asistencia en fechas especiales, evitando que se registren faltas

en días donde no hay obligación de asistir. Los campos incluyen la fecha del feriado, nombre, descripción, alcance geográfico (nacional, provincial o local), y un estado activo/inactivo para su gestión.

- **Tabla Notificaciones:** Permite el envío de alertas personalizadas a los usuarios del sistema. Cada notificación incluye un título, mensaje, fecha de creación y un indicador de lectura (Leído). Esta funcionalidad se utiliza para informar al personal sobre llegadas tardías, recordatorios de registro de salida, o comunicados administrativos importantes. Las notificaciones se asocian directamente con cada usuario mediante su número de cédula, garantizando que cada persona reciba únicamente las alertas que le corresponden.

- **Relaciones entre tablas**

El diseño establece relaciones claras entre las entidades del sistema como se observa en la Figura 7:



Figura 7. Relaciones entre tablas

- La tabla Usuarios se relaciona con Roles mediante la clave foránea RoleId, estableciendo una relación de muchos a uno (muchos usuarios pueden tener el mismo rol).

- Usuarios también se relaciona con Horarios mediante HorarioId, permitiendo asignar diferentes horarios a distintos grupos de empleados.
- La tabla Registro se vincula con Usuarios a través del número de Cedula, creando una relación de uno a muchos (un usuario puede tener múltiples registros de asistencia).
- Notificaciones también se conecta con Usuarios mediante Cedula, permitiendo el envío personalizado de alertas.

c. Creación del servidor REST en ASP.NET Core.

El servicio (backend) está conformado por varias secciones como se muestra a continuación en la Figura 8

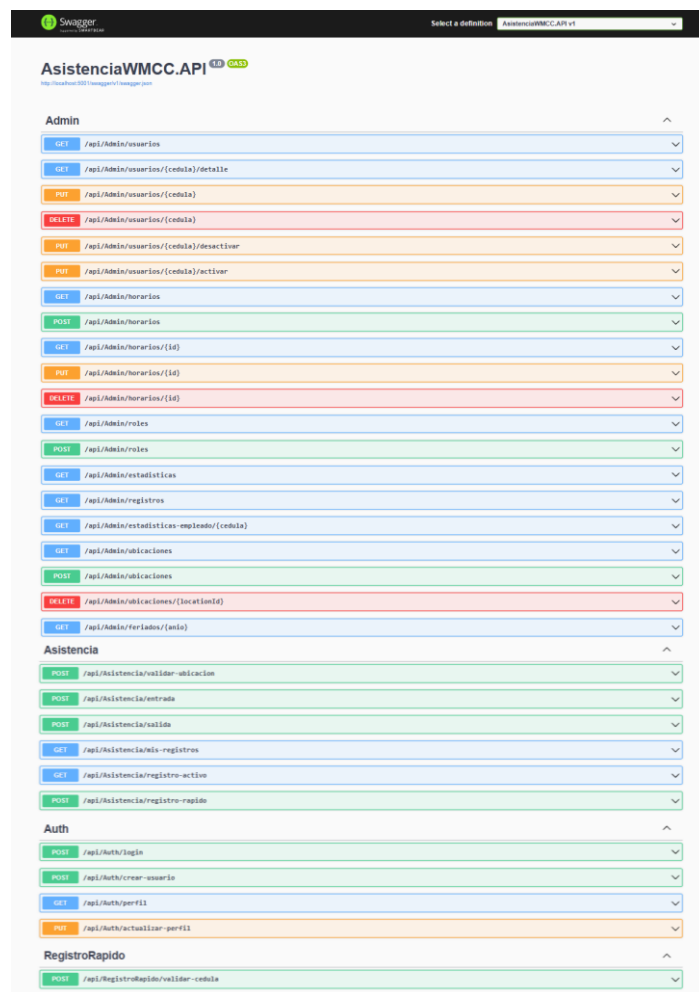


Figura 8. Backend AspNet Core

- **Admin**

Este módulo concentra todas las funcionalidades administrativas del sistema (Tabla 25). Permite la gestión completa de usuarios, horarios, ubicaciones permitidas y días feriados. Solo los usuarios con rol de Administrador pueden acceder a estos endpoints, garantizando que la configuración del sistema esté protegida.

Tabla 25. Apis Admin

Endpoint	Método	Descripción
/api/Admin/usuarios	GET	Lista todos los usuarios registrados en el sistema
/api/Admin/usuarios/{cedula}/detalle	GET	Obtiene información detallada de un usuario específico
/api/Admin/usuarios/{cedula}	PUT	Actualiza información de un usuario existente
/api/Admin/usuarios/{cedula}	DELETE	Elimina permanentemente un usuario del sistema
/api/Admin/usuarios/{cedula}/desactivar	PUT	Desactiva temporalmente un usuario
/api/Admin/usuarios/{cedula}/activar	PUT	Reactiva un usuario previamente desactivado
/api/Admin/horarios	GET	Lista todos los horarios configurados
/api/Admin/horarios	POST	Crea un nuevo horario laboral
/api/Admin/horarios/{id}	GET	Obtiene los detalles de un horario específico
/api/Admin/horarios/{id}	PUT	Actualiza un horario existente
/api/Admin/horarios/{id}	DELETE	Elimina un horario del sistema
/api/Admin/roles	GET	Lista todos los roles disponibles
/api/Admin/roles	POST	Crea un nuevo rol con permisos específicos
/api/Admin/estadisticas	GET	Obtiene estadísticas generales del sistema
/api/Admin/registros	GET	Obtiene todos los registros de asistencia (para reportes)
/api/Admin/estadisticas-empleado/{cedula}	GET	Consultas estadísticas de asistencia de un empleado específico
/api/Admin/ubicaciones	GET	Lista las ubicaciones geográficas permitidas
/api/Admin/ubicaciones	POST	Agrega una nueva ubicación autorizada
/api/Admin/ubicaciones/{locationId}	DELETE	Elimina una ubicación del sistema
/api/Admin/feriados/{anio}	GET	Consulta los días feriados de un año específico

- **Asistencia**

Este módulo es la parte principal del sistema. Gestiona todo el ciclo de vida del registro de asistencia laboral como muestra la Tabla 26 desde la validación de ubicación mediante GPS, pasando por el registro de entradas y salidas, hasta la consulta del historial de asistencias. Implementa validaciones automáticas de horarios, cálculo de horas trabajadas, y garantiza que los empleados solo puedan registrar su asistencia desde ubicaciones autorizadas por la empresa.

Tabla 26. Apis Asistencia

Endpoint	Método	Descripción
/api/asistencia/validar-ubicacion	POST	Verifica si las coordenadas GPS están dentro del radio permitido
/api/asistencia/entrada	POST	Registra entrada del empleado (con validación de ubicación y hora)
/api/asistencia/salida	POST	Registra salida del empleado (calcula horas trabajadas)
/api/asistencia/mis-registros	GET	Obtiene historial de registros del usuario autenticado
/api/asistencia/registro-activo	GET	Verifica si el usuario tiene entrada sin salida registrada
/api/Asistencia/registro-rapido	POST	Permite registro rápido sin usuario registrado en el sistema

- **Auth**

Esta sección gestiona todo lo relacionado con la autenticación, autorización y administración de perfiles de usuario en el sistema como indica la Tabla 27. Es el núcleo de seguridad de la aplicación, controlando quién puede acceder, qué permisos tiene, y cómo los usuarios pueden gestionar su información personal. Todos los endpoints posteriores dependen de la autenticación exitosa en esta sección para funcionar correctamente.

Tabla 27. Apis Auth

Endpoint	Método	Descripción
/api/auth/login	POST	Inicia sesión con email/contraseña, devuelve token JWT
/api/auth/crear-usuario	POST	Crea nuevo usuario (solo admin)
/api/auth/perfil	GET	Obtiene datos del usuario autenticado
/api/auth/actualizar-perfil	PUT	Actualiza nombre, apellido, teléfono, imagen de perfil

- **Registro rápido**

Funcionalidad especial diseñada para visitantes, colaboradores externos o personal temporal que necesita registrar su asistencia sin tener una cuenta completa en el sistema. Permite un registro rápido y simplificado usando únicamente el número de cédula ecuatoriana. El sistema valida automáticamente la cédula usando el algoritmo oficial de validación ecuatoriano. La Tabla 28 muestra la descripción de esta función.

Tabla 28. Api Registro rápido

Endpoint	Método	Descripción
/api/registrorapido/validar-cedula	POST	Valida cédula ecuatoriana y verifica si el usuario existe en el sistema

d. Construcción de la estructura base del proyecto en Flutter.

La idea fue organizar todo de manera que sea fácil de mantener y escalar. Básicamente se dividió, así como se muestra en la Figura 9:

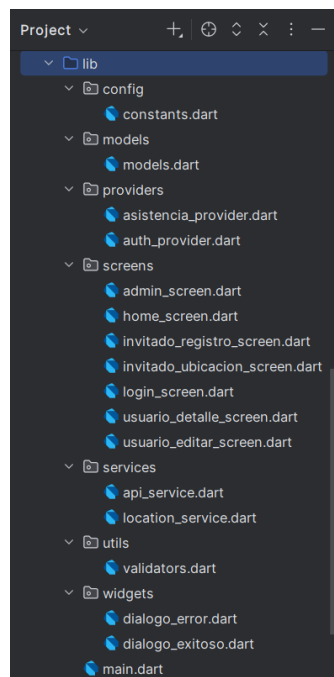


Figura 9. Estructura base del proyecto Flutter

- **lib/config/**

Guarda todas las constantes y configuraciones del proyecto.

constants.dart: URLs del backend, colores corporativos configuraciones generales. Básicamente todo lo que no se necesita cambiar en los archivos diferentes.

- ***lib/models/***

Define la estructura de los datos que maneja la app.

models.dart: Contiene Usuario y Registro. Convierte el JSON que viene del backend en objetos Dart que se usa fácilmente.

- ***lib/providers/***

Maneja el estado global de la app (datos que varias pantallas necesitan).

auth_provider.dart:

- Guarda quién está logueado
- Maneja login/logout
- Guarda el token JWT
- Dice si el usuario es admin o empleado

asistencia_provider.dart:

- Maneja los registros de entrada/salida
- Guarda el historial de asistencias

- ***lib/screens/***

Muestra las pantallas que observa el usuario

- **login_screen.dart:** Pantalla de inicio de sesión.
- **home_screen.dart:** Pantalla principal con menú inferior (inicio, historial, admin, perfil).
- **asistencia_screen.dart:** Donde se registra entrada/salida, muestra botones.
- **historial_screen.dart:** Lista de todos los registros con filtros de fecha.

- **admin_screen.dart**: Solo para administradores muestra estadísticas y gestiona usuarios.
- **perfil_screen.dart**: Información del usuario.
- **invitado_registro_screen.dart**: Registro rápido solo con cédula.
- **invitado_ubicacion_screen.dart**: Validación de ubicación para invitados.
- **usuario_detalle_screen.dart**: Detalles de un empleado específico.
- **usuario_editar_screen.dart**: Editar la información de usuarios.

- ***lib/services/***

Indica los servicios que se comunican con la aplicación.

api_service.dart: Hace todas las peticiones HTTP al backend, login, registro entrada/salida, obtener historial, etc., maneja el token de autenticación.

location_service.dart: Permite solicitar permisos de GPS, obtiene la ubicación actual del sistema y calcula distancias (para controlar que se encuentre dentro del rango permitido).

- ***lib/utils/***

Incluye funciones útiles que se necesitan en las diferentes partes de la aplicación.

validators.dart: apartado donde se controla cédulas ecuatorianas, valida emails y valida contraseñas

- ***lib/widgets/***

Componentes reutilizables para simplificar código.

dialogo_error.dart: Modal de error consistente

dialogo_exitoso.dart: Modal de éxito consistente

e. Construcción de pantallas del aplicativo móvil.

Una vez establecida la estructura base del proyecto, se realizó la construcción de las interfaces visuales que constituyen el sistema. Esta etapa representó la materialización

de los requisitos funcionales en componentes visibles con los que los usuarios finales interactuarían diariamente.

- ***Pantalla: Control Asistencia Iniciar sesión***

La Figura 10 es la primera pantalla de la aplicación donde muestra dos opciones de acceso:



Figura 10. Inicio de sesión de la aplicación

- Empleado/Administrador: ingresan con correo y contraseña.
- Invitado/Colaborador: ingresa usando únicamente su número de cédula. Se debe validar a la persona si se encuentra registrada. Es la puerta de entrada al sistema.

- ***Pantalla: Menú Administrador***

La Figura 11 muestra el menú de un usuario con rol Administrador el cual permite acceder a todas las funciones administrativas:

- Registrar asistencia.
- Listar y crear usuarios.
- Listar y crear horarios.
- Generar reportes.



Figura 11. Menú Administrador

- ***Pantalla: Registro de Asistencia (Administrador)***

Permite al administrador registrar su propia asistencia o la asistencia de otro usuario. Si el administrador va a registrar su asistencia se mostrará la pantalla que se observa en la Figura 12 pero si el administrador va a registrar la asistencia de un empleado después de justificar el motivo registra con la validación de la cédula como se muestra en la Figura 13.



Figura 12. Registro de asistencia

WMCC
— AUTO REPUESTOS —

Control de Asistencia

Registro

Cedula

Validar

Hora

08:00:00 am

Registrar

Figura 13. Registro a otro usuario

- ***Pantalla: Listar Usuarios (Administrador)***

Muestra la tabla de usuarios registrados con opción de búsqueda como se observa en la Figura 14. Incluye botón para **Agregar nuevos usuarios**.

WMCC
— AUTO REPUESTOS —

Menu Administrador

Listar Usuarios

Buscar:

Agregar

ID	Nombre	Apellido	Email	Rol	Acciones
12345678	Juan	Perez	juan.perez@wmcc.com	Administrador	
87654321	Maria	Garcia	maria.garcia@wmcc.com	Usuario	
98765432	Carlos	Rodriguez	carlos.rodriguez@wmcc.com	Usuario	
10987654	Ana	Sanchez	ana.sanchez@wmcc.com	Usuario	
21098765	Diego	Martinez	diego.martinez@wmcc.com	Usuario	
32109876	Isabella	Alvarez	isabella.alvarez@wmcc.com	Usuario	
43210987	Roberto	Jimenez	roberto.jimenez@wmcc.com	Usuario	
54321098	Valeria	Castro	valeria.castro@wmcc.com	Usuario	
65432109	Sebastian	Ortiz	sebastian.ortiz@wmcc.com	Usuario	
76543210	Camila	Chavez	camila.chavez@wmcc.com	Usuario	
87654321	Diego	Sanchez	diego.sanchez@wmcc.com	Usuario	
98765432	Valeria	Alvarez	valeria.alvarez@wmcc.com	Usuario	
10987654	Roberto	Jimenez	roberto.jimenez@wmcc.com	Usuario	
21098765	Isabella	Castro	isabella.castro@wmcc.com	Usuario	
32109876	Sebastian	Ortiz	sebastian.ortiz@wmcc.com	Usuario	
43210987	Camila	Chavez	camila.chavez@wmcc.com	Usuario	
54321098	Diego	Sanchez	diego.sanchez@wmcc.com	Usuario	
65432109	Valeria	Alvarez	valeria.alvarez@wmcc.com	Usuario	
76543210	Roberto	Jimenez	roberto.jimenez@wmcc.com	Usuario	
87654321	Isabella	Castro	isabella.castro@wmcc.com	Usuario	
98765432	Sebastian	Ortiz	sebastian.ortiz@wmcc.com	Usuario	
10987654	Camila	Chavez	camila.chavez@wmcc.com	Usuario	

Atras

Figura 14. Listar Usuarios

- ***Pantalla: Crear Usuario (Administrador)***

Formulario completo que permite registrar un usuario nuevo con los siguientes campos: Cédula, nombre, apellido, email, contraseña, rol, horario, número de teléfono e imagen. Es utilizada para la administración del personal (Figura 15).

Figura 15. Crear usuarios

- ***Pantalla: Listar Horarios (Administrador)***

Se observa en la Figura 16 la pantalla en la cual muestra los horarios cargados y configurados en el sistema. Incluye búsqueda y botón para crear nuevos horarios.

ID	Día	Hora	Inicio	Fin	Estado	Acciones
1	Lunes	08:00	08:00	12:00	Activo	[Iconos]
2	Martes	08:00	08:00	12:00	Activo	[Iconos]
3	Miércoles	08:00	08:00	12:00	Activo	[Iconos]
4	Jueves	08:00	08:00	12:00	Activo	[Iconos]
5	Viernes	08:00	08:00	12:00	Activo	[Iconos]
6	Sábado	08:00	08:00	12:00	Activo	[Iconos]
7	Domingo	08:00	08:00	12:00	Activo	[Iconos]

Figura 16. Listar horarios

- ***Pantalla Crear Horarios (Administrador)***

La pantalla de Crear Horario (Figura 17) se llena un formulario para la creación de horarios laborales:

- Selección de día
- Hora de entrada

- Hora de salida
- Inicio de almuerzo
- Fin de almuerzo

Permite tener control detallado de la jornada laboral.

Figura 17. Crear horarios

- ***Pantalla: Verificación de Ubicación***

Antes de registrar asistencia, el sistema valida la ubicación del usuario. Garantiza que el registro se realice dentro del área autorizada como se muestra la Figura 18.



Figura 18. Verificación de ubicación

- ***Pantalla: Registro con Validación***

Esta pantalla solicita la cédula del usuario para validar su identidad antes del registro. Esta pantalla está diseñada para los colaboradores que se registren sin ser empleados y se pueda pagar lo justo (Figura 19).



Figura 19. Registro con cédula

- ***Pantalla: Registro de Asistencia (Hora)***

En la pantalla de Hora para el registro como se observa en la Figura 20 muestra la hora actual y el botón para registrar. Es la pantalla donde el empleado confirma el registro.



Figura 20. Registro de empleados

- ***Pantalla: Registro Exitoso - Registro No Completado***

- Registro exitoso (Figura 21): Confirma que la asistencia ha sido registrada correctamente. Incluye la hora y un botón para salir.
- Registro no completado (Figura 22): Muestra un mensaje indicando que el registro no pudo realizarse, acompañado del motivo. Permite al usuario salir o intentarlo nuevamente.



Figura 21. Registro exitoso



Figura 22. Registro incompleto

3.5.5 Pruebas

La fase de pruebas constituye una etapa fundamental para garantizar que el sistema cumple con los requerimientos establecidos y funciona correctamente en condiciones reales de operación. Se implementó un proceso de validación exhaustivo que abarcó múltiples dimensiones del sistema, desde la funcionalidad básica hasta aspectos críticos de seguridad y rendimiento como se observa en la Tabla 29:

Tabla 29. Etapa 5 Pruebas

Tarea	Descripción	Estado
Pruebas funcionales	Validación de todas las funcionalidades del sistema según requerimientos.	Completado
Pruebas de integración	Verificación de comunicación entre frontend, backend y base de datos	Completado
Pruebas de seguridad	Evaluación de autenticación, autorización y protección de datos	Completado
Pruebas de geolocalización	Validación precisa del sistema GPS y cálculo de distancias	Completado
Pruebas de usabilidad	Evaluación con usuarios reales en dispositivos diversos.	Completado

a. Pruebas funcionales

Las pruebas funcionales se enfocaron en verificar que cada módulo del sistema cumpliera con los requisitos especificados. Se diseñaron casos de prueba para cubrir tanto escenarios normales como situaciones excepcionales que podrían presentarse durante la operación diaria.

• Módulo de Autenticación

Se validó el proceso completo de inicio de sesión, incluyendo casos exitosos y fallidos. Las pruebas incluyeron:

- Inicio de sesión con credenciales válidas para empleados y administradores
- Intento de inicio de sesión con contraseña incorrecta (máximo 3 intentos)
- Intento de inicio de sesión con usuario inexistente
- Validación de formato de email

- Verificación de tokens JWT generados correctamente
- Cierre de sesión y eliminación de tokens
- Manejo de sesiones simultáneas en múltiples dispositivos

Resultado: Se identificó que inicialmente el sistema permitía más de 3 intentos fallidos de login, lo cual se corrigió implementando un bloqueo temporal de 15 minutos después del tercer intento. Todos los casos de prueba pasaron satisfactoriamente después de esta corrección.

- ***Módulo de Registro de Asistencia***

Este módulo constituye la función principal del sistema:

- Registro de entrada dentro del horario permitido
- Registro de entrada con retraso (validación de tolerancia)
- Intento de registro duplicado (entrada cuando ya existe entrada activa)
- Registro de salida con cálculo automático de horas trabajadas
- Validación de registro de almuerzo (inicio y fin)
- Registro rápido solo con cédula (para colaboradores externos)
- Manejo de registros incompletos (entrada sin salida del día anterior)

Resultado: Se detectó un error en el cálculo de horas trabajadas cuando el registro de salida ocurría después de medianoche. Se implementó una corrección que considera correctamente los registros que abarcan dos días calendarios diferentes. Adicionalmente, se mejoró el mensaje de error cuando un usuario intenta registrar entrada teniendo ya una entrada activa sin salida correspondiente.

- ***Módulo de Gestión de Usuarios (Administrador)***

Las pruebas del módulo administrativo cubrieron:

- Creación de nuevos usuarios con todos los campos requeridos
- Validación de cédula ecuatoriana mediante algoritmo
- Edición de información de usuarios existentes

- Asignación y modificación de roles
- Asignación de horarios personalizados
- Activación y desactivación de usuarios
- Búsqueda y filtrado de usuarios por diferentes criterios

Resultado: Todas las validaciones funcionaron correctamente. Se agregó una confirmación adicional antes de desactivar usuarios para evitar cambios accidentales.

b. Pruebas de seguridad

La seguridad fue evaluada considerando que el sistema maneja información sensible de empleados:

- ***Autenticación y Autorización***

- Verificación de que los tokens JWT expiren correctamente después de 8 horas
- Intento de acceso a endpoints protegidos sin token válido
- Validación de que empleados no puedan acceder a funciones administrativas
- Comprobación de que tokens manipulados sean rechazados
- Verificación de que las contraseñas se almacenen encriptadas (nunca en texto plano)

Resultado: Se confirmó que el sistema de autenticación funciona correctamente. Las contraseñas se encriptan usando BCrypt con factor de trabajo 12, y los tokens JWT incluyen información del rol del usuario para validación en cada petición.

- ***Protección de Datos Personales***

- Verificación de que las coordenadas GPS solo se almacenen durante registros de asistencia
- Comprobación de que la información personal no sea visible en logs del servidor
- Validación de que las imágenes de perfil se almacenen con permisos restringidos
- Prueba de que usuarios no puedan acceder a datos de otros empleados

Resultado: Todas las pruebas pasaron satisfactoriamente. Se implementó encriptación adicional para números de cédula en tránsito usando HTTPS obligatorio.

c. Pruebas de geolocalización

Dado que la validación de ubicación es crítica para el funcionamiento del sistema, se realizaron pruebas exhaustivas del módulo de geolocalización:

- ***Precisión del GPS***

Se realizaron 50 registros de prueba desde la ubicación autorizada de Autorepuestos W.M.C.C., midiendo la variación en las coordenadas obtenidas:

- Promedio de precisión: ± 3.2 metros
- Desviación estándar: 1.8 metros
- Todos los registros dentro del radio de 5 metros configurado}

Resultado: El radio de 5 metros fue confirmado como adecuado, permitiendo registros válidos desde cualquier punto dentro de las instalaciones de la empresa mientras rechaza intentos desde ubicaciones externas.

- ***Permisos de Ubicación***

Se probó el comportamiento del sistema ante diferentes estados de permisos GPS:

- Permisos concedidos: Sistema funciona normalmente
- Permisos denegados inicialmente: Muestra mensaje explicativo y solicita permisos nuevamente
- Permisos revocados durante uso: Detecta cambio y solicita reactivación
- GPS desactivado en el dispositivo: Indica al usuario que debe activar el GPS

Resultado: Se implementó un flujo claro que guía al usuario paso a paso cuando los permisos de ubicación no están disponibles, reduciendo confusión y llamadas de soporte.

3.5.6 Finalizar antes de empezar

Esta etapa es fundamental en la metodología Kanban, ya que establece límites en el trabajo en progreso (Work In Progress - WIP) para evitar la sobrecarga del equipo y garantizar que las tareas se completen antes de iniciar nuevas.

a. Principio "Stop Starting, Start Finishing"

El principio fundamental de esta etapa es priorizar la finalización de tareas (Tabla 30) en curso antes de iniciar nuevas actividades.

Tabla 30. Limitantes

Columna del Tablero	Límite WIP	Justificación
Por hacer	Sin límite	Backlog de tareas planificadas
En progreso	3 tareas máximo	Evita multitarea excesiva
En revisión	2 tareas máximo	Garantiza revisión exhaustiva
Bloqueado	1 tarea máximo	Resolución prioritaria de impedimentos
Finalizado	Sin límite	Registro histórico completo

b. Reglas de transición entre estados

Para garantizar el cumplimiento de los límites WIP, se establecieron reglas claras para el movimiento de tareas entre columnas definidas en la Tabla 31:

Tabla 31. Reglas de transición de tareas

Transición	Condición Requerida	Responsable	Acción si WIP completo
Por hacer → En progreso	WIP < 3 en "En progreso"	Desarrollador	Esperar a completar una tarea existente
En progreso → En revisión	WIP < 2 en "En revisión"	Desarrollador	Finalizar revisiones pendientes primero
En revisión → Finalizado	Pruebas aprobadas	Gerente	N/A
Cualquier → Bloqueado	Impedimento técnico	Desarrollador	Resolver bloqueo inmediatamente

3.5.7 Despliegue

La etapa de despliegue comprende todas las actividades necesarias para poner el sistema en producción y disponible para los usuarios finales.

a. Despliegue del Backend (API)

Para el despliegue del backend se siguió un proceso el cual se puede evidenciar en la Tabla 32.

Tabla 32. Pasos despliegue API

Tarea	Descripción	Estado
Compilar proyecto para producción	Generar build de ASP.NET Core sin errores ni advertencias	Completado
Configurar variables de entorno	Establecer ConnectionStrings y JWT Secret Key en Servidor Local Key Vault	Completado
Publicar API en Servidor Local App Service	Desplegar la API mediante Visual Studio Publish en el servidor de producción	Completado
Configurar CORS	Añadir dominios autorizados para permitir peticiones solo desde la app móvil	Completado
Habilitar HTTPS forzado	Configurar redirección automática de HTTP a HTTPS para todas las peticiones	Completado
Configurar Rate Limiting	Establecer límite de 100 requests por minuto para prevenir ataques DDoS	Completado
Probar endpoints en producción	Verificar funcionamiento de todos los endpoints usando Postman	Completado

b. Despliegue del Frontend

Para el despliegue de la aplicación móvil Flutter se siguió un proceso sistemático que garantiza la correcta instalación y funcionamiento en los dispositivos de los empleados, como se evidencia en la Tabla 33.

Tabla 33. Pasos despliegue aplicación móvil

Tarea	Descripción	Estado
Configurar variables de entorno	Establecer URL del backend de producción en archivo de configuración (constants.dart)	Completado
Actualizar dependencias	Ejecutar flutter pub get y verificar compatibilidad de todas las librerías	Completado

Generar build de producción	Compilar aplicación en modo release eliminando código de depuración mediante flutter build apk --release	Completado
Configurar permisos Android	Validar permisos de ubicación, internet y almacenamiento en AndroidManifest.xml	Completado
Optimizar tamaño del APK	Aplicar ofuscación de código y compresión de recursos para reducir tamaño final	Completado
Probar instalación en dispositivos reales	Instalar APK en al menos 3 dispositivos diferentes (Samsung, Xiaomi, Apple) para validar compatibilidad	Completado
Crear manual de instalación	Elaborar guía paso a paso con capturas de pantalla para instalación del APK	Completado
Distribuir APK internamente	Compartir APK mediante Google Drive o WhatsApp con el personal de Autorepuestos W.M.C.C.	Completado
Habilitar instalación de fuentes desconocidas	Capacitar al personal sobre cómo permitir instalación de aplicaciones fuera de Play Store	Completado
Validar funcionamiento completo	Verificar iniciar sesión, registro de asistencia, geolocalización y reportes en dispositivos de producción	Completado

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

El presente capítulo sintetiza los hallazgos más relevantes obtenidos durante el desarrollo de este proyecto de investigación, el cual tuvo como propósito fundamental la implementación de una aplicación móvil con Flutter y ASP.NET Core para el control de cumplimiento laboral en la empresa Autorepuestos W.M.C.C.

4.1 Conclusiones

- Se logró analizar de manera exhaustiva el nivel de cumplimiento laboral actual en la empresa Autorepuestos W.M.C.C., identificando las principales falencias en los métodos de control existentes. Mediante la aplicación de encuestas, entrevistas y observación directa, se evidenció que el sistema tradicional de registro manual presentaba inconsistencias significativas como: ausencia de verificación de ubicación, facilidad para marcaciones fraudulentas y carencia de reportes automatizados. Estos hallazgos permitieron establecer con precisión los requerimientos funcionales y no funcionales que debía cumplir la solución tecnológica, garantizando que el sistema desarrollado responda efectivamente a las necesidades operativas identificadas en el personal administrativo, operativo y de supervisión de la empresa.
- La investigación de la tecnología Flutter para el desarrollo de aplicaciones móviles multiplataforma demostró su alta efectividad y versatilidad en el contexto del proyecto. Esta capacidad multiplataforma, interfaz de usuario altamente personalizable y amplio medio de paquetes para funcionalidades avanzadas como geolocalización y autenticación, confirmaron que Flutter constituye una opción óptima para proyectos que requieren alcance multiplataforma sin sacrificar calidad ni experiencia de usuario.
- El sistema integral de control de cumplimiento laboral incluye 7 módulos funcionales principales: autenticación segura mediante tokens JWT, validación de geolocalización mediante GPS con radio de precisión de 5 metros, registro automatizado de asistencia con fecha, hora y ubicación, sistema de notificaciones en tiempo real y gestión de usuarios y horarios. La metodología

Kanban aplicada durante el desarrollo permitió una gestión eficiente mediante 7 etapas iterativas, facilitando la validación continua de funcionalidades y la adaptación oportuna a los requerimientos de los empleados participantes. El sistema implementado responde directamente a las falencias identificadas: el 86% del personal (12/14 empleados) expresó disposición favorable para usar la aplicación, y el 100% de los encuestados confía en que el sistema mejorará la eficiencia del control de asistencia, contrastando con el sistema manual previo donde solo el 64% cumplía con puntualidad y se registró un 7% de manipulación de registros (1 caso confirmado de 14). La solución tecnológica proporciona información precisa y verificable mediante 15 endpoints API desarrollados que procesan validaciones geográficas, registros de asistencia y generación de reportes para la toma de decisiones gerenciales.

4.2 Recomendaciones

- La combinación de Kanban con Flutter y ASP.NET Core que se utilizó en este proyecto puede aplicarse en otros sistemas similares para pequeñas y medianas empresas, como control de inventarios, seguimiento de entregas, facturación electrónica o gestión de mantenimiento. La gestión visual con ClickUp y los límites de tareas en progreso funcionaron bien para un equipo pequeño y un tiempo limitado. Para futuras investigaciones, sería interesante probar metodologías híbridas que combinen Kanban con Scrum, o aplicar este mismo enfoque en proyectos más complejos que incluyan inteligencia artificial o IoT, para comparar qué metodología funciona mejor según el tipo de proyecto y el tamaño del equipo.
- Se recomienda a los estudiantes y docentes de la carrera a continuar investigando sobre transformación digital en PyMEs ecuatorianas. Los resultados obtenidos (86% de aceptación del sistema, 36% de impuntualidad en el método manual, 7% de manipulación de registros) pueden servir como base para estudios futuros: un seguimiento del impacto a 6 y 12 meses después de implementar el sistema, una comparación con otras tecnologías como huella digital o reconocimiento facial para ver cuál funciona mejor, o un análisis de

costos versus beneficios considerando el tiempo ahorrado y las pérdidas evitadas.

- Se recomienda a Autorepuestos W.M.C.C. implementar el sistema en tres fases.
 1. Fase 1: Capacitar al personal de forma presencial durante 2 horas, entregar manuales con imágenes claras, crear un grupo de WhatsApp para soporte técnico, instalar la app en los celulares de todos y comunicar oficialmente las nuevas reglas de asistencia.
 2. Fase 2: Revisar semanalmente si hay errores en el sistema, analizar cuántas personas están usando la app correctamente, hacer respaldos automáticos de la base de datos, aplicar actualizaciones de seguridad necesarias y reunirse cada 20 días con el personal para escuchar sus comentarios.
 3. Fase 3: Medir si realmente bajó la impuntualidad comparando con el 36% inicial, calcular cuánto tiempo se ahorró en reportes (antes 4 horas semanales), hacer una encuesta para saber si el personal está satisfecho con el sistema, y considerar agregar nuevas funciones como solicitud de permisos, control de horas extras, evaluación de desempeño y reportes automáticos. Estas mejoras convertirían al sistema en una herramienta completa de gestión de personal y maximizarían el beneficio de la inversión realizada.

REFERENCIAS BIBLIOGRÁFICAS

- [1] G. Flores y L. Molina, "Sistemas de control de asistencia laboral en pymes ecuatorianas," *Revista Científica Dominio de las Ciencias*, vol. 8, no. 2, pp. 455–471, 2022.
- [2] A. Rodríguez, "Limitaciones del control manual de asistencia y la necesidad de automatización," *Revista de Gestión Empresarial*, vol. 14, no. 3, pp. 97–102, 2021.
- [3] D. Pérez, "Transformación digital y su impacto en la gestión de recursos humanos," *Revista Innovación y Tecnología*, vol. 10, no. 1, pp. 35–42, 2022.
- [4] Google, "Flutter - Build apps for any screen," [En línea]. Disponible: <https://flutter.dev/>.
- [5] Microsoft, "ASP.NET Core documentation," [En línea]. Disponible: <https://docs.microsoft.com/en-us/aspnet/core/>.
- [6] R. Mamoun, M. Nasor y S. H. Abulikailik, "Design and Development of Mobile Healthcare Application Prototype Using Flutter," en *2020 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, 2021, doi: 10.1109/ICCCEEE49695.2021.9429595.
- [7] A. Zefrinaldi, N. Selviandro y G. S. Wulandari, "Analysis and Development of a Football Scouting App based on Flutter: A Case Study of A3N," *ADI Journal on Recent Innovation*, vol. 5, no. 2, pp. 181–191, 2024, doi: 10.34306/ajri.v5i2.1051
- [8] S. Tiwari et al., "Utsarjan: A smartphone App for providing kidney care and real-time assistance to children with nephrotic syndrome," *arXiv preprint arXiv:2504.13863*, 2025.
- [9] A. Saxena et al., "Design and Implementation of Flutter based Multi-platform Docker Controller App," *arXiv preprint arXiv:2502.11708*, 2025.
- [10] P. Alves y B. P. Cipriano, "Automated Assessment in Mobile Programming Courses: Leveraging GitHub Classroom and Flutter for Enhanced Student Outcomes," *arXiv preprint arXiv:2504.04230*, 2025.

- [11] C. Belloch Ortí, "Las tecnologías de la información y la comunicación en la educación," Universidad de Valencia. [En línea]. Disponible en: <https://www.uv.es/~bellochc/pdf/pwtic1.pdf>.
- [12] Real Academia Española, "Informática," Diccionario de la lengua española, 2001. [En línea]. Disponible en: <https://www.rae.es/drae2001/inform%C3%A1tica>.
- [13] IBM, "¿Qué es el desarrollo de software?," [En línea]. Disponible en: <https://www.ibm.com/es-es/topics/software-development>.
- [14] Flutter, "Crea aplicaciones para cualquier pantalla," [En línea]. Disponible en: <https://esflutter.dev/>.
- [15] A. Augusto, G. Guevara y A. Arteaga, "La industria automotriz en América Latina," Revista Polis, UAM, 2020. [En línea]. Disponible en: <https://polismexico.izt.uam.mx/index.php/rp/article/view/747/733>.
- [16] Kimaldi, "Control de Personal," [En línea]. Disponible en: https://www.kimaldi.com/blog/control_de_acceso_y_presencia/control_de_personal/.
- [17] Genuino Cloud, "La Importancia de Tipos en la Reducción de Errores," [En línea]. Disponible en: <https://genuinocloud.com/blog/la-importancia-de-tipos-en-la-reduccion-de-errores/>.
- [18] Economipedia, "Eficiencia operativa," [En línea]. Disponible en: <https://economipedia.com/definiciones/eficiencia-operativa.html>.
- [19] R. Pressman y B. Maxim, *Ingeniería del Software: Un Enfoque Práctico*, 9ª ed., México D.F.: McGraw-Hill, 2021.
- [20] D. J. Anderson, *Kanban: Successful Evolutionary Change for Your Technology Business*, Sequim, WA: Blue Hole Press, 2010.
- [21] Google, *Android Studio User Guide*, Mountain View, CA: Google LLC, 2024. [En línea]. Disponible en: <https://developer.android.com/studio>
- [22] Microsoft, *Introduction to ASP.NET Core*, Redmond, WA: Microsoft Corporation, 2024. [En línea]. Disponible en: <https://learn.microsoft.com/aspnet/core>
- [23] Google, *Google Maps Platform Documentation*, Mountain View, CA: Google LLC, 2024. [En línea]. Disponible en: <https://developers.google.com/maps>

- [24] Microsoft, *C# Programming Guide*, Redmond, WA: Microsoft Corporation, 2024. [En línea]. Disponible en: <https://learn.microsoft.com/dotnet/csharp>
- [25] Google, *Flutter Documentation*, Mountain View, CA: Google LLC, 2024. [En línea]. Disponible en: <https://flutter.dev/docs>
- [26] Microsoft, *Microsoft SQL Server Documentation*, Redmond, WA: Microsoft Corporation, 2024. [En línea]. Disponible en: <https://learn.microsoft.com/sql>
- [27] L. Hernández, "Técnicas de recolección de datos en proyectos tecnológicos," *Revista Científica de Investigación Aplicada*, vol. 9, no. 1, pp. 55–64, 2023.
- [28] E. Windmill y M. Napoli, *Flutter in Action*, Manning Publications, 2020.
- [29] M. Moroney, "Introduction to Flutter Development Using Dart," *Apress*, 2020.
- [30] A. Biessek, "Flutter: Mobile Development Framework," *IEEE Software*, vol. 37, no. 2, pp. 38-44, 2020.
- [31] P. Nayak y P. Pawar, "Cross-platform mobile application development: A comparison study of React Native and Flutter," *International Journal of Engineering Research & Technology (IJERT)*, vol. 9, no. 5, pp. 394-398, 2020.
- [32] A. Ferreira, V. Silva y J. Martins, "Performance evaluation of cross-platform mobile development frameworks," *Procedia Computer Science*, vol. 170, pp. 841-846, 2020.
- [33] J. Wu, "Research and application of Flutter framework in mobile application development," *Journal of Physics: Conference Series*, vol. 1187, no. 5, pp. 052-063, 2019.
- [34] S. Latif, "Cross-platform development approaches for mobile applications," *Journal of Computer Science*, vol. 16, no. 7, pp. 1022-1029, 2020.
- [35] M. Linares-Vásquez, G. Bavota, C. Bernal-Cárdenas et al., "Multi-platform mobile app development: A comparison of React Native and Flutter," *Empirical Software Engineering*, vol. 25, pp. 2763-2799, 2020.
- [36] R. Francese, C. Gravino, M. Risi y G. Tortora, "Mobile development with Flutter: An evaluation," en *International Conference on Mobile Web and Intelligent Information Systems*, Springer, 2019, pp. 119-132.
- [37] Google, "Flutter architectural overview," Flutter.dev Documentation, 2024. [En línea]. Disponible: <https://flutter.dev/docs/resources/architectural-overview>
- [38] J. Majchrzak, G. Söllner y T. Ernsting, "State of practice in flutter-based mobile application development," *arXiv preprint arXiv:2103.12905*, 2021.

- [39] D. Jiang, "Analysis and comparison of cross-platform mobile application development framework," *Journal of Physics: Conference Series*, vol. 1544, no. 1, pp. 012-102, 2020.
- [40] N. Irtaza, A. Bibi y M. Ahmad, "Comparative analysis of Flutter with existing technologies," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 6, pp. 572-579, 2020.
- [41] Dart team, "Dart programming language," Dart.dev, 2024. [En línea]. Disponible: <https://dart.dev/>

ANEXOS