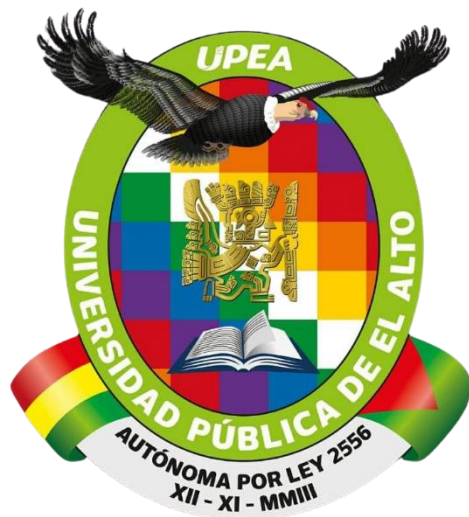


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

**“SISTEMA MULTIPLATAFORMA PARA LA GESTIÓN DE
OPERACIONES EN EL SECTOR DE
TELECOMUNICACIONES”**

CASO: DAIFO TELECOM S.R.L.

Para optar al título de Licenciatura en Ingeniería de Sistemas

MENCIÓN: GESTIÓN Y PRODUCCIÓN

Postulante: Abdon Cristhian Perez Garnica

Tutor Metodológico: M.Sc. Lic. Ing. Fanny Helen Perez Mamani

Tutor Especialista: Lic. Ing. Sergio Ramiro Rojas Saire

Tutor Revisor: Lic. Ing. Dionicio Henry Pacheco Rios

EL ALTO – BOLIVIA

2024

DECLARACIÓN JURADA DE AUTENTICIDAD Y RESPONSABILIDAD

Yo **Abdon Cristhian Perez Garnica** estudiante con C.I. **8422858 LP**. mediante la presente declaro de manera pública que la propuesta del **TRABAJO DE GRADO** titulada: “**SISTEMA MULTIPLATAFORMA PARA LA GESTIÓN DE OPERACIONES EN EL SECTOR DE TELECOMUNICACIONES**”, **CASO: DAIFO TELECOM S.R.L.** es original, siendo resultado de mi trabajo personal y no constituye una copia o replica de trabajos similares elaborados.

Autorizo la publicación del resumen de mi propuesta en internet y me comprometo a responder a todos los cuestionamientos que se desprenden de su lectura.

Asimismo, me hago responsable ante la universidad o terceros, de cualquier irregularidad o daño que pudiera ocasionar, por el incumplimiento de lo declarado.

De identificarse falsificación, plagio, fraude, o que el **TRABAJO DE GRADO** haya sido publicado anteriormente; asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándome por todas las cargas legales que se deriven de ello sometiéndome a las normas establecidas y vigentes de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

El Alto, noviembre de 2024

Abdon Cristhian Perez Garnica
C.I. 8422858 LP
e-mail: abdonperez32@gmail.com

DEDICATORIA

A mi esposa, mi compañera de vida, quien ha estado a mi lado en cada momento, ofreciendo su apoyo, paciencia y amor incondicional, lo que ha sido esencial para superar los desafíos y celebrar los logros juntos.

A mi madre, cuyo apoyo desde el inicio de este camino ha sido fundamental. Su amor y su fortaleza han sido pilares en cada paso que he dado, enseñándome la perseverancia necesaria para alcanzar mis objetivos.

Y a mis hijos, Cristhian y Jazmin, quienes son mi mayor alegría y motivación. Ellos son la razón por la que aspiro a ser mejor cada día y a quien espero inspirar con el ejemplo de que con esfuerzo y dedicación, cualquier sueño es alcanzable.

Dedico este esfuerzo a todos ustedes, cuya presencia ha sido la luz que guía mi camino. Este logro es tanto mío como suyo, reflejo de las enseñanzas y el amor que he recibido a lo largo de este viaje.

AGRADECIMIENTOS

Quiero expresar mi profundo agradecimiento a mis tutores, cuya dedicación y sabiduría han sido fundamentales para mi desarrollo y la culminación de este proyecto:

A mi Tutora Metodológica M.Sc. Lic. Ing. Fanny Helen Perez Mamani, cuya guía experta en cada etapa del proyecto fue indispensable. Su paciencia y motivación no solo enriquecieron mi aprendizaje, sino que también impulsaron mi compromiso hacia la excelencia.

A mi Tutor Especialista Lic. Ing. Sergio Ramiro Rojas Saire, a quien agradezco profundamente por su tiempo y paciencia en el seguimiento detallado y la revisión crítica de este trabajo. Sus observaciones agudas y recomendaciones han sido cruciales para dar forma al documento final.

A mi Tutor Revisor Lic. Ing. Dionicio Henry Pacheco Rios, por su apoyo constante, sus revisiones meticulosas, y sus palabras alentadoras que me motivaron a persistir ante los desafíos. Su orientación precisa fue vital para mantener el rumbo correcto del proyecto.

También extendo mi gratitud a la Universidad Pública de El Alto y la Carrera de Ingeniería de Sistemas por ser mi hogar académico y punto de partida de este viaje de aprendizaje. La universidad no solo me proporcionó los recursos necesarios, sino también un entorno en el que pude crecer tanto personal como profesionalmente.

ÍNDICE DE CONTENIDO

1. MARCO PRELIMINAR	2
1.1. Introducción	2
1.2. Antecedentes.....	3
1.2.1. Antecedentes Institucionales	3
1.2.1.1. Visión.....	3
1.2.1.2. Misión.....	4
1.2.2. Antecedentes afines al proyecto de grado.....	4
1.2.2.1. Antecedentes Internacionales.....	4
1.2.2.2. Antecedentes Nacionales.....	5
1.3. Planteamiento del problema.....	6
1.3.1. Problema principal.....	6
1.3.2. Problemas secundarios	6
1.3.3. Formulación del problema	7
1.4. Objetivos.....	7
1.4.1. Objetivo General	7
1.4.2. Objetivos Específicos	7
1.5. Justificación	7
1.5.1. Justificación Técnica	7
1.5.2. Justificación Económica	8
1.5.3. Justificación Social	8
1.6. Metodología.....	8
1.6.1. Metodología de Desarrollo Iterativo e Incremental.....	8
1.6.1.1. Características de la Metodología Iterativa e Incremental.....	9
1.6.1.2. Principales Aspectos de la Metodología Iterativa e Incremental.....	9
1.6.1.3. Fases del Desarrollo Iterativo e Incremental.....	10
1.6.2. Métricas de Calidad.....	10
1.6.2.1. Funcionalidad.....	10
1.6.2.2. Confiabilidad.....	10
1.6.2.3. Usabilidad.....	10
1.6.2.4. Eficiencia.....	11
1.6.2.5. Mantenibilidad.....	11
1.6.2.6. Portabilidad.....	11
1.6.3. Costo.....	11
1.6.4. Seguridad.....	12
1.6.5. Pruebas de Software	12
1.6.5.1. Pruebas Unitarias.....	12
1.6.5.2. Pruebas de Integración.....	13
1.6.5.3. Pruebas Funcionalidad.....	13
1.7. Herramientas	13
1.7.1. Herramientas de Software	13
1.7.2. Herramientas de Hardware.....	14
1.8. Límites y alcances	15
1.8.1. Límites.....	15

1.8.2. Alcances.....	16
1.9. Aportes	16
2. MARCO TEÓRICO	19
2.1. Introducción	19
2.2. Sistema de Información	21
2.2.1. Definición de Sistema de Información.....	21
2.2.2. Características de un Sistema Multiplataforma.....	21
2.2.3. Gestión de Operaciones en Telecomunicaciones.....	22
2.2.4. Operaciones Específicas en el Sector de Telecomunicaciones	22
2.3. Herramientas de Desarrollo: Frontend y Backend.....	23
2.3.1. Vue.js	23
2.3.2. Node.js.....	24
2.3.3. MySQL	24
2.3.4. Vuetify	24
2.3.5. Leaflet	25
2.3.6. WebSockets	25
2.3.7. Sequelize	25
2.3.8. Sharp	25
2.3.9. Nanoid.....	26
2.4. Herramientas de Desarrollo: Aplicaciones Android y Windows	26
2.4.1. Apache Cordova.....	26
2.4.2. Android.....	27
2.4.3. Aplicativos	27
2.4.4. Electron	27
2.4.5. Windows.....	28
2.4.6. Aplicaciones de Escritorio.....	28
2.5. Herramientas de desarrollo: Hardware.....	29
2.5.1. ESP32.....	29
2.5.2. BME280	29
2.5.3. MPU6050	30
2.5.4. OLED	30
2.5.5. YL-83	30
2.6. Metodología de Desarrollo Iterativo e Incremental	31
2.6.1. Características de la Metodología Iterativa e Incremental.....	31
2.6.2. Fases del Desarrollo Iterativo e Incremental.....	32
2.7. Métricas de Calidad	34
2.7.1. Funcionalidad.....	35
2.7.2. Confiabilidad.....	35
2.7.3. Usabilidad	36
2.7.4. Eficiencia.....	36
2.7.5. Mantenibilidad	36
2.7.6. Portabilidad	37
2.8. Costo	37
2.8.1. Identificación de Funcionalidades.....	38

2.8.2. Asignación de Complejidad	38
2.8.3. Cálculo de Esfuerzo	38
2.9. Seguridad	39
2.9.1. Autenticación y Contraseñas	39
2.9.2. Encriptación y Tokens	39
2.9.3. Captura de Firmas.....	40
2.9.4. Validación de Peticiones	40
2.9.5. Copias de Seguridad	40
2.9.6. Firma de Aplicaciones	41
3. MARCO APLICATIVO	43
3.1. Introducción	43
3.2. Esquema del sistema.....	44
3.3. Aplicación de la Metodología Iterativa e Incremental	45
3.3.1. Primera Iteración	45
3.3.1.1. Planificación de Iteración	45
3.3.1.2. Desarrollo y Pruebas.....	54
3.3.1.3. Revisión y Retroalimentación	65
3.3.1.4. Lanzamiento de Versión (1.0.0)	66
3.3.2. Segunda Iteración	68
3.3.2.1. Planificación de Iteración	68
3.3.2.2. Desarrollo y Pruebas.....	71
3.3.2.3. Revisión y Retroalimentación	80
3.3.2.4. Lanzamiento de Versión (1.1.0)	81
3.3.3. Tercera Iteración	81
3.3.3.1. Planificación de Iteración	81
3.3.3.2. Desarrollo y Pruebas.....	85
3.3.3.3. Revisión y Retroalimentación	94
3.3.3.4. Lanzamiento de Versión (1.2.0)	94
3.3.4. Cuarta Iteración.....	94
3.3.4.1. Planificación de Iteración	95
3.3.4.2. Desarrollo y Pruebas.....	98
3.3.4.3. Revisión y Retroalimentación	104
3.3.4.4. Lanzamiento de Versión (1.3.0)	104
3.3.5. Quinta Iteración	105
3.3.5.1. Planificación de Iteración	105
3.3.5.2. Desarrollo y Pruebas.....	107
3.3.5.3. Revisión y Retroalimentación	110
3.3.5.4. Lanzamiento de Versión (2.0.0)	110
3.3.6. Sexta Iteración	111
3.3.6.1. Planificación de Iteración	111
3.3.6.2. Desarrollo y Pruebas.....	112
3.3.6.3. Revisión y Retroalimentación	116
3.3.6.4. Lanzamiento de Versión (2.1.0)	116
3.4. Revisión de Versión Final	116

3.4.1. Estructura de la Base de Datos	116
3.4.2. Diagrama de Conexión del Sistema de Monitoreo.....	118
4. CALIDAD, COSTO Y SEGURIDAD	120
4.1. Introducción	120
4.2. Métricas de Calidad	120
4.2.1. Funcionalidad	121
4.2.2. Confiabilidad.....	123
4.2.3. Usabilidad	124
4.2.4. Eficiencia.....	124
4.2.4.1. Uso de CPU	125
4.2.4.2. Uso de Memoria.....	125
4.2.5. Mantenibilidad	126
4.2.6. Portabilidad	127
4.2.7. Conclusión de las Métricas de Calidad.....	128
4.3. Costos	129
4.3.1. Identificación de Funcionalidades.....	129
4.3.2. Asignación de Complejidad	130
4.3.3. Cálculo de Esfuerzo	131
4.3.4. Conclusiones sobre los Costos.....	132
4.4. Seguridad	133
4.4.1.1. Autenticación y Contraseñas.....	133
4.4.1.2. Encriptación y Tokens.....	133
4.4.1.3. Captura de Firmas	134
4.4.1.4. Validación de Peticiones	134
4.4.1.5. Copias de Seguridad y Versionado	134
4.4.1.6. Firma de Aplicaciones.....	134
4.4.1.7. Almacenamiento de Archivos con UUID.....	135
4.4.1.8. Conclusiones sobre Seguridad.....	135
5. CONCLUSIONES Y RECOMENDACIONES	137
5.1. Conclusiones	137
5.2. Recomendaciones	139
BIBLIOGRAFÍA.....	141
ANEXOS	145

ÍNDICE DE FIGURAS

Figura 1.Ciclo de vida de la metodología iterativa e incremental.....	34
Figura 2.Eschema general del sistema	44
Figura 3.Diseño de la Base de datos (1° iteración)	52
Figura 4.Conexión a la base de datos Sequelize	54
Figura 5.Vista responsiva del sistema (Móvil)	55
Figura 6.Métodos asíncronos del controlador Orden.....	56
Figura 7.Diseño de tabla con filtros y ordenación.....	56
Figura 8.Interfaz de Integración Orden – Cliente – Plan – Nodo	58
Figura 9.Diagrama de casos de uso Gestión de Ordenes de Trabajo	60
Figura 10.Captura Digital de Firma Manual.....	62
Figura 11.Utilización de Firma de Conformidad	63
Figura 12.Consultas Filtradas con Sequelize	64
Figura 13.Diseño parcial de la Base de datos (2° iteración)	70
Figura 14.Compresión de imágenes con Sharp	71
Figura 15.Diagrama de Caso de Uso Site Survey.....	77
Figura 16.Diagrama de Caso de uso Roles	77
Figura 17.Formulario dinámico Site Survey	78
Figura 18.Formulario general Site Survey	79
Figura 19.Asignación de permisos.....	80
Figura 20.Diseño Parcial de la Base de datos (3° iteración)	84
Figura 21.Interfaz Gestión de Pagos.....	85
Figura 22.Comprobante de Pago Digital	86
Figura 23.Reporte de Pagos por Periodo.....	87
Figura 24.Diagrama de Caso de Uso Finanzas.....	93
Figura 25.Diagrama de Caso de Uso Asistencias	93
Figura 26.Métodos de Notificaciones	98
Figura 27.Vinculación de WhatsApp	98
Figura 28.Selección de Impresora Térmica.....	99
Figura 29.Comprobante de Pago Impreso	99
Figura 30.Diagrama de Caso de Uso Notificaciones	103
Figura 31.Diagrama de Caso de Uso Impresión de Comprobantes	104
Figura 32.Aplicativo Escorp Windows	108
Figura 33.Instalador Escorp Windows.....	108
Figura 34.Aplicativo Escorp Android	109
Figura 35.Código raíz aplicativo Android.....	110
Figura 36.Conexión de sensores al microcontrolador	113
Figura 37.Diseño de Base de Datos Parcial (6° iteración).....	113
Figura 38.Interfaz de Monitoreo en Tiempo Real (Estructural)	114
Figura 39.Interfaz de Monitoreo en Tiempo Real (Ambiental)	115
Figura 40.Diagrama de Base de Datos Final	117
Figura 41.Modelado 3D del diseño de PBC	118
Figura 42.Uso de CPU y RAM del Servidor	126

ÍNDICE DE TABLAS

Tabla 1.Requerimiento Funcional Usuarios	46
Tabla 2.Requerimiento Funcional Empresas	47
Tabla 3.Requerimiento Funcional Planes.....	47
Tabla 4.Requerimiento Funcional Nodos	48
Tabla 5.Requerimiento Funcional Servicios	49
Tabla 6.Requerimiento Funcional Tipos de Clientes	49
Tabla 7.Requerimiento Funcional Clientes.....	50
Tabla 8.Requerimiento Funcional Órdenes de Trabajo	51
Tabla 9.Pruebas unitarias Ordenes de Trabajo.....	57
Tabla 10.Pruebas de Integración Ordenes de Trabajo.....	59
Tabla 11.Librerías de seguridad	61
Tabla 12.Requerimiento Funcional Site Survey	68
Tabla 13.Requerimiento Funcional Roles	69
Tabla 14.Pruebas Unitarias Site Survey.....	72
Tabla 15.Pruebas Unitarias Roles.....	73
Tabla 16.Pruebas de Integración Site Survey	74
Tabla 17.Pruebas de Integración Roles	75
Tabla 18.Requerimiento Funcional Asistencias.....	82
Tabla 19.Requerimiento Funcional Finanzas	83
Tabla 20.Pruebas Unitarias Finanzas	88
Tabla 21.Pruebas Unitarias Asistencias.....	89
Tabla 22.Pruebas de Integración Finanzas.....	90
Tabla 23.Pruebas de Integración Asistencias	91
Tabla 24.Requerimiento Funciona Notificaciones	96
Tabla 25.Requerimiento Funcional Impresión de Comprobantes.....	97
Tabla 26.Pruebas Unitarias Notificaciones.....	100
Tabla 27.Pruebas Unitarias Impresión de Comprobantes	101
Tabla 28.Pruebas de Integración Notificaciones por WhatsApp.....	102
Tabla 29.Pruebas de Integración Impresora Térmica	102
Tabla 30.Requerimiento Funcional Aplicación Windows	106
Tabla 31.Requerimiento Funcional Aplicación Android	106
Tabla 32.Requerimiento Funcional Monitoreo de Torres.....	111
Tabla 33.Requisitos Funcionales	121

ÍNDICE DE ECUACIONES

(1). Ecuación de Cobertura de Requisitos.....	35
(2). Ecuación de Confiabilidad	35
(3). Ecuación de Usabilidad	36
(4). Ecuación Mantenibilidad.....	37
(5). Ecuación Compatibilidad de Plataforma	37
(6). Ecuación Puntos de Función	38
(7). Ecuación Esfuerzo (HH)	38
(8). Ecuación Porcentaje de Cobertura de Requisitos.....	122
(9). Ecuación Tasa de Fallos	123
(10). Ecuación Porcentaje de Confiabilidad	123
(11). Ecuación Tasa de Error	124
(12). Ecuación Porcentaje de Usabilidad	124
(13). Ecuación Densidad de Errores	126
(14). Ecuación Porcentaje de Mantenibilidad	126
(15). Ecuación Porcentaje de Compatibilidad de Plataforma.....	127
(16). Ecuación Número de Puntos de Función.....	130
(17). Ecuación Cálculo de Esfuerzo	132
(18). Ecuación Cálculo Costo Total.....	132

RESUMEN

El presente proyecto de grado desarrolla e implementa un sistema de gestión integral para DAIFO TELECOM S.R.L., una empresa de telecomunicaciones. Bajo una metodología Iterativa e Incremental, se creó un sistema multiplataforma operativo en entornos web, escritorio y móviles, que facilita la gestión administrativa y técnica.

El sistema abarca funciones clave como gestión de órdenes de trabajo, clientes, finanzas y monitoreo en tiempo real de torres de comunicación. Este último utiliza sensores integrados para medir temperatura, humedad, lluvia y estabilidad estructural, permitiendo una respuesta ágil ante emergencias. Los dispositivos ESP32 y sensores transmiten datos por redes móviles, visualizables en tiempo real mediante WebSockets.

Incorpora notificaciones por WhatsApp para informar a los clientes sobre cuentas y pagos, además de integración con impresoras térmicas para emitir comprobantes. Este enfoque mejora la confiabilidad y eficiencia, garantizando la satisfacción del cliente.

El sistema ha sido evaluado bajo la norma ISO/IEC 25010, verificando su funcionalidad y confiabilidad. Los resultados evidencian una cobertura completa de requisitos y un bajo índice de fallos, avalando su implementación en entornos exigentes.

DAIFO TELECOM S.R.L. cuenta ahora con una herramienta robusta y escalable que optimiza sus operaciones y establece bases sólidas para futuras mejoras

Palabras Clave: Telecomunicaciones, Monitoreo, Gestión

ABSTRACT

This project develops and implements an integrated management system for DAIFO TELECOM S.R.L., a telecommunications company. Using an Iterative and Incremental methodology, a cross-platform system was created for web, desktop, and mobile environments, streamlining administrative and technical management processes.

The system includes key functions such as work order management, client administration, financial operations, and real-time monitoring of communication towers. Sensors integrated with ESP32 devices measure temperature, humidity, rainfall, and structural stability, enabling swift emergency responses. Data is transmitted over mobile networks and visualized in real time using WebSockets.

Additional features include WhatsApp notifications for client account updates and thermal printer integration for issuing receipts. This approach enhances reliability and efficiency, ensuring customer satisfaction.

The system was evaluated under ISO/IEC 25010 standards, confirming its functionality and reliability. Results show complete requirement coverage and a low failure rate, supporting its deployment in demanding environments.

DAIFO TELECOM S.R.L. now benefits from a robust, scalable tool that optimizes operations and lays the groundwork for future enhancements.

Keywords: Telecommunications, Monitoring, Management

LISTADO DE ABREVIATURAS

1. **API:** Interfaz de Programación de Aplicaciones
2. **CRUD:** Crear, Leer, Actualizar y Eliminar
3. **DBMS:** Sistema de Gestión de Bases de Datos
4. **HTTP:** Protocolo de Transferencia de Hipertexto
5. **IoT:** Internet de las Cosas
6. **JSON:** Notación de Objetos de JavaScript
7. **ORM:** Mapeo Objeto-Relacional
8. **SQL:** Lenguaje de Consulta Estructurado
9. **SSL/TLS:** Capa de Conexión Segura / Seguridad de la Capa de Transporte
10. **TCP:** Protocolo de Control de Transmisión
11. **URL:** Localizador Uniforme de Recursos
12. **UUID:** Identificador Universalmente Único
13. **Wi-Fi:** Estándar de red inalámbrica
14. **XML:** Lenguaje de Marcado Extensible
15. **FPA:** Análisis de Puntos de Función

CAPÍTULO I

MARCO PRELIMINAR

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



1. MARCO PRELIMINAR

1.1. Introducción

En un mundo donde la tecnología marca el ritmo del progreso, las empresas e instituciones buscan cada vez más implementar sistemas que les permitan aumentar su rendimiento y productividad. En el siglo pasado, era imprescindible para las empresas contar al menos con una máquina de escribir; hoy, disponemos de herramientas y sistemas mucho más eficientes, como la escritura por voz. Esto demuestra que, a pasos agigantados, estamos reemplazando los procesos manuales y repetitivos por soluciones automáticas y optimizadas.

Según Medina (2010), “la tecnología permite que el trabajador y los equipos de trabajo se desarrollen en entornos de alto nivel y realicen eficiente y efectivamente sus tareas, tanto dentro de las instalaciones de la empresa como a distancia, lo que se traduce en mejora de la productividad”.

La necesidad de las empresas de adoptar herramientas y sistemas para mejorar la productividad abarca todos los sectores. Uno de los sectores más destacados en esta era de conectividad global es el sector de telecomunicaciones, que juega un papel crucial al conectar hogares y empresas con el mundo a través de internet.

La empresa de telecomunicaciones DAIFO TELECOM S.R.L. busca constantemente optimizar y agilizar sus operaciones, registros de asistencias a clientes, pagos, nodos e inspecciones. Sus procesos actuales son repetitivos y manuales, como la elaboración de reportes, el registro de datos y la actualización de información. La ausencia de un sistema eficiente lleva al almacenamiento de registros redundantes y poco fiables, tanto en formato físico como en respaldo digital en los ordenadores de la empresa.

DAIFO TELECOM S.R.L. realiza trabajos de instalaciones, inspecciones y asistencias con personal técnico calificado, este proceso implica almacenar datos en carpetas y llenar

formularios manualmente. Tras completar una operación y recopilar la información, se debe esperar al menos 24 horas para obtener el reporte de la operación, que luego debe ser verificado y aprobado por el cliente. Este proceso prolongado resulta inadecuado para los estándares actuales de eficiencia y rapidez.

El presente proyecto tiene como objetivo desarrollar un sistema que facilite la gestión de los procesos y operaciones, proporcionando una herramienta moderna para manejar la información de manera adecuada a cada proceso y necesidad de DAIFO TELECOM S.R.L. Con un nuevo sistema, se pretende optimizar los tiempos de ejecución, mejorar el flujo de información y reducir la redundancia de datos, facilitando así una toma de decisiones más ágil y efectiva. Además, el sistema propuesto integrará funcionalidades avanzadas que permitirán una mayor visibilidad y control en las áreas operativas de la empresa.

1.2. Antecedentes

1.2.1. Antecedentes Institucionales

La empresa DAIFO TELECOM S.R.L. se especializa en el sector de las telecomunicaciones y el diseño de infraestructura de redes de datos, cableado estructurado y fibra óptica desde 2014. Nació con la idea de innovar en el área de tecnologías de redes de fibra óptica. Además, ofrece soluciones integrales para proyectos de diseño y ejecución, consultoría y servicios informáticos dirigidos a pequeñas y medianas empresas, tanto públicas como privadas.

1.2.1.1. Visión.

Constituirnos en la empresa de mayor fiabilidad, responsabilidad y profesionalismo, liderando en servicios de telecomunicaciones y soporte tecnológico con la mejor y más amplia cobertura. Nuestra presencia nacional, excelente servicio y atención personalizada están diseñados para satisfacer a la perfección las necesidades de nuestros clientes.

1.2.1.2. Misión.

Brindar servicios de alta calidad y especializados en infraestructura de telecomunicaciones que satisfagan eficientemente las necesidades del cliente, integrando excelencia en el servicio con tecnología de alta calidad y última generación.

1.2.2. Antecedentes afines al proyecto de grado

1.2.2.1. Antecedentes Internacionales.

A continuación, se describe los antecedentes internacionales que se pudieron recabar:

- (Bartolo, 2021), Desarrollo de un sistema multiplataforma basado en IONIC para el control de asistencia del personal para Empresas de Multiservicios es un sistema que emplea la arquitectura de Ionic para el control de asistencia del personal. El proyecto, realizado por Lucero Analí Laurente Bartolo, utiliza Ionic como marco de desarrollo multiplataforma y se apoya en la metodología RUP. El sistema evalúa indicadores como tiempo de espera de ingreso y salida, porcentaje de ausentismo y tardanzas, mostrando mejoras significativas en comparación con los resultados antes de la implementación.
- (Quispe y Coaguila, 2017), Propuesta de implementación de un sistema integrado de gestión en la empresa Maven Ingenieros S.A.C. es un sistema que tiene como objetivo implementar un sistema integrado de gestión para controlar la calidad, medio ambiente y seguridad y salud en el trabajo. El proyecto, realizado por Ana Ruby Quispe Vilca y Giancarlo Fabricio Coaguila Bolivar en la Universidad Nacional de San Agustín de Arequipa, abarca desde el planteamiento del problema hasta la metodología de implementación y análisis de resultados, siguiendo normas ISO 9001, ISO 14001 y OHSAS 18001.

- (Lazo y Huamán, 2019), Software multiplataforma para restaurantes utilizando tecnologías híbridas Node.js, Electron.js y React Native es un sistema diseñado para gestionar las operaciones de un restaurante a través de múltiples plataformas, incluyendo escritorio, web y móvil. El software emplea Node.js y Electron para desarrollar la aplicación de escritorio que controla los procesos administrativos del restaurante, como el registro de productos, sucursales, mesas, y la generación de documentos como recibos y tickets. Además, esta aplicación de escritorio se despliega en el entorno web, permitiendo a los administradores gestionar las operaciones del restaurante de forma remota. Para la automatización de la toma de pedidos de los comensales, se implementa una aplicación móvil utilizando tecnologías híbridas.

1.2.2.2. Antecedentes Nacionales.

A continuación, se describen los antecedentes nacionales que se pudieron recabar. En los últimos años, el país ha mostrado un interés creciente en el tema a través de iniciativas académicas y políticas gubernamentales.

- (Salazar, 2011), Sistema para la Superintendencia de Telecomunicaciones (SITTEL). El proyecto aborda la regulación y supervisión del sector telecomunicaciones en Bolivia, beneficiando tanto a operadores como usuarios. Utilizando las metodologías RUP y OOHDM, el sistema fue desarrollado en cinco capítulos que incluyen análisis, diseño, implementación y evaluación de la calidad del software. El trabajo concluye con recomendaciones para mejorar los procesos en SITTEL.
- (Cayo, 2015), Sistema web de registro, control y seguimiento de conexiones de redes e internet: Caso Empresa Boliviana de Telecomunicaciones Planta Externa. Este proyecto automatiza procesos internos y centraliza la información en una base de datos relacional para mejorar la eficiencia en E.B.T.P.E.X. Se utilizó la metodología Ágil

SCRUM y UWE para el desarrollo, empleando PHP, MySQL, y Apache como tecnologías clave.

- (Huayhua, 2010), Desarrollo e implementación de un sistema integrado de control administrativo, financiero y de proyectos para ELEKTROPACHA Ltda.. El proyecto busca mejorar la gestión empresarial mediante una aplicación de Inteligencia de Negocios que soporte la toma de decisiones en diversas áreas clave. Desarrollado con Delphi 7.0 y SQL Server 2005, utilizando la metodología RUP y UML, el sistema se implementa bajo la norma ISO 9001:2000, abarcando Administración, Producción y Comercialización.

1.3. Planteamiento del problema

1.3.1. Problema principal

La empresa DAIFO TELECOM S.R.L. enfrenta un problema crítico en la gestión de su información, lo que resulta en la realización de tareas de manera manual y repetitiva. Esta situación incrementa el tiempo de ejecución y afecta tanto la eficiencia como la precisión de los datos. Además, limita la capacidad de la empresa para adaptarse a un entorno competitivo y optimizar sus recursos, repercutiendo negativamente en su rendimiento general.

1.3.2. Problemas secundarios

- Los registros manuales provocan inconsistencia en la información tanto en el llenado de registros como en su revisión.
- La información solo disponible en la oficina central retrasa a los técnicos en campo.
- El Monitoreo de proyectos manual aumenta el tiempo de ejecución causando retrasos.
- Los procesos manuales como el envío de reportes o asignación de tareas reducen la eficiencia y elevan los costos operativos.

1.3.3. Formulación del problema

¿De qué manera se puede mejorar la gestión de datos y operaciones en DAIFO TELECOM S.R.L., específicamente en las áreas de supervisión y trabajos operativos del sector de telecomunicaciones?

1.4. Objetivos

1.4.1. Objetivo General

Implementar un sistema multiplataforma en DAIFO TELECOM S.R.L. para optimizar la gestión de sus datos y operaciones.

1.4.2. Objetivos Específicos

- Establecer procedimientos estandarizados para asegurar la consistencia en la información.
- Habilitar el acceso a la información desde cualquier ubicación para facilitar el trabajo de los técnicos en campo.
- Centralizar la gestión de proyectos para mejorar el seguimiento y reducir el tiempo de ejecución.
- Optimizar los procesos para aumentar la eficiencia y reducir los costos operativos.

1.5. Justificación

1.5.1. Justificación Técnica

El proyecto se justifica técnicamente porque DAIFO TELECOM S.R.L. cuenta con la infraestructura tecnológica adecuada para soportar un sistema de gestión. Se utilizarán tecnologías avanzadas como Node.js, Electron, Apache Cordova y Vue.js para desarrollar un sistema multiplataforma, accesible desde Windows, web y dispositivos Android. Esto

garantizará una integración fluida y un acceso optimizado a la información, mejorando la eficiencia operativa y la toma de decisiones.

1.5.2. Justificación Económica

El proyecto aprovechará tecnologías de software libre como MySQL, Vue.js, Apache Cordova y Electron, eliminando costos asociados a licencias. Al utilizar el servidor interno de la institución, se evitan gastos adicionales en hardware o alojamiento externo. Además, la automatización de procesos reduce tiempos en tareas manuales, optimizando recursos y permitiendo al personal enfocarse en actividades estratégicas, garantizando así una solución económica y sostenible. Cabe destacar que los gastos en hardware son mínimos comparados con los costos de transporte y logística que actualmente se generan para realizar el monitoreo de las torres, lo que convierte al proyecto en una alternativa altamente eficiente y rentable.

1.5.3. Justificación Social

El sistema se justifica socialmente al mejorar las condiciones laborales para el personal de DAIFO TELECOM S.R.L. Al automatizar procesos y centralizar la información, se facilitará el trabajo de los empleados, reduciendo el riesgo de errores y agilizando las tareas. Esto contribuirá a un entorno laboral más eficiente y menos estresante, además de proporcionar a los gerentes información precisa y oportuna para una mejor toma de decisiones.

1.6. Metodología

1.6.1. Metodología de Desarrollo Iterativo e Incremental

La Metodología de Desarrollo Iterativo e Incremental es un enfoque ágil para el desarrollo de software que se centra en la creación y mejora continua del sistema a través de ciclos repetidos o iteraciones. Cada iteración proporciona una versión funcional del sistema con nuevas características y mejoras, permitiendo ajustes basados en la retroalimentación

recibida durante el desarrollo. Este enfoque facilita la adaptación a cambios y asegura una evolución constante del sistema (Sommerville, 2011).

1.6.1.1. Características de la Metodología Iterativa e Incremental.

El enfoque iterativo e incremental se define por varias características esenciales que lo convierten en una opción eficaz para proyectos de desarrollo:

- Desarrollo en ciclos repetitivos: Cada iteración se centra en un conjunto específico de objetivos, permitiendo que el sistema crezca de manera ordenada y estructurada.
- Construcción progresiva: Las nuevas funcionalidades o mejoras se añaden en cada iteración, consolidando el trabajo previamente realizado y garantizando un avance continuo.
- Ajustes basados en retroalimentación: La interacción constante con los usuarios asegura que el sistema se refine continuamente, adaptándose a las necesidades emergentes.
- Flexibilidad frente a cambios: Este enfoque permite manejar con efectividad modificaciones en los requisitos.

1.6.1.2. Principales Aspectos de la Metodología Iterativa e Incremental.

La metodología se organiza alrededor de aspectos fundamentales que aseguran su efectividad:

- Ciclos de desarrollo cortos: El trabajo se divide en iteraciones breves y enfocadas, cada una de las cuales genera un incremento funcional del sistema.
- Integración de retroalimentación: La evaluación al final de cada iteración contribuye al perfeccionamiento del sistema en cada nueva fase.
- Evolución funcional continua: La incorporación constante de mejoras permite que el sistema crezca y se ajuste progresivamente a los objetivos del proyecto.

- Adaptación permanente: Los cambios y ajustes necesarios se implementan de forma ágil, reduciendo el riesgo de desviaciones significativas.

1.6.1.3. Fases del Desarrollo Iterativo e Incremental.

El proceso iterativo e incremental se estructura en varias fases claves:

- Planificación de iteraciones: En esta etapa, se identifican las metas y las funcionalidades que se desarrollarán en cada ciclo.
- Desarrollo y pruebas: Se implementan las características definidas y se realizan pruebas para garantizar la calidad de cada iteración.
- Revisión continua: Al final de cada ciclo, se evalúan los resultados y se ajusta el enfoque para el siguiente.
- Publicación progresiva: Las nuevas versiones del sistema, enriquecidas con las funcionalidades añadidas, se entregan de forma constante al usuario.

1.6.2. Métricas de Calidad

1.6.2.1. Funcionalidad.

Evalúa cómo el sistema cumple con los requisitos específicos del negocio, asegurando que las funciones esenciales se realizan con precisión y cubriendo los casos de uso definidos.

1.6.2.2. Confiabilidad.

Mide el tiempo durante el cual el sistema está operativo y accesible para los usuarios, considerando su capacidad para recuperarse de fallos de manera eficiente.

1.6.2.3. Usabilidad.

Determina cuán intuitivo y accesible es el sistema para los usuarios finales, evaluando la claridad de la interfaz y la facilidad de aprendizaje.

1.6.2.4. Eficiencia.

Analiza cómo el sistema gestiona los recursos de hardware, como CPU y memoria, asegurando un rendimiento óptimo y evitando sobrecargas innecesarias.

1.6.2.5. Mantenibilidad.

Valora la facilidad para realizar correcciones, mejoras o adaptaciones en el sistema, evaluando la claridad del código y la modularidad del diseño.

1.6.2.6. Portabilidad.

Examina la capacidad del sistema para funcionar en múltiples plataformas y dispositivos sin requerir modificaciones significativas, asegurando su flexibilidad y escalabilidad.

1.6.3. Costo

Para la estimación de costos en el proyecto, se utilizará el Modelo de Análisis de Puntos de Función (FPA). El FPA es una metodología que mide el tamaño del software en términos de las funcionalidades que proporciona al usuario final. Este enfoque se basa en el análisis detallado de los casos de uso y las funcionalidades requeridas, y estima el esfuerzo y los costos asociados a partir de estos puntos de función.

El proceso de estimación con FPA incluye los siguientes pasos:

- **Identificación de Funcionalidades:** Se identifican y se cuentan los puntos de función basados en los requerimientos del sistema, incluyendo entradas, salidas, archivos internos y consultas.
- **Asignación de Complejidad:** Se clasifica la complejidad de cada funcionalidad para ajustar la estimación de acuerdo con el nivel de dificultad.

- **Cálculo de Esfuerzo:** Utilizando las fórmulas del FPA, se calcula el esfuerzo necesario para desarrollar el sistema en términos de horas-hombre o costo total.

Este método permite una evaluación precisa y detallada de los costos, basada en las funcionalidades específicas requeridas por el proyecto, y facilita la planificación y control de costos durante el ciclo de vida del desarrollo.

1.6.4. Seguridad

- **Autenticación y Contraseñas:** Contraseñas configurables para cada usuario, cifradas con bcrypt para proteger la información sensible.
- **Encriptación y Tokens:** Datos de autenticación protegidos con JWT (JSON Web Tokens), generados y verificados usando jsonwebtoken. Tokens almacenados en localStorage para persistencia de sesión.
- **Captura de Firmas:** Uso de VueSignature para firmas electrónicas con marca de tiempo, garantizando la validez y rastreabilidad.
- **Validación de Peticiones:** Validación del origen de cada petición en el servidor para asegurar que solo usuarios autorizados interactúen con el sistema.
- **Copias de Seguridad:** Respaldo regular de datos en un servidor dedicado como Plesk, para proteger contra fallos de hardware o pérdidas de datos.
- **Firma de Aplicaciones:** Compilación y firma digital de archivos APK usando una clave privada almacenada localmente en el entorno de desarrollo, asegurando la autenticidad e integridad del archivo.

1.6.5. Pruebas de Software

1.6.5.1. Pruebas Unitarias.

Verificación de cada componente individual del sistema para asegurar que cada unidad de código funcione correctamente de forma aislada.

1.6.5.2. Pruebas de Integración.

Evaluación de la interacción entre diferentes módulos del sistema para garantizar que trabajen juntos de manera efectiva.

1.6.5.3. Pruebas Funcionalidad.

Evaluación del sistema por parte de los usuarios finales para confirmar que cumple con sus necesidades y expectativas.

1.7. Herramientas

1.7.1. Herramientas de Software

- Vue.js: Framework progresivo de JavaScript para construir interfaces de usuario y aplicaciones web. Facilita el desarrollo de aplicaciones interactivas y reactivas con un enfoque en la componente y la reactividad (Evan You, 2014).
- Node.js: Entorno de ejecución para JavaScript del lado del servidor, que permite el desarrollo de aplicaciones de red escalables y eficientes, utilizando un modelo de I/O no bloqueante (OpenJS Foundation, 2023).
- MySQL: Sistema de gestión de bases de datos relacional de código abierto, ampliamente utilizado para la administración y almacenamiento de datos, compatible con diversas aplicaciones web y móviles (MySQL AB, 2023).
- Electron: Framework para construir aplicaciones de escritorio multiplataforma con tecnologías web como JavaScript, HTML y CSS. Permite crear aplicaciones nativas para Windows, macOS y Linux (GitHub, 2023).
- Apache Cordova: Plataforma de desarrollo de aplicaciones móviles híbridas que utiliza HTML5, CSS3 y JavaScript para crear aplicaciones que se ejecutan en múltiples sistemas operativos móviles desde una única base de código (Apache Software Foundation, 2023).

- Vuetify: Biblioteca de componentes de interfaz de usuario basada en Material Design para Vue.js. Proporciona una colección de componentes visuales que ayudan a crear aplicaciones web con una apariencia consistente y moderna (Vuetify Team, 2023).
- Nodemailer: Módulo para Node.js utilizado para enviar correos electrónicos desde una aplicación. Ofrece soporte para diversos servicios de correo y métodos de envío (Nodemailer Team, 2023).
- Puppeteer: Biblioteca de Node.js que proporciona una API de alto nivel para controlar navegadores Chromium y Chrome, permitiendo la generación de capturas de pantalla, PDFs, y la automatización de pruebas de interfaz (Google, 2023).
- VueSignature: Plugin para Vue.js que permite capturar firmas electrónicas en aplicaciones web, proporcionando una interfaz para registrar firmas con precisión y facilidad (VueSignature, 2023).
- Sharp: Biblioteca de procesamiento de imágenes para Node.js que permite la manipulación eficiente y la optimización de imágenes, incluyendo la conversión y redimensionamiento (Lovell, 2023).
- Leaflet: Biblioteca de JavaScript para la creación de mapas interactivos. Ofrece una interfaz sencilla para agregar y manipular mapas en aplicaciones web, con soporte para diversas capas y marcadores (Leaflet, 2023).

1.7.2. Herramientas de Hardware

- ESP32: microcontrolador de bajo costo y bajo consumo que integra Wi-Fi y Bluetooth BLE (Bluetooth Low Energy), haciéndolo una opción popular para el desarrollo de aplicaciones en el Internet de las Cosas (IoT). Desarrollado por Espressif Systems, ESP32 es conocido por su rendimiento robusto, sus amplias capacidades de conectividad y su arquitectura versátil, que soporta una variedad de aplicaciones desde

automatización del hogar hasta soluciones industriales complejas (Espressif Systems, 2023).

- BME280: Sensor ambiental que combina medición de temperatura, presión barométrica y humedad en un solo dispositivo compacto. Fabricado por Bosch Sensortec, el BME280 es ideal para aplicaciones de monitoreo ambiental en proyectos IoT debido a su alta precisión y bajo consumo de energía (Llamas, 2020).
- MPU6050: Módulo que integra un acelerómetro y un giroscopio de 6 ejes, diseñado por InvenSense. Este sensor es ampliamente utilizado en proyectos que requieren medición de movimiento y orientación, como sistemas de navegación, estabilización y aplicaciones de realidad aumentada, debido a su precisión y capacidad de procesamiento (Naylamp Mechatronics, 2018).
- Pantalla OLED: Pantalla basada en Diodos Orgánicos de Emisión de Luz, conocida por su alta eficiencia energética, excelente contraste y amplios ángulos de visión. Es ideal para proyectos embebidos que requieren mostrar información en espacios reducidos y con bajo consumo energético (Cabo Tinoso, 2023).
- Sensor de lluvia YL-83: Sensor utilizado para detectar la presencia de agua en forma líquida. Emite una señal digital cuando se detecta lluvia o humedad en su superficie, siendo útil en aplicaciones de monitoreo ambiental y automatización, como sistemas de riego o estaciones meteorológicas (SpainLabs, 2017).

1.8. Límites y alcances

1.8.1. Límites

- El sistema será solamente para el uso del personal de la empresa
- El sistema no gestionara al personal dependiente de la empresa ni sus obligaciones y responsabilidades
- El sistema no podrá funcionar sin conectividad a internet

1.8.2. Alcances

Los alcances son:

- Establecer procedimientos estandarizados para asegurar la consistencia en la información mediante la automatización de reportes, impresión de comprobantes con QR y registro de eventos en el sistema.
- Habilitar el acceso a la información desde cualquier ubicación para facilitar el trabajo de los técnicos en campo, utilizando la visualización en tiempo real y el envío de notificaciones por WhatsApp.
- Centralizar la gestión de proyectos para mejorar el seguimiento y reducir el tiempo de ejecución con módulos como Órdenes de Trabajo, Finanzas y Site Survey.
- Optimizar los procesos para aumentar la eficiencia y reducir los costos operativos a través de la captura digital de firmas, impresión de comprobantes con QR y automatización de comunicaciones.

1.9. Aportes

El avance tecnológico actual demanda que las instituciones y empresas se adapten y aprovechen nuevas soluciones digitales. El presente proyecto ofrece los siguientes aportes:

- Implementación de un entorno sistematizado para la gestión de clientes, órdenes de trabajo, asistencias, finanzas y otros aspectos críticos, mejorando la eficiencia operativa de la empresa.
- Adopción de tecnologías modernas como Vue.js, Node.js y otros, que aseguran la utilización de herramientas avanzadas para el desarrollo de software.
- Generación de reportes precisos en formato PDF, visualización de gráficas clave y emisión de comprobantes de pago con QR, facilitando el análisis de datos y la toma de decisiones.

- Centralización de la información en una única base de datos, optimizando el acceso y la gestión de datos.
- Mejora en la comunicación mediante notificaciones por WhatsApp y envíos de correos electrónicos con documentos adjuntos, fortaleciendo la interacción con clientes y usuarios.
- Agilización de procesos manuales mediante la captura y procesamiento digital de firmas, y la automatización de tareas rutinarias, lo que reduce tiempos de atención y aumenta la productividad.
- Monitoreo en tiempo real del estado estructural de las torres de comunicación.

CAPÍTULO II

MARCO TEÓRICO

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



2. MARCO TEÓRICO

2.1. Introducción

Este capítulo tiene como propósito establecer las definiciones y conceptos fundamentales para comprender los procesos involucrados en el desarrollo de este Proyecto de Grado. Es imprescindible tener un marco teórico claro que sustente la investigación y sirva como guía para el diseño e implementación del sistema propuesto. Para ello, se abordarán los términos técnicos y metodológicos clave que se utilizarán a lo largo del proyecto, asegurando que exista una comprensión común de los elementos centrales.

Se comenzará con una explicación detallada de las teorías y modelos que guiarán el diseño del sistema, proporcionando una base conceptual sólida. En particular, se destacará la importancia de la gestión de información en el contexto de la empresa de telecomunicaciones, explorando cómo la tecnología puede ser un catalizador para la optimización de los procesos operativos y administrativos. Asimismo, se revisarán las estrategias más efectivas para integrar tecnologías multiplataforma y cómo estas permiten una mayor accesibilidad y eficiencia en la ejecución de operaciones en campo y en la administración interna.

Un enfoque adicional será la estandarización de procesos, la cual resulta esencial para la mejora continua y el fortalecimiento de la competitividad. La implementación de un sistema de información eficiente implica no solo la integración de nuevas tecnologías, sino también la creación de procedimientos que aseguren la calidad, consistencia y actualización de la información gestionada. De esta manera, se proporcionará una estructura organizativa que permita a la empresa adaptarse a las demandas cambiantes del mercado y aprovechar las oportunidades tecnológicas.

Asimismo, se abordarán los aspectos técnicos fundamentales, incluyendo la arquitectura de software y la seguridad de la información. La arquitectura del sistema se diseñará bajo principios de modularidad y escalabilidad, garantizando la capacidad de

crecimiento del sistema en función de los futuros requerimientos de la empresa. En cuanto a la seguridad de la información, se discutirán los mecanismos y protocolos necesarios para proteger los datos sensibles, respetando tanto la privacidad de los usuarios como la integridad de la información manejada. Se hará especial hincapié en la implementación de estándares de seguridad internacionales y buenas prácticas de desarrollo de software.

Otro componente clave de este capítulo será la revisión de las normativas y estándares relevantes para la protección de datos y la gestión de la información en el ámbito de las telecomunicaciones. La implementación de un sistema de gestión de operaciones en una empresa que maneja datos de clientes y operaciones en campo debe cumplir con requisitos legales y éticos, garantizando un manejo seguro y transparente de la información. De este modo, se asegura la confianza de los usuarios en el sistema y la continuidad de los servicios sin vulneraciones a la seguridad.

Finalmente, se presentará una visión general de las herramientas y tecnologías que se emplearán en el desarrollo del sistema. Cada herramienta será explicada en detalle, resaltando su función específica dentro del proyecto y su contribución a la consecución de los objetivos planteados. Se analizarán las tecnologías empleadas para el desarrollo frontend y backend, así como las herramientas de soporte para la gestión de bases de datos y la automatización de procesos.

Con esta base conceptual y técnica, se facilitará la comprensión de las decisiones de diseño y desarrollo que se tomarán durante el proyecto, asegurando la alineación con los objetivos de mejora operativa y optimización de procesos en la empresa. De esta forma, se proporcionará un marco teórico que servirá de referencia para las etapas posteriores del proyecto, guiando el desarrollo hacia una solución integral y efectiva.

2.2. Sistema de Información

2.2.1. Definición de Sistema de Información

Los sistemas de información son estructuras compuestas por hardware, software, datos y procedimientos, junto con personal cualificado, cuyo propósito es gestionar y procesar datos para generar información útil. Esta información apoya la toma de decisiones dentro de una organización, optimizando sus operaciones y facilitando el cumplimiento de sus objetivos estratégicos (Laudon y Laudon, 2021).

En el sector de telecomunicaciones, la implementación de sistemas de información es fundamental para administrar de manera eficiente los recursos técnicos y humanos, automatizar procesos operativos, y mejorar la interacción con los clientes mediante herramientas digitales de comunicación y gestión (Gómez, 2021).

2.2.2. Características de un Sistema Multiplataforma

Los sistemas multiplataforma son aquellos diseñados para funcionar de manera consistente en diferentes dispositivos y sistemas operativos, garantizando la accesibilidad y uniformidad de la experiencia del usuario. En el ámbito de telecomunicaciones, esta característica es fundamental para que los técnicos en campo puedan acceder a la información en tiempo real desde dispositivos móviles, como smartphones o tablets, y para que los administradores en la oficina tengan acceso simultáneo desde sus estaciones de trabajo (Tanenbaum y Van Steen, 2017).

La flexibilidad que ofrecen estos sistemas permite a las empresas implementar soluciones dinámicas que se adapten a las diversas necesidades de los usuarios y de la infraestructura de la empresa, asegurando la continuidad de las operaciones y la eficiencia en la gestión de los recursos (Rojas y Pérez, 2018).

2.2.3. Gestión de Operaciones en Telecomunicaciones

La gestión de operaciones en el sector de telecomunicaciones se refiere a la planificación, implementación, monitoreo y mantenimiento de los servicios e infraestructuras necesarias para proporcionar conectividad y acceso a internet a los usuarios finales. Esta gestión implica la administración de recursos como nodos de red, equipos de conexión y personal técnico, asegurando que los servicios sean ininterrumpidos y de alta calidad (Heizer, Render y Munson, 2021).

Los sistemas de información proporcionan herramientas para gestionar eficazmente estas operaciones, facilitando la toma de decisiones basada en datos actualizados y ofreciendo visibilidad en tiempo real del estado de la infraestructura y las tareas operativas (Martínez y Sánchez, 2019).

2.2.4. Operaciones Específicas en el Sector de Telecomunicaciones

En el contexto de telecomunicaciones, las operaciones específicas abarcan una serie de actividades críticas que garantizan la provisión de servicios de telecomunicaciones a los clientes. Estas operaciones incluyen:

- **Trabajos en Campo:** Actividades como instalaciones y reparaciones de infraestructuras de conexión, donde los técnicos deben realizar configuraciones y ajustes en el sitio. La eficiencia de estas actividades se mejora significativamente mediante el acceso a herramientas digitales que permiten visualizar diagramas de red, consultar historiales de servicio y registrar datos en tiempo real (González y Flores, 2020).
- **Gestión de Asistencias:** Los sistemas de información permiten gestionar las asistencias técnicas de forma eficiente mediante la asignación automática de horarios y tareas a los técnicos. Esto se logra ajustando los calendarios en función de la demanda de servicio o de las emergencias que puedan surgir, optimizando así el uso de los recursos humanos y mejorando la atención al cliente (Pérez y García, 2021).

- Inspecciones y Mantenimientos: La ejecución de inspecciones y tareas de mantenimiento preventivo o correctivo requiere una planificación adecuada y la capacidad de registrar información detallada en el lugar de trabajo. Las aplicaciones móviles integradas en el sistema de gestión permiten a los técnicos registrar y transmitir información desde el sitio, reduciendo el tiempo de procesamiento y mejorando la precisión y disponibilidad de los registros (Patterson, Hennessy y Arpaci-Dusseau, 2018).

2.3. Herramientas de Desarrollo: Frontend y Backend

Las herramientas de desarrollo frontend y backend permiten construir aplicaciones robustas y escalables, facilitando la implementación de interfaces interactivas y la gestión eficiente de la lógica de negocio y los datos. En el desarrollo de sistemas para DAIFO TELECOM S.R.L., se utilizará una combinación de tecnologías modernas que garantizan una experiencia de usuario óptima y una infraestructura backend confiable.

2.3.1. *Vue.js*

Vue.js es un framework progresivo de JavaScript orientado al desarrollo de interfaces de usuario dinámicas. A través de su arquitectura basada en componentes reutilizables y su sistema de reactividad, Vue.js permite a los desarrolladores crear aplicaciones interactivas y responsivas con eficiencia. Su integración con tecnologías backend lo convierte en una solución ideal para proyectos de telecomunicaciones, ya que facilita la implementación de dashboards, sistemas de monitoreo en tiempo real y interfaces intuitivas para técnicos y administradores (You, 2014). Además, Vue.js proporciona un soporte robusto para el manejo del estado de la aplicación, la manipulación dinámica del DOM y el enrutamiento de vistas (Rodríguez y Ramírez, 2019).

2.3.2. Node.js

Node.js es un entorno de ejecución basado en JavaScript que utiliza un modelo de entrada/salida no bloqueante y orientado a eventos, lo que lo hace ideal para aplicaciones de red escalables que manejan múltiples conexiones simultáneamente. En proyectos de telecomunicaciones, esta arquitectura permite gestionar grandes volúmenes de datos en tiempo real, facilitando la recolección y el procesamiento de información operativa crítica, como registros de eventos y seguimiento de actividades en campo (González y Pérez, 2020). Además, la capacidad de Node.js para trabajar con WebSockets proporciona un canal bidireccional para la comunicación en tiempo real entre el cliente y el servidor, crucial para aplicaciones que requieren actualizaciones en vivo (Núñez, 2020).

2.3.3. MySQL

MySQL es un sistema de gestión de bases de datos relacional (RDBMS) de código abierto ampliamente utilizado debido a su escalabilidad, fiabilidad y compatibilidad con diversas plataformas de desarrollo. Su capacidad para manejar grandes volúmenes de datos y garantizar la consistencia de los mismos es clave en el entorno de telecomunicaciones, donde la información debe ser centralizada y accesible en tiempo real para la toma de decisiones (MySQL AB, 2023).

2.3.4. Vuetify

Vuetify es una biblioteca de componentes de interfaz de usuario basada en Material Design, diseñada para ser utilizada junto con Vue.js. Ofrece una amplia gama de componentes predefinidos que ayudan a crear aplicaciones web visualmente coherentes y estéticamente atractivas. En el contexto de aplicaciones de telecomunicaciones, Vuetify facilita la creación de formularios, gráficos y paneles de control que cumplen con altos estándares de diseño y usabilidad (Vuetify Team, 2023).

2.3.5. Leaflet

Leaflet es una biblioteca de JavaScript de código abierto que facilita la creación de mapas interactivos en aplicaciones web. Es especialmente útil en proyectos de telecomunicaciones que requieren la visualización de ubicaciones geográficas, como nodos de red, clientes y áreas de cobertura. Leaflet permite agregar y manipular marcadores, capas y rutas, proporcionando a los técnicos y administradores una herramienta visual poderosa para planificar y gestionar operaciones de campo (Fernández y Díaz, 2018).

2.3.6. WebSockets

WebSockets es un protocolo que permite la comunicación bidireccional y en tiempo real entre un cliente y un servidor a través de una única conexión TCP. En el contexto de telecomunicaciones, WebSockets se utiliza para transmitir actualizaciones instantáneas y sincronizar información entre diferentes usuarios o dispositivos, garantizando la eficiencia en el intercambio de datos críticos, como notificaciones de eventos y reportes en vivo (Núñez, 2020).

2.3.7. Sequelize

Sequelize es un ORM para Node.js que proporciona una interfaz intuitiva para interactuar con bases de datos relacionales como MySQL. Facilita la gestión de modelos de datos y simplifica la creación de consultas complejas, lo cual es beneficioso en aplicaciones de telecomunicaciones que deben manejar y manipular grandes volúmenes de datos de manera estructurada y eficiente (Soto y Villalobos, 2021).

2.3.8. Sharp

Sharp es una biblioteca de procesamiento de imágenes para Node.js que permite la manipulación eficiente y optimización de imágenes. En proyectos de telecomunicaciones, Sharp se utiliza para la conversión y redimensionamiento de imágenes de inspección de

campo, documentos visuales y reportes fotográficos, mejorando la calidad y reduciendo el tamaño de los archivos para un almacenamiento y transmisión eficientes (Lovell, 2023).

2.3.9. Nanoid

Nanoid es una biblioteca para generar identificadores únicos cortos y seguros. En aplicaciones de telecomunicaciones, la generación de IDs únicos es fundamental para identificar de manera segura y precisa recursos críticos como nodos de red, usuarios, registros de operaciones y órdenes de trabajo, evitando conflictos de datos y duplicados (García y Luna, 2019).

2.4. Herramientas de Desarrollo: Aplicaciones Android y Windows

Las herramientas de desarrollo multiplataforma permiten crear aplicaciones nativas y móviles que operan de manera eficiente en diferentes entornos y sistemas operativos. En el caso de DAIFO TELECOM S.R.L., se han utilizado tecnologías específicas para desarrollar soluciones que faciliten la gestión y operación de los técnicos en campo y los administradores en la oficina. Las dos tecnologías clave son Apache Cordova y Electron, ambas seleccionadas por su flexibilidad y capacidad para crear aplicaciones con una única base de código que funcionan en múltiples plataformas.

2.4.1. Apache Cordova

Apache Cordova es una plataforma de desarrollo de aplicaciones móviles híbridas que permite a los desarrolladores crear aplicaciones utilizando tecnologías web estándar como HTML5, CSS3 y JavaScript. La principal ventaja de Cordova es que proporciona un enfoque de desarrollo unificado para múltiples sistemas operativos móviles, eliminando la necesidad de escribir código nativo para cada plataforma específica (Apache Software Foundation, 2023). En el caso de DAIFO TELECOM S.R.L., la utilización de Cordova permite desarrollar

aplicaciones que funcionen tanto en dispositivos Android como en otros sistemas operativos, facilitando la portabilidad y reduciendo el tiempo de desarrollo.

2.4.2. Android

Android es un sistema operativo basado en Linux, diseñado principalmente para dispositivos móviles con pantallas táctiles, como smartphones y tablets. Desarrollado por Google, Android es uno de los sistemas operativos más populares a nivel mundial, caracterizándose por su flexibilidad y la gran cantidad de aplicaciones disponibles en Google Play Store. Su estructura abierta permite a los desarrolladores aprovechar características avanzadas de hardware y software, lo cual es fundamental para las aplicaciones de gestión de telecomunicaciones que requieren acceso a funcionalidades del dispositivo, como GPS y cámara (Rodríguez y Pérez, 2018).

2.4.3. Aplicativos

Aplicativos se refiere a los programas informáticos diseñados para realizar tareas específicas en un dispositivo, ya sea una computadora o un dispositivo móvil. Los aplicativos pueden incluir desde simples herramientas de productividad hasta complejos sistemas de gestión de datos (González y Flores, 2020). En el contexto de DAIFO TELECOM S.R.L., los aplicativos desarrollados permiten a los técnicos acceder a la información de clientes, gestionar órdenes de trabajo y registrar actividades en tiempo real desde sus dispositivos móviles.

2.4.4. Electron

Electron es un framework que permite a los desarrolladores crear aplicaciones de escritorio multiplataforma utilizando tecnologías web como JavaScript, HTML y CSS. Electron combina el motor de renderizado de Chromium con el entorno de ejecución de Node.js, permitiendo la creación de aplicaciones de escritorio que se integran profundamente con el

sistema operativo, ofreciendo acceso a APIs nativas y funcionalidades avanzadas (GitHub, 2023). Para DAIFO TELECOM S.R.L., Electron es la herramienta ideal para desarrollar aplicaciones de gestión de escritorio que requieren funciones específicas del sistema operativo, como la impresión de documentos, el manejo de archivos y la integración con bases de datos locales.

2.4.5. Windows

Windows es un sistema operativo desarrollado por Microsoft y utilizado ampliamente en computadoras de escritorio y portátiles. Windows es conocido por su interfaz gráfica de usuario y su amplia compatibilidad con aplicaciones de software de terceros, lo que lo convierte en el sistema operativo preferido para muchas aplicaciones de negocios y gestión. Windows ofrece un entorno estable y accesible para el desarrollo de herramientas administrativas, permitiendo una fácil integración con tecnologías como Electron y bases de datos relacionales (Fernández y Ramírez, 2020).

2.4.6. Aplicaciones de Escritorio

Aplicaciones de Escritorio son programas informáticos que se ejecutan en sistemas operativos de computadoras personales. Estas aplicaciones suelen ofrecer un conjunto de funcionalidades más robustas en comparación con las aplicaciones móviles, ya que pueden acceder a recursos más avanzados del sistema y ofrecer interfaces de usuario más complejas. Las aplicaciones de escritorio desarrolladas con Electron permiten a los administradores gestionar datos de clientes, realizar análisis de operaciones y generar reportes de forma eficiente (Rojas y Ramírez, 2019).

2.5. Herramientas de desarrollo: Hardware

2.5.1. ESP32

El ESP32 es un microcontrolador desarrollado por Espressif Systems que se caracteriza por su alta eficiencia y bajo costo, siendo una de las opciones más populares en proyectos de Internet de las Cosas (IoT). Este dispositivo integra conectividad Wi-Fi y Bluetooth BLE (Bluetooth Low Energy), lo que permite que los desarrolladores implementen soluciones versátiles y eficientes en una variedad de entornos.

Entre sus principales ventajas destacan su capacidad de manejar múltiples conexiones simultáneamente, su arquitectura de doble núcleo que permite la ejecución de procesos paralelos, y su soporte para una amplia gama de sensores y periféricos. Esto lo hace ideal para proyectos de automatización del hogar, monitoreo ambiental y soluciones industriales complejas. Gracias a su comunidad activa, los desarrolladores cuentan con una amplia documentación, ejemplos de código y bibliotecas que facilitan su implementación (Espressif Systems, 2023).

2.5.2. BME280

El BME280, fabricado por Bosch Sensortec, es un sensor ambiental que combina tres funciones esenciales en un solo dispositivo: medición de temperatura, presión barométrica y humedad relativa. Este sensor es ampliamente reconocido por su precisión y su bajo consumo energético, características que lo convierten en una solución ideal para proyectos IoT, estaciones meteorológicas y sistemas de monitoreo ambiental. El sensor utiliza tecnología avanzada para garantizar mediciones precisas incluso en condiciones ambientales cambiantes, lo que es crucial para aplicaciones que requieren datos en tiempo real y alta fiabilidad. Además, su diseño compacto permite integrarlo fácilmente en sistemas embebidos (Llamas, 2020).

2.5.3. MPU6050

El MPU6050 es un módulo que integra un acelerómetro y un giroscopio de 6 ejes en un solo dispositivo. Este sensor, desarrollado por InvenSense, es ideal para aplicaciones que requieren monitoreo de movimiento y orientación. Es ampliamente utilizado en drones, robots, sistemas de estabilización de cámaras y dispositivos portátiles.

La capacidad del MPU6050 para medir la aceleración y la velocidad angular lo convierte en una herramienta esencial para proyectos que involucran navegación, estabilización y reconocimiento de gestos. Además, cuenta con un procesador de movimiento digital (DMP) integrado, que permite realizar cálculos complejos directamente en el sensor, reduciendo la carga del microcontrolador principal (Naylamp Mechatronics, 2018).

2.5.4. OLED

Las pantallas OLED (Diodos Orgánicos de Emisión de Luz) se destacan por su excelente contraste, colores vibrantes y eficiencia energética. A diferencia de las pantallas LCD tradicionales, las OLED no requieren retroiluminación, ya que cada píxel emite su propia luz. Esto no solo mejora la calidad visual, sino que también reduce el consumo de energía, haciéndolas ideales para dispositivos portátiles y proyectos embebidos.

En proyectos IoT, las pantallas OLED se utilizan comúnmente para mostrar datos en tiempo real, como lecturas de sensores, notificaciones y gráficos, gracias a su capacidad de representar información de forma clara incluso en dispositivos pequeños (Cabo Tinoso, 2023).

2.5.5. YL-83

El YL-83 es un sensor diseñado para detectar la presencia de agua en forma de lluvia o humedad. Este dispositivo emite una señal digital cuando su superficie detecta agua, lo que lo hace ideal para sistemas de monitoreo ambiental, automatización del riego y estaciones meteorológicas.

El diseño del YL-83 incluye una placa sensora que cambia su resistencia al entrar en contacto con el agua, proporcionando una señal precisa y confiable. Su bajo costo y facilidad de uso lo convierten en una herramienta popular para proyectos de nivel educativo e industrial (SpainLabs, 2017).

2.6. Metodología de Desarrollo Iterativo e Incremental

La metodología de desarrollo iterativo e incremental es un enfoque ampliamente utilizado en la ingeniería de software que permite la evolución progresiva del sistema mediante ciclos repetidos de planificación, desarrollo, prueba y evaluación. Este método se basa en la idea de construir el sistema gradualmente, añadiendo nuevas funcionalidades en cada ciclo, lo que proporciona flexibilidad para realizar ajustes según las necesidades emergentes y la retroalimentación de los usuarios. Este enfoque es particularmente valioso en proyectos de telecomunicaciones, donde los requisitos pueden cambiar rápidamente debido a la naturaleza dinámica del sector y la necesidad de implementar tecnologías emergentes (Sommerville, 2011).

2.6.1. Características de la Metodología Iterativa e Incremental

- **Desarrollo en Ciclos Repetidos:** La metodología se organiza en iteraciones o ciclos, cada uno de los cuales se enfoca en desarrollar, probar y mejorar un conjunto específico de funcionalidades. Al dividir el proyecto en segmentos manejables, se facilita la identificación y corrección temprana de problemas, lo cual reduce el riesgo de retrasos o errores graves en las etapas finales del desarrollo (Pressman, 2010). Según Sommerville (2011), cada iteración puede durar entre dos y cuatro semanas, dependiendo de la complejidad de las funcionalidades a desarrollar.
- **Incremento Progresivo:** Durante cada ciclo, se añaden nuevas funcionalidades al sistema, construyendo sobre las bases establecidas en iteraciones previas. Este enfoque permite integrar de manera continua nuevas características y mejoras,

asegurando que cada versión del sistema sea completamente funcional (Pressman, 2010).

- **Retroalimentación Continua:** Al final de cada iteración, se recopila la retroalimentación de los usuarios finales y de los stakeholders, lo que permite ajustar el desarrollo a las necesidades y expectativas reales del entorno de trabajo. Este flujo continuo de información es crucial para asegurar que el sistema evolucione de acuerdo con los requerimientos de la empresa y se alinee con sus objetivos estratégicos (García y Rodríguez, 2019).
- **Adaptabilidad a Cambios:** La metodología Iterativa e Incremental es flexible y permite hacer ajustes en el plan de desarrollo en función de los cambios en los requisitos o el entorno del proyecto. Esta capacidad de adaptación es especialmente útil en el contexto de telecomunicaciones, donde los cambios tecnológicos y las nuevas demandas del mercado pueden requerir una rápida modificación del sistema (Sommerville, 2011).

2.6.2. Fases del Desarrollo Iterativo e Incremental

Las fases de esta metodología son fundamentales para estructurar y organizar el proyecto de forma eficaz. Cada fase tiene un conjunto de actividades específicas que garantizan la evolución progresiva y controlada del sistema.

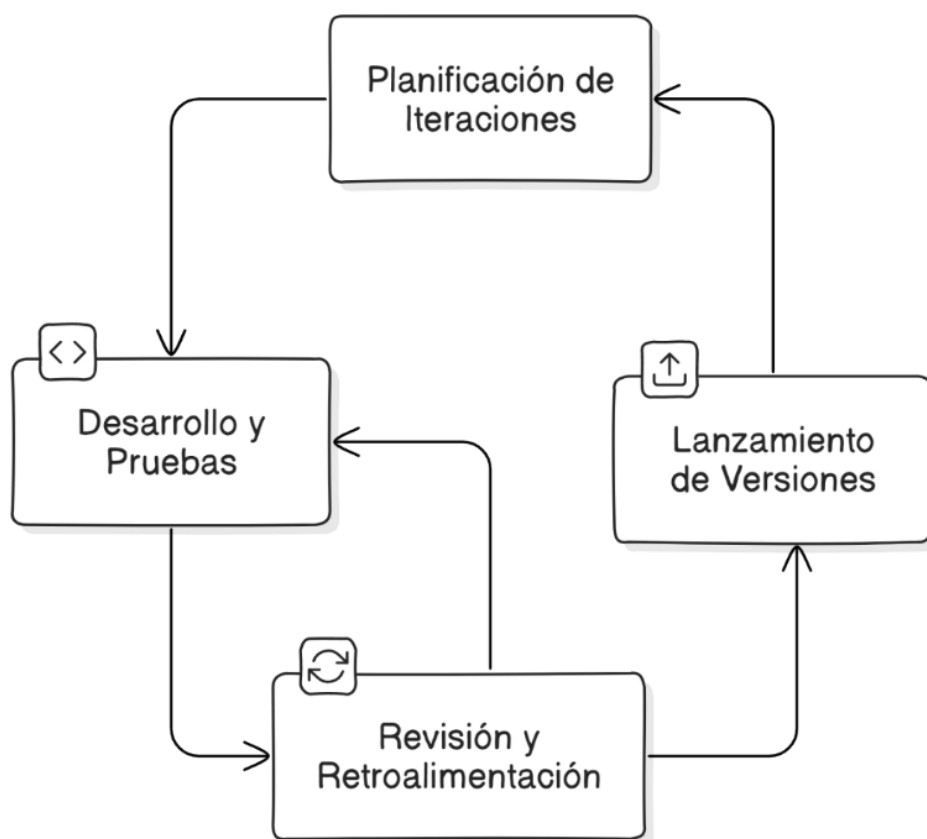
- **Planificación de Iteraciones:** Cada iteración comienza con una fase de planificación detallada. En esta fase, se definen los objetivos específicos de la iteración, los módulos o funcionalidades a desarrollar, y se asignan los recursos y responsabilidades. La planificación se basa en la evaluación de las iteraciones anteriores y en la retroalimentación obtenida, lo que permite ajustar los objetivos y las tareas para la siguiente iteración (Rodríguez y Luna, 2018). Según Sommerville (2011), una buena

planificación implica la identificación de tareas, la asignación de prioridades y la estimación de tiempos de desarrollo y pruebas.

- **Desarrollo y Pruebas:** Durante esta fase, se implementan las funcionalidades planificadas en cada iteración. Es fundamental que el desarrollo se realice de manera modular para facilitar la integración de nuevas características y asegurar la independencia de cada componente. Posteriormente, se realizan pruebas para verificar la correcta funcionalidad y detectar posibles errores. Estas pruebas pueden incluir pruebas unitarias, de integración y de regresión, dependiendo del módulo en desarrollo (Pressman, 2010).
- **Revisión y Retroalimentación:** Al final de cada iteración, se lleva a cabo una revisión del sistema en su estado actual. Esta revisión implica una fase de pruebas en un entorno de preproducción o en un entorno controlado de producción, donde se evalúa el rendimiento del sistema y se recopila la retroalimentación de los usuarios finales. La información obtenida es vital para realizar mejoras en la siguiente iteración y ajustar el plan de desarrollo según las necesidades detectadas (García y Rodríguez, 2019).
- **Lanzamiento de Versiones:** Tras cada iteración, se libera una nueva versión del sistema, que incluye todas las funcionalidades y mejoras integradas durante el ciclo. Esto permite facilitar una adaptación gradual al nuevo sistema. Además, el lanzamiento iterativo reduce el riesgo de fallos en el sistema y proporciona un entorno de desarrollo más controlado (Pressman, 2010).

Figura 1.

Ciclo de vida de la metodología iterativa e incremental



Nota: El gráfico representa el ciclo de vida de la metodología iterativa e incremental (Monreal, 2018).

2.7. Métricas de Calidad

Las métricas de calidad en ingeniería de software son herramientas fundamentales que permiten medir el desempeño y la efectividad de un sistema desarrollado, asegurando que cumpla con los objetivos y requisitos establecidos. Estas métricas proporcionan criterios objetivos para evaluar diversos aspectos del sistema, como su precisión, robustez, facilidad de uso, eficiencia en el manejo de recursos, mantenimiento y capacidad de adaptación en diferentes entornos. La norma ISO/IEC 25010 establece un modelo de calidad de software

basado en una serie de características y subcaracterísticas que cubren las dimensiones clave de la calidad del producto (ISO/IEC, 2011).

A continuación, se presentan las métricas de calidad más relevantes aplicadas al sistema desarrollado para DAIFO TELECOM S.R.L., con explicaciones y fórmulas donde sean aplicables:

2.7.1. Funcionalidad

La funcionalidad mide cómo el sistema cumple con los requisitos específicos del negocio, evaluando la precisión y completitud de las funciones proporcionadas. Es crucial que un sistema de gestión en telecomunicaciones garantice la correcta ejecución de tareas críticas, como la administración de clientes y servicios, la generación de reportes y el monitoreo de operaciones en tiempo real. La cobertura de requisitos funcionales es una métrica clave que se puede calcular mediante la fórmula:

$$\text{Cobertura de Requisitos} = \frac{\text{Número de Requisitos Implementados}}{\text{Número Total de Requisitos}} \times 100 \quad (1)$$

Esta métrica ayuda a asegurar que todas las funcionalidades planificadas han sido implementadas adecuadamente (Ochoa y Mejía, 2014).

2.7.2. Confiabilidad

La confiabilidad se refiere a la capacidad del sistema para funcionar sin fallas bajo condiciones específicas durante un período de tiempo determinado. Es una característica fundamental en sectores como las telecomunicaciones, donde la interrupción del servicio puede tener un gran impacto. La Tasa de Fallos es una métrica que se puede emplear para evaluar la confiabilidad del sistema, y se calcula de la siguiente manera:

$$\text{Confiabilidad} = 100 - (\text{Tasa de Fallos} \times 100) \quad (2)$$

Además, se pueden utilizar métricas como el Tiempo Medio entre Fallos (MTBF) y el Tiempo Medio de Reparación (MTTR) para evaluar la capacidad del sistema para operar de forma continua y recuperarse rápidamente en caso de fallo (ISO/IEC, 2011).

2.7.3. Usabilidad

La usabilidad mide la facilidad con la que los usuarios pueden aprender a utilizar el sistema y completar las tareas requeridas de manera eficiente. La experiencia de usuario es un factor clave en la usabilidad, que se puede evaluar mediante pruebas de usuarios, análisis de tareas y encuestas de satisfacción. Entre las métricas de usabilidad, destaca la Tasa de Error del Usuario, que se calcula como:

$$\text{Usabilidad} = 100 - (\text{Tasa de Error} \times 100) \quad (3)$$

Otra métrica relevante es el Tiempo Promedio para Completar Tareas, que mide cuán eficiente es el sistema para que los usuarios realicen sus actividades (Nielsen, 2012).

2.7.4. Eficiencia

La eficiencia evalúa cómo el sistema utiliza los recursos del hardware, como CPU y memoria, para realizar tareas específicas. Un software eficiente debe minimizar el consumo de recursos y evitar sobrecargas que puedan afectar el rendimiento general.

Otra métrica comúnmente utilizada es la Latencia de Respuesta, que mide el tiempo que tarda el sistema en responder a una solicitud del usuario (Sommerville, 2011).

2.7.5. Mantenibilidad

La mantenibilidad mide la facilidad con la que se pueden realizar modificaciones en el sistema para corregir defectos, mejorar el rendimiento o adaptarlo a nuevos requisitos. Esta métrica es esencial para asegurar que el sistema se pueda actualizar y mejorar a lo largo del

tiempo sin afectar su estabilidad. La métrica de Densidad de Errores se puede calcular mediante la fórmula:

$$\text{Mantenibilidad} = 100 - \left(\frac{\text{Densidad de Errores}}{\text{Densidad de Errores Permitida}} \times 100 \right) \quad (4)$$

Esta métrica permite identificar áreas del código que pueden requerir mejoras en la calidad o la estructura (Pressman, 2010).

2.7.6. Portabilidad

La portabilidad se refiere a la capacidad del sistema para ser transferido de un entorno a otro sin requerir modificaciones significativas. En un entorno multiplataforma, como el desarrollado para DAIFO TELECOM S.R.L., la portabilidad es crucial para garantizar la compatibilidad con diferentes dispositivos y sistemas operativos. La métrica de Compatibilidad de Plataforma mide la cantidad de plataformas en las que se ha verificado la operación del sistema, calculada como:

$$\text{Compatibilidad de Plataforma} = \frac{\text{Número de Plataformas Compatibles}}{\text{Número Total de Plataformas Objetivo}} \times 100 \quad (5)$$

Esto permite asegurar que el sistema se pueda ejecutar de manera eficiente en todas las plataformas designadas, como Android, Windows y Linux (ISO/IEC, 2011).

2.8. Costo

La gestión y el control de costos son aspectos fundamentales en el desarrollo de proyectos de software. La capacidad de estimar los costos con precisión permite a los gestores de proyectos asignar recursos de manera eficiente y controlar el presupuesto durante todo el ciclo de vida del desarrollo. Para este propósito, se utiliza el Modelo de Análisis de Puntos de Función (FPA), una técnica que mide el tamaño del software en términos de las funcionalidades que proporciona al usuario final (Albrecht y Gaffney, 1983).

2.8.1. Identificación de Funcionalidades

En el modelo FPA, la identificación de funcionalidades implica un análisis detallado de los requisitos del sistema para determinar las funciones clave, tales como entradas, salidas, consultas, interfaces externas y archivos internos. Cada funcionalidad se cuantifica como un punto de función, el cual contribuye al tamaño total del software y permite estimar de manera objetiva la cantidad de trabajo requerido para su desarrollo. La fórmula básica para calcular los puntos de función es:

$$\begin{aligned} \text{Puntos de Función} & \qquad \qquad \qquad (6) \\ & = \sum(\text{Entradas} + \text{Salidas} + \text{Consultas} + \text{Archivos Internos} \\ & \quad + \text{Interfaces Externas}) \end{aligned}$$

Donde cada elemento se pondera según su complejidad: baja, media o alta (Albrecht, 1983).

2.8.2. Asignación de Complejidad

Cada funcionalidad se clasifica según su nivel de complejidad, lo cual permite ajustar las estimaciones de costo de acuerdo con la dificultad de su implementación. Por ejemplo, una entrada simple que recopila datos básicos tiene una ponderación menor que una entrada compleja que incluye validaciones avanzadas y procesamiento de datos. La asignación de complejidad es esencial para obtener una estimación realista del esfuerzo necesario (Villalobos y López, 2014).

2.8.3. Cálculo de Esfuerzo

Para calcular el esfuerzo necesario, se utilizan fórmulas que relacionan los puntos de función con las horas-persona y los costos monetarios. Una fórmula comúnmente empleada para estimar el esfuerzo en términos de horas-persona es:

$$\text{Esfuerzo (HH)} = \text{Puntos de Función} \times \text{Productividad (HH/FP)} \qquad (7)$$

Donde HH/FP representa la productividad del equipo de desarrollo en horas-hombre por punto de función. Esta métrica varía según la experiencia del equipo y la complejidad del proyecto. Una vez estimado el esfuerzo, se puede calcular el costo total multiplicando las horas-persona por el costo por hora (Villalobos y López, 2014). Este enfoque permite una evaluación precisa y detallada de los costos del proyecto, facilitando su planificación y control financiero.

2.9. Seguridad

La seguridad es un elemento crítico en el desarrollo de sistemas de información, especialmente en sectores donde se manejan datos críticos y confidenciales, como en telecomunicaciones. Los estándares internacionales como ISO/IEC 27001 proporcionan un marco para gestionar y asegurar la información de manera efectiva, estableciendo políticas y controles para proteger la confidencialidad, integridad y disponibilidad de los datos (ISO/IEC, 2013).

2.9.1. Autenticación y Contraseñas

La autenticación es un mecanismo fundamental para verificar la identidad de los usuarios antes de permitirles el acceso al sistema. En este proyecto, se utilizan contraseñas robustas que deben cumplir con políticas de seguridad, como la longitud mínima, el uso de caracteres especiales y la renovación periódica. Además, las contraseñas se cifran mediante algoritmos avanzados como bcrypt, que aplica un proceso de hashing que hace extremadamente difícil descifrar las contraseñas mediante ataques de fuerza bruta (ISO/IEC, 2013).

2.9.2. Encriptación y Tokens

La encriptación de datos es esencial para proteger la información sensible tanto durante la transmisión como en el almacenamiento. En este proyecto, se emplea encriptación

simétrica y asimétrica para asegurar las comunicaciones entre el cliente y el servidor. Además, se utilizan JSON Web Tokens (JWT) para gestionar las sesiones de los usuarios. Los tokens permiten validar de manera segura la identidad del usuario y mantener la persistencia de la sesión, evitando ataques de suplantación (García y Rodríguez, 2015).

2.9.3. Captura de Firmas

La implementación de firmas electrónicas proporciona una forma segura y legalmente válida de autenticar transacciones y documentos. Las firmas se capturan mediante una interfaz que registra tanto la firma como una marca de tiempo, garantizando su autenticidad y rastreabilidad en entornos digitales. Esto es especialmente útil en sistemas de telecomunicaciones donde se requiere firmar y aprobar documentos de operaciones en campo (Gómez y Martínez, 2017).

2.9.4. Validación de Peticiones

La validación de peticiones implica la verificación del origen y la autenticidad de todas las solicitudes enviadas al servidor. En este proyecto, se implementan técnicas de verificación basadas en tokens y se utiliza Validación de Origen CORS (Cross-Origin Resource Sharing) para asegurar que solo las solicitudes provenientes de orígenes confiables puedan interactuar con el sistema. Esto previene ataques de falsificación de peticiones entre sitios (ISO/IEC, 2013).

2.9.5. Copias de Seguridad

Las copias de seguridad son fundamentales para proteger los datos contra fallos de hardware, pérdidas accidentales o ataques cibernéticos. En este proyecto, se realizan copias de seguridad de manera periódica y se almacenan en servidores seguros, utilizando técnicas de backup incremental para optimizar el espacio de almacenamiento. Además, se emplea la encriptación de los datos respaldados para evitar accesos no autorizados (Pressman, 2010).

2.9.6. Firma de Aplicaciones

La firma digital de aplicaciones, como los archivos APK en Android, es crucial para garantizar la autenticidad e integridad del software que se distribuye. El proceso de firma utiliza una clave privada que verifica la identidad del desarrollador y asegura que los archivos no han sido alterados desde su creación. Esto protege a los usuarios de la instalación de aplicaciones maliciosas y asegura la confianza en el software distribuido (ISO/IEC, 2013).

CAPÍTULO III

MARCO APLICATIVO

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



3. MARCO APLICATIVO

3.1. Introducción

El presente capítulo aborda la aplicación práctica de la metodología de desarrollo Iterativa e Incremental, implementada para la construcción del sistema multiplataforma de gestión de operaciones en DAIFO TELECOM S.R.L. En este contexto, se detallará el uso de las herramientas de desarrollo frontend y backend, la integración de componentes de hardware, y la implementación de módulos específicos orientados a optimizar la gestión de las tareas operativas y administrativas de la empresa.

El objetivo principal de este capítulo es describir el proceso de diseño, desarrollo e implementación del sistema, utilizando los modelos y tecnologías descritos previamente. La metodología Iterativa e Incremental se ha empleado para dividir el desarrollo del sistema en ciclos cortos, cada uno de los cuales incorpora nuevas funcionalidades que responden a las necesidades identificadas. Además, se expondrán los esquemas de diseño de software y las interfaces de usuario, fundamentadas en un enfoque que prioriza la experiencia del usuario y la accesibilidad en diferentes plataformas y dispositivos.

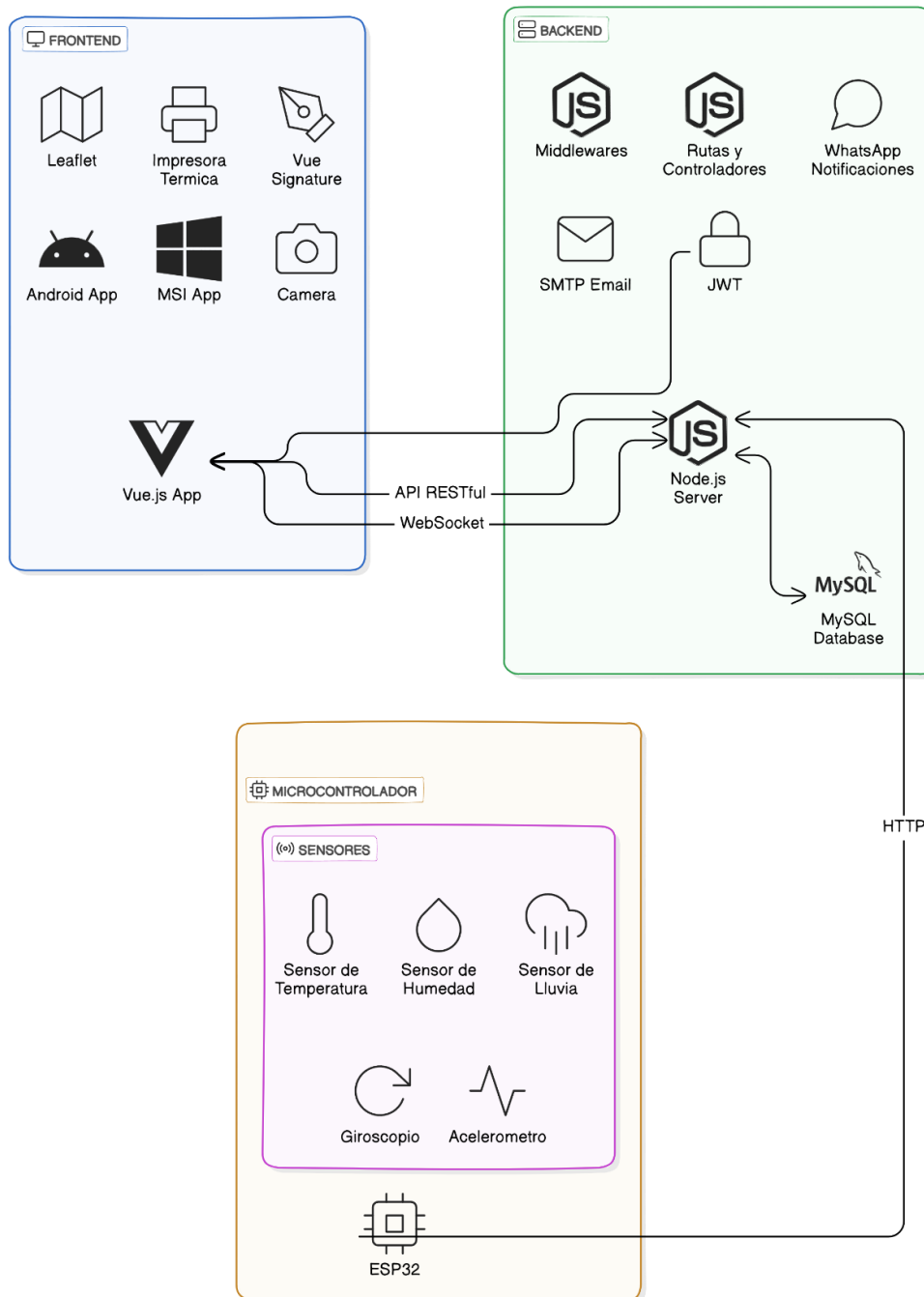
Asimismo, se presentará la estructura del sistema, los requisitos funcionales y no funcionales obtenidos mediante entrevistas y análisis del entorno operativo de DAIFO TELECOM S.R.L. Estos requerimientos se traducen en diagramas de casos de uso y modelos conceptuales que permiten comprender las interacciones entre los distintos actores del sistema y las funcionalidades específicas que cada uno de ellos requiere.

El capítulo concluirá con la descripción del despliegue del sistema en un entorno de producción, explicando las configuraciones y ajustes necesarios para su funcionamiento adecuado. Este enfoque práctico asegura que el sistema propuesto cumpla con los objetivos de optimización de operaciones, mejora de la eficiencia y reducción de tareas repetitivas en la empresa, proporcionando una solución integral y escalable para DAIFO TELECOM S.R.L.

3.2. Esquema del sistema

Figura 2.

Esquema general del sistema



3.3. Aplicación de la Metodología Iterativa e Incremental

A continuación, se desarrollan las iteraciones realizadas en el presente proyecto siguiendo la metodología iterativa e incremental.

3.3.1. Primera Iteración

En esta primera iteración se desarrollaron las fases de la metodología iterativa e incremental, fundamentales para establecer una base sólida y escalable en esta etapa del proyecto.

3.3.1.1. Planificación de Iteración

3.3.1.1.1. Definición de Objetivos y Funcionalidades Iniciales

El objetivo primordial en esta primera iteración fue establecer una solución sólida y eficiente para el manejo de las órdenes de trabajo, que serviría como la columna vertebral del sistema. Este módulo tenía como propósito centralizar y gestionar de manera ordenada todas las operaciones relacionadas con las órdenes de trabajo, abarcando desde la creación, asignación y seguimiento, hasta la revisión y cierre de cada orden. La solución debía integrar funciones clave para la administración de personal técnico, optimizando la asignación de tareas y mejorando la comunicación con los clientes de DAIFO TELECOM S.R.L. Con la implementación de este módulo, se buscó reducir los tiempos de gestión, minimizar los errores en la asignación de tareas y proporcionar una visibilidad completa del estado de cada orden en todo momento.

3.3.1.1.2. Análisis de Requerimientos del Usuario

Durante esta fase, se llevaron a cabo sesiones de recolección de requisitos con los futuros usuarios del sistema, quienes proporcionaron una visión detallada de sus necesidades específicas para el registro, la supervisión y la actualización de las órdenes de trabajo. Estas reuniones permitieron captar expectativas sobre la funcionalidad del sistema, incluyendo

aspectos esenciales como la asignación de personal técnico a cada orden, la posibilidad de adjuntar imágenes y firmas digitales para validar los trabajos realizados, y la capacidad de enviar notificaciones automáticas a los clientes para informarles sobre el estado de sus órdenes. Además, se discutieron y definieron las prioridades de implementación para asegurar que las funcionalidades más críticas estuvieran disponibles desde el principio.

Tabla 1.

Requerimiento Funcional Usuarios

ID	RFU-1: Requerimiento Funcional Usuarios
Elaborado por	Abdon Cristhian
Actores	Usuario
Descripción	Permitir la creación, lectura, actualización y eliminación de usuarios, con posibilidad de asignación de roles en etapas posteriores.
Restricción	No hay restricciones de rol en esta etapa.
Precondiciones	El usuario debe estar autenticado en el sistema.
Incluye	Buscador y filtros por nombre de usuario.
Frecuencia de uso	Alta
Prioridad	Alta
Nota	Implementar medidas de seguridad en las contraseñas mediante encriptación bcrypt.

Tabla 2.

Requerimiento Funcional Empresas

ID	RFE-1: Requerimiento Funcional Empresas
Elaborado por	Abdon Cristhian
Actores	Usuario
Descripción	Permitir la edición de empresas en el sistema, asociándolas a clientes y servicios.
Restricción	No hay restricciones de rol en esta etapa.
Precondiciones	El usuario debe estar autenticado en el sistema.
Incluye	
Frecuencia de uso	Moderada
Prioridad	Media
Nota	Los valores de la empresa serán utilizados para los reportes y comprobantes

Tabla 3.

Requerimiento Funcional Planes

ID	RFP-1: Requerimiento Funcional Planes
Elaborado por	Abdon Cristhian
Actores	Usuario
Descripción	Permitir la creación, modificación y eliminación de planes, con posibilidad de asociarlos a clientes.
Restricción	No hay restricciones de rol en esta etapa.

Precondiciones	El usuario debe estar autenticado en el sistema.
Incluye	Buscador y filtros por nombre y características del plan.
Frecuencia de uso	Alta
Prioridad	Alta
Nota	Asegurar que los cambios en los planes no afectan los datos asociados a clientes y empresas.

Tabla 4.

Requerimiento Funcional Nodos

ID	RFN-1: Requerimiento Funcional Nodos
Elaborado por	Abdon Cristhian
Actores	Usuario
Descripción	Permitir la creación, edición y eliminación de nodos de red, vinculándolos con empresas y clientes.
Restricción	No hay restricciones de rol en esta etapa.
Precondiciones	El usuario debe estar autenticado en el sistema.
Incluye	Buscador con filtro por ubicación o nombre de nodo.
Frecuencia de uso	Moderada
Prioridad	Media
Nota	La modificación de un nodo debe garantizar la actualización de clientes vinculados sin perder la integridad de los datos.

Tabla 5.*Requerimiento Funcional Servicios*

ID	RFS-1: Requerimiento Funcional Servicios
Elaborado por	Abdon Cristhian
Actores	Usuario
Descripción	Permitir la creación, modificación y eliminación de servicios que se ofrecen a los clientes, vinculándolos a empresas.
Restricción	No hay restricciones de rol en esta etapa.
Precondiciones	El usuario debe estar autenticado en el sistema.
Incluye	Buscador y filtros por tipo de servicio o empresa asociada.
Frecuencia de uso	Alta
Prioridad	Alta
Nota	Asegurar que los servicios editados o eliminados mantengan la coherencia en registros históricos de clientes.

Tabla 6.*Requerimiento Funcional Tipos de Clientes*

ID	RFTC-1: Requerimiento Funcional Tipos de Clientes
Elaborado por	Abdon Cristhian
Actores	Usuario
Descripción	Permitir la gestión de tipos de clientes
Restricción	No hay restricciones de rol en esta etapa.

Precondiciones	El usuario debe estar autenticado en el sistema.
Incluye	Buscador y filtros por tipo de cliente.
Frecuencia de uso	Media
Prioridad	Media
Nota	Garantizar la adaptación de tipos de clientes a diferentes perfiles de servicios ofrecidos.

Tabla 7.

Requerimiento Funcional Clientes

ID	RFC-1: Requerimiento Funcional Clientes
Elaborado por	Abdon Cristhian
Actores	Usuario
Descripción	Permitir la creación, modificación y eliminación de clientes, asignándoles nodos, planes y tipos de cliente.
Restricción	No hay restricciones de rol en esta etapa.
Precondiciones	El usuario debe estar autenticado en el sistema.
Incluye	Buscador con filtros por nombre, tipo de cliente, y nodo.
Frecuencia de uso	Alta
Prioridad	Alta
Nota	

Tabla 8.*Requerimiento Funcional Órdenes de Trabajo*

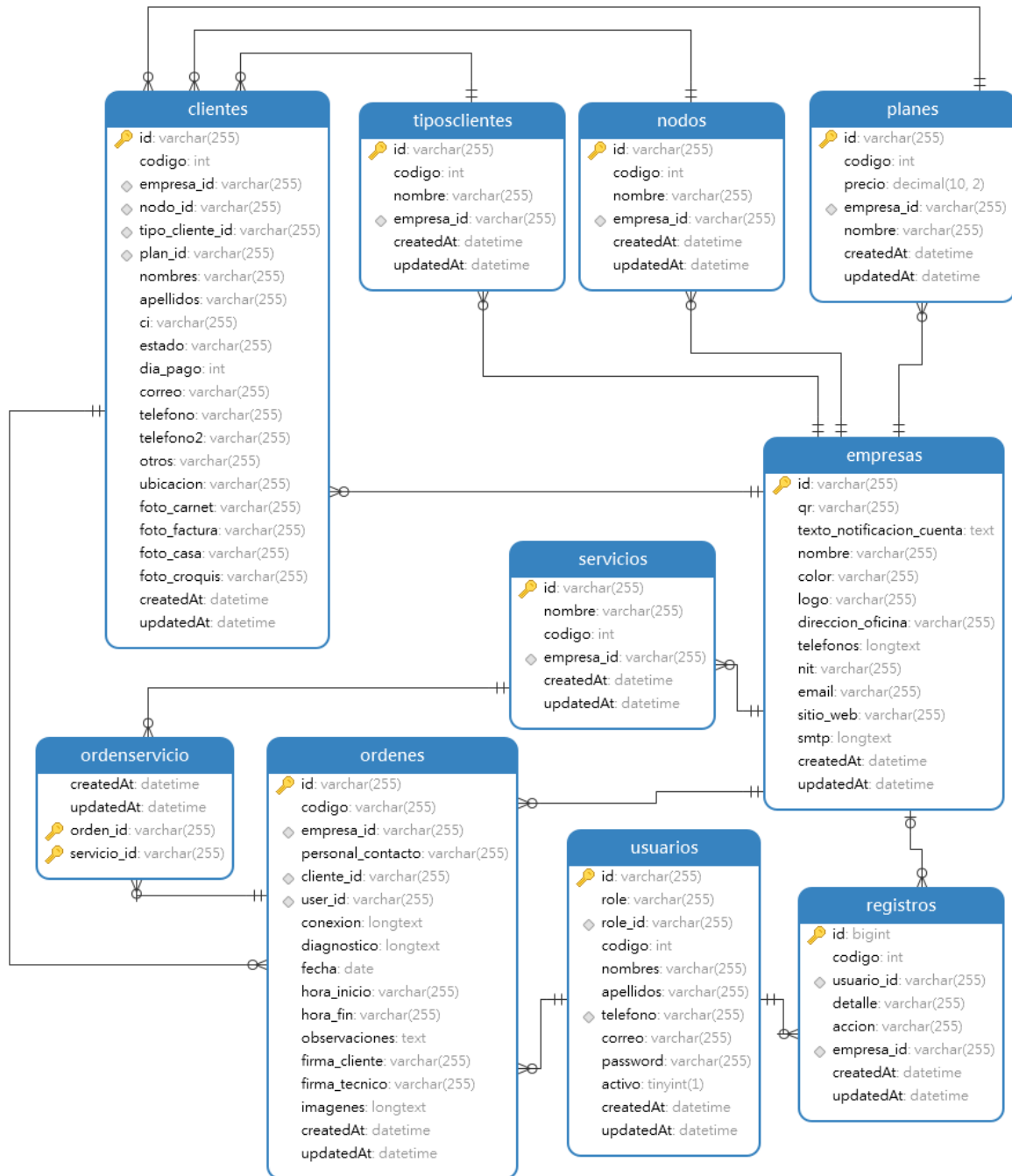
ID	RFO-1: Requerimiento Funcional Órdenes de Trabajo
Elaborado por	Abdon Cristhian
Actores	Usuario
Descripción	Permitir la creación, seguimiento y finalización de órdenes de trabajo, con posibilidad de asignar servicios y clientes.
Restricción	No hay restricciones de rol en esta etapa.
Precondiciones	El usuario debe estar autenticado en el sistema.
Incluye	Buscador con filtros por estado, cliente y fecha de creación.
Frecuencia de uso	Alta
Prioridad	Alta
Nota	Asegurar que los datos de la orden de trabajo se mantengan seguros y accesibles.

3.3.1.1.3. Diseño de la Base de Datos.

Basado en los requisitos recolectados, se diseñaron las siguientes estructuras de base de datos fundamentales para soportar el módulo de órdenes de trabajo:

Figura 3.

Diseño de la Base de datos (1ª iteración)



3.3.1.1.4. Arquitectura del sistema

- **Modelo Ordenes:** Diseñado para registrar cada detalle de las órdenes de trabajo, incluyendo información del cliente, diagnósticos y observaciones. Se incluyeron campos como conexión y diagnostico para almacenar información estructurada en formato JSON, permitiendo flexibilidad en los datos registrados.
- **Hooks de Sequelize:** Se implementaron hooks dentro del modelo para automatizar la creación de códigos únicos por año fiscal y garantizar la unicidad y trazabilidad de cada orden.
- **CRUD de Órdenes de Trabajo:** Se planificó el desarrollo de interfaces para la creación, lectura, actualización y eliminación de órdenes de trabajo, asegurando que el sistema pueda manejar operaciones completas de gestión de datos.
- **Autenticación de Usuarios:** Se estableció la necesidad de desarrollar un sistema robusto para el inicio de sesión, garantizando que solo usuarios autorizados puedan acceder al sistema y realizar operaciones.
- **Selección de Tecnologías:** Se decidió utilizar Node.js para el backend por su eficiencia en manejo de I/O y Sequelize ORM para interactuar con la base de datos, facilitando el manejo de relaciones y migraciones.
- **Entorno de Desarrollo:** Se configuraron entornos de desarrollo, prueba y producción, utilizando prácticas de integración continua para asegurar la calidad y la estabilidad del código.

Se definieron claramente los entregables para la primera iteración, incluyendo la implementación inicial del módulo de órdenes de trabajo y su funcionalidad básica de CRUD, junto con la infraestructura de autenticación básica.

3.3.1.2. Desarrollo y Pruebas

3.3.1.2.1. Integración de la Base de Datos

La integración se realizó utilizando Sequelize ORM, que facilitó la gestión de relaciones entre las tablas.

Figura 4.

Conexión a la base de datos Sequelize

```
const { Sequelize } = require('sequelize');
require('dotenv').config();
const sequelize = new Sequelize(
  process.env.DB_NAME,
  process.env.DB_USER,
  process.env.DB_PASSWORD, {
    host: 'localhost',
    dialect: 'mysql',
    logging: false,
    timezone: '-04:00',
  });
module.exports = sequelize;
```

3.3.1.2.2. Desarrollo de Interfaces de Usuario

Se emplearon prácticas de diseño responsivo para asegurar que las interfaces fueran accesibles y eficientes en diferentes dispositivos, mejorando la accesibilidad y la experiencia del usuario.

Figura 5.

Vista responsiva del sistema (Móvil)



3.3.1.2.3. Implementación de Funcionalidades

Focalizado en crear una experiencia de usuario fluida y funcional, se desarrollaron operaciones CRUD completas para las órdenes de trabajo. Esto incluyó funcionalidades avanzadas como filtros de búsqueda, ordenación y paginación, diseñadas para manejar eficientemente grandes volúmenes de datos.

Figura 6.

Métodos asíncronos del controlador Orden

```
const OrdenController = {  
  getOrdenes: async (req, res) => { ...  
  },  
  getOrden: async (req, res) => { ...  
  },  
  editOrden: async (req, res) => { ...  
  },  
  addOrden: async (req, res) => { ...  
  },  
  deleteot: async (req, res) => { ...  
  },  
};
```

Figura 7.

Diseño de tabla con filtros y ordenación

Ordenes de Trabajo AÑADIR

Noviembre 2025

🔍 Buscar

Sin registros

Registros por página 20

0-0 de 0 |< < > >|

3.3.1.2.4. Pruebas Unitarias

Se implementaron extensas pruebas para cada funcionalidad del módulo, asegurando que cada componente funcionara correctamente bajo diferentes escenarios.

Tabla 9.

Pruebas unitarias Ordenes de Trabajo

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT1-001	getOrdenes	Retorna órdenes de trabajo de una empresa en un rango de fechas específico	empresa_id, mes, gestion	Lista de órdenes filtradas
UT1-002	getOrden	Obtiene detalles de una orden de trabajo a partir de su ID	id	Detalles de la orden en formato JSON
UT1-003	addOrden	Crea una nueva orden de trabajo, guarda imágenes y firmas	Varias	Mensaje: "Orden de Trabajo registrada"
UT1-004	deleteot	Elimina una orden y sus archivos asociados	id	Mensaje: "OT eliminado correctamente"
UT1-005	editOrden	Actualiza detalles de una orden y sus imágenes	Varias	Mensaje: "Orden de Trabajo actualizada"

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT1-006	getpdfot	Genera un PDF de la orden de trabajo	codigo	PDF de la orden
UT1-007	sendmailot	Envía un correo con el PDF de la orden adjunto	codigo, asunto, correo	Mensaje: "Correo enviado"

3.3.1.2.5. Pruebas de Integración

Se validó la interoperabilidad entre el módulo de órdenes de trabajo y otros módulos, como el módulo de usuarios y clientes, para garantizar una integración fluida y sin errores.

Figura 8.

Interfaz de Integración Orden – Cliente – Plan – Nodo

Datos Generales

Personal de contacto* **Tipo de Cliente***

Juan Perez Clientes ISP La Paz ▼

Cliente* **Técnico Asignado***

1035 - DIEGO GAVINCHA ▼ Abdon Cristhian Perez Garnica

Datos del cliente

Código: 1035
Nombres y apellidos: DIEGO GAVINCHA
Nodo: PERIFERICA
Tipo de cliente: Clientes ISP La Paz
Plan: PLAN CONVENIO
Correo:
Teléfono: 70152710

Servicios*

RETIRO DE EQUIPOS POR BAJA DE SERVICIO ✕ ▼

Tabla 10.*Pruebas de Integración Ordenes de Trabajo*

ID	Función	Descripción	Entrada	Resultado Esperado
Prueba				
IT1-001	addOrden + Cliente	Verifica la creación de una orden con asignación correcta a un cliente	Data, cliente_id	Orden creada y vinculada correctamente
IT1-002	addOrden + Servicio	Comprueba la asociación de múltiples servicios seleccionados al registrar una nueva orden	Data, selectedS ervicios	Orden creada con todos los servicios asociados
IT1-003	addOrden + Usuario	Valida la asignación de la orden al usuario técnico que la crea	Data, user_id	Orden creada y asignada al usuario
IT1-004	editOrden + Cliente	Verifica la capacidad de edición de una orden y la actualización del cliente asociado	Data, cliente_id	Cliente de la orden actualizado correctamente
IT1-005	deleteot + Cliente, Servicio, Usuario	Comprueba la eliminación de una orden de trabajo	id	Orden de trabajo eliminada correctamente, sin errores de dependencias

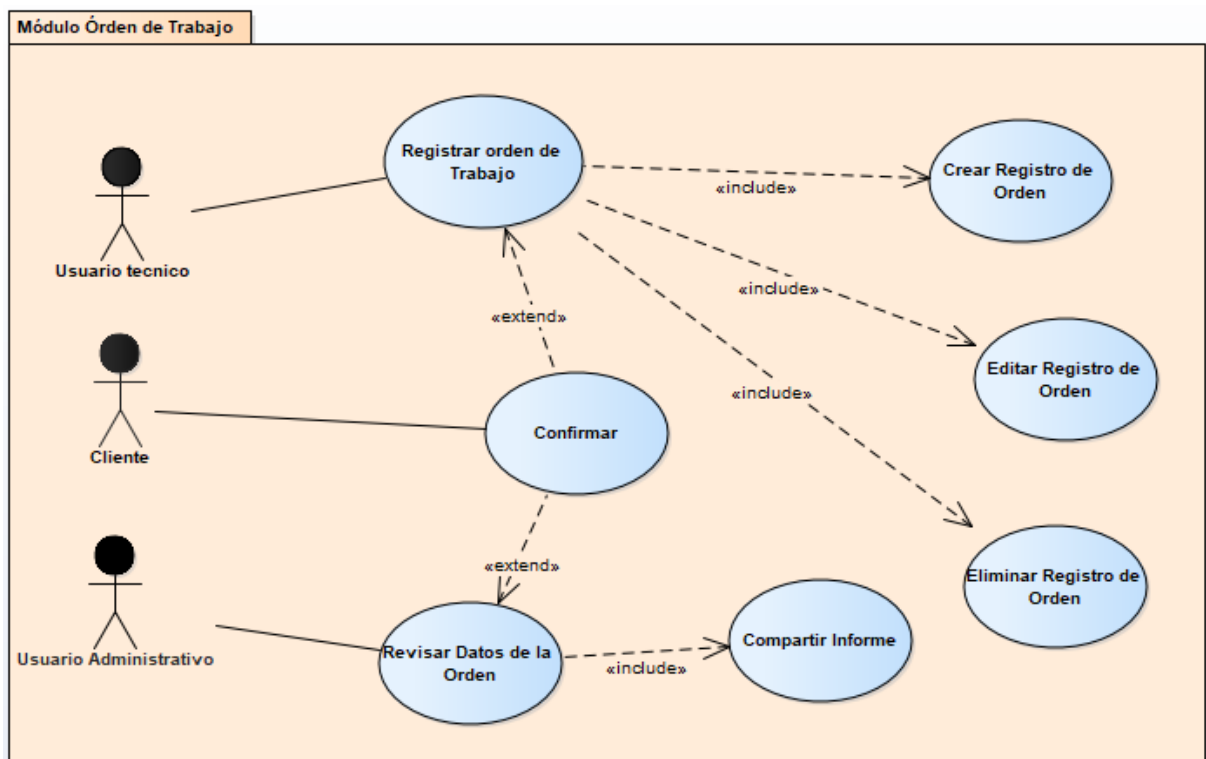
ID	Función	Descripción	Entrada	Resultado Esperado
Prueba				
IT1-006	getOrden + Cliente, Servicio	Valida que getOrden devuelva todos los datos vinculados	id	Orden recuperada con cliente, servicios y usuario asociados
IT1-007	sendmailot	Verifica el envío de correo	destinatari o	Correo enviado con datos precisos

3.3.1.2.6. Pruebas de Funcionalidad

Se llevaron a cabo pruebas exhaustivas de los flujos de trabajo completos en el módulo de órdenes de trabajo para verificar que todos los procesos se ejecutaran sin interrupciones.

Figura 9.

Diagrama de casos de uso Gestión de Ordenes de Trabajo



3.3.1.2.7. Seguridad de Datos

Se establecieron protocolos de seguridad rigurosos, incluyendo encriptación de datos sensibles y la implementación de medidas de seguridad en el acceso a la base de datos.

Tabla 11.

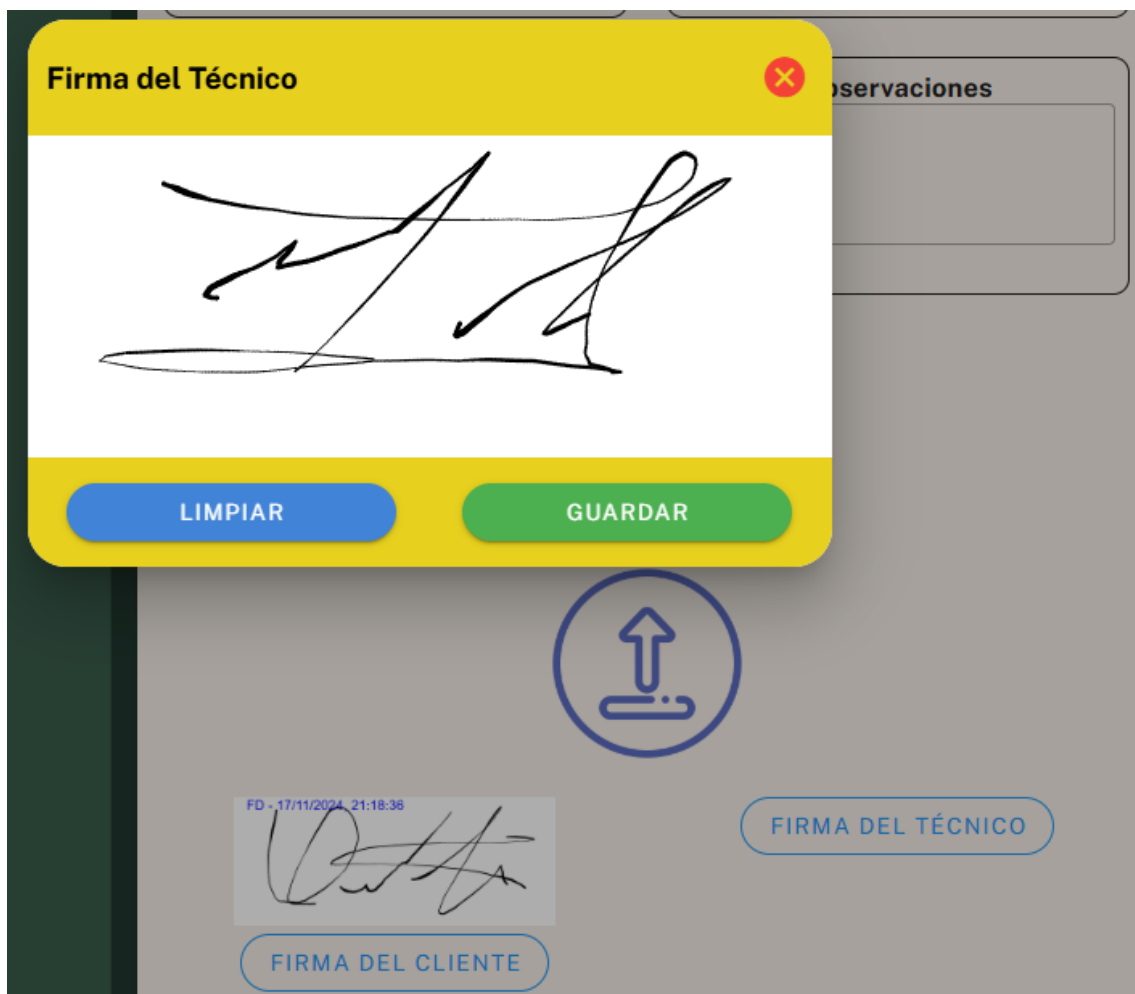
Librerías de seguridad

Librería	Versión
shortid	2.2
uuid	10.0
bcrypt	5.1
cors	2.8
casl/mongoose	8.0
jsonwebtoken	9.0

3.3.1.2.8. Firma Digital Manual

Se implementó un módulo de firma digital manual utilizando la herramienta Vue Signature, la cual proporcionaba un espacio en pantalla donde los usuarios podían firmar utilizando su dedo o un lápiz táctil (stylus). Esta funcionalidad fue diseñada para facilitar el registro de conformidad en distintos procesos operativos, como la verificación de órdenes de trabajo o la recepción de servicios.

Una vez que el usuario completaba la firma, el sistema la convertía en una imagen digital. Además, se añadía automáticamente una marca de tiempo para autenticar el momento en que fue realizada y prevenir cualquier mal uso o manipulación posterior.

Figura 10.*Captura Digital de Firma Manual*

Es importante destacar que el propósito de esta firma digital manual fue simplemente documentar la conformidad de los usuarios o clientes en situaciones específicas, como la aceptación de trabajos o la recepción de asistencias. No estaba destinada a reemplazar las firmas formales para documentos legales, contratos u otros procesos de mayor complejidad jurídica. La inclusión de esta funcionalidad buscó optimizar los flujos de trabajo y mejorar la fiabilidad del registro de conformidad en la gestión de operaciones.

Figura 11.

Utilización de Firma de Conformidad

 DaifoTelecom	ORDEN DE TRABAJO Oficina Central: Av. Gral. Juan José Torrez N° 444, La Paz Teléfonos: 2215505 / 76772794 Email: info@daifotelecom.net - www.daifotelecom.net	N° 417/2024
	DATOS DE REFERENCIA DEL CLIENTE	
Cliente: JOSE LUIS CALLE ESPINAL Correo: Teléfono: 63039670 Ubicación: -16.4608836,-68.2397958	Personal de contacto: Ing Lizeth Nodo: VALLE HERMOSO Tipo de cliente: Clientes ISP El Alto Fecha de trabajo: 2024-10-28	
SERVICIOS CAMBIO DE FUENTE DE MEDIA CONVERTER		
CONEXIÓN	DIAGNÓSTICO	
Dirección Origen: Nodo San Felipe Dirección Destino: Sector Valle Hermoso Observaciones: Participaron Ing Rolando Ing Alvaro	Descripción del trabajo realizado: Análisis de equipos y conexiones Sugerencias: USO DE CORTA PICOS Equipos y accesorios instalados: Fuente de media converter 5v	
TÉCNICO ASIGNADO	OBSERVACIONES	
Ing. Alvaro Colque Fecha: 2024-10-28 Hora de inicio: 09:00 Hora de fin: 09:10	Se detectó corte en fuente de media converter Se realizó el cambio correspondiente Realize las pruebas de tráfico de Datos Se dejó todos los equipos operativos y en buen estado	
CONFORMIDAD		
FD - 28/10/2024, 9:11:10 	FD - 28/10/2024, 9:11:19 	

3.3.1.2.9. Optimización del Rendimiento

Se realizaron pruebas de carga para evaluar y mejorar la capacidad del sistema de manejar un número significativo de transacciones simultáneas, asegurando que el sistema fuera robusto y escalable.

Figura 12.

Consultas Filtradas con Sequelize

```
const ordenes = await Orden.findAll({
  where: {
    empresa_id,
    fecha: {
      [Op.and]: [
        { [Op.gte]: new Date(gestion, mes - 1, 1) },
        { [Op.lte]: new Date(gestion, mes, 0) }
      ]
    }
  },
  include: [
    {
      model: Cliente, as: 'cliente', include: [
        { model: Nodo, as: 'nodo' },
        { model: TiposClientes, as: 'tipoCliente' }
      ]
    },
    { model: Usuario, as: 'usuario' },
    { model: Servicio, as: 'servicios' }
  ],
  order: [['codigo', 'DESC']],
});
```

3.3.1.2.10. Preparación del Entorno de Producción

El sistema fue desplegado en un servidor Ubuntu con Plesk, configurado para maximizar la seguridad y el rendimiento. Se implementaron medidas de seguridad adicionales en el servidor, incluyendo firewalls y políticas de acceso restringido.

3.3.1.2.11. Configuración de Dominios

Se establecieron dos dominios distintos para separar claramente el backend del frontend. Esto no solo mejora la seguridad al limitar el acceso directo al backend, sino que también optimiza el rendimiento del frontend al servir contenido estático de manera eficiente.

3.3.1.2.12. Monitoreo y Mantenimiento Continuo

Tras el despliegue, se inició un proceso de monitoreo continuo para detectar y resolver cualquier problema operativo rápidamente. Esto incluyó la implementación de herramientas de monitoreo de rendimiento y logs de errores para asegurar una respuesta rápida a cualquier incidente.

Esta fase de desarrollo y pruebas culmina con un despliegue estratégico que no solo pone el sistema en producción, sino que también establece las bases para un mantenimiento y una escalabilidad eficaces. Esto asegura que el sistema no solo funcione como se espera inicialmente, sino que también pueda adaptarse y expandirse en respuesta a las necesidades futuras de DAIFO TELECOM S.R.L.

3.3.1.3. Revisión y Retroalimentación

Sesiones de Evaluación con Usuarios: Se organizaron sesiones en las que los usuarios finales interactuaron con el sistema. Estas sesiones permitieron observar directamente cómo los usuarios gestionaban sus tareas cotidianas usando el nuevo sistema y recoger comentarios esenciales sobre su experiencia.

3.3.1.3.1. Análisis de los Comentarios

Los comentarios y observaciones recogidos se evaluaron para identificar problemas críticos y comenzar a considerar soluciones apropiadas.

3.3.1.3.2. Mejoras en la Tabla de Órdenes

Se añadieron campos para la 'fecha de registro', la 'firma del cliente' y la 'firma del técnico' a la tabla de órdenes para mejorar la trazabilidad y la autenticación formal de las transacciones.

3.3.1.3.3. Ampliación de la Tabla de Clientes

Se incorporaron campos para 'otros datos' (campo libre para notas), 'teléfono secundario', y 'fotografía de documentos y ubicación' para proporcionar una gestión más detallada y eficaz de la información del cliente.

3.3.1.3.4. Agenda de Desarrollo Personal

Se planificaron una serie de iteraciones de desarrollo para implementar las mejoras identificadas, que incluyeron ajustes en el diseño del software y actualizaciones en la base de datos.

3.3.1.3.5. Implementación y Pruebas

Se implementaron los cambios necesarios y se realizaron pruebas para asegurar que las mejoras funcionaran como se esperaba, manteniendo altos estándares de calidad y funcionalidad.

Esta fase de revisión y retroalimentación fue crucial para capturar y actuar sobre la experiencia del usuario, permitiendo ajustes continuos que aseguran que el sistema se desarrolle de manera que refleje y responda efectivamente a las necesidades operativas y de gestión.

3.3.1.4. Lanzamiento de Versión (1.0.0)

3.3.1.4.1. Revisión Final

Antes del lanzamiento, se realizó una revisión final exhaustiva del sistema para asegurar que todas las funcionalidades estuvieran funcionando según lo previsto y que todos los errores identificados durante las pruebas y la revisión hubieran sido corregidos.

3.3.1.4.2. Optimización del Sistema

Se llevaron a cabo optimizaciones finales, centradas en mejorar el rendimiento y la seguridad del sistema, preparándolo para un entorno de producción.

3.3.1.4.3. Despliegue en Producción

La versión 1.0 del sistema fue desplegada en el servidor de producción de DAIFO TELECOM S.R.L. Este lanzamiento marcó el inicio oficial del uso del sistema para la gestión diaria de las operaciones de la empresa.

3.3.1.4.4. Configuración Inicial del Sistema

Los operarios de la empresa comenzaron el proceso de llenar los registros necesarios en el sistema, incluyendo la creación de perfiles de usuarios, tipos de clientes, nodos y planes. Esta información es fundamental para el registro y gestión de las órdenes de trabajo dentro del sistema.

3.3.1.4.5. Sesiones de Capacitación

Se llevaron a cabo sesiones de capacitación para todos los usuarios del sistema, asegurando que comprendieran cómo utilizar las nuevas herramientas y funcionalidades de manera efectiva. Estas sesiones fueron esenciales para facilitar la transición al nuevo sistema.

3.3.1.4.6. Soporte Continuo

Se estableció un soporte continuo para atender cualquier duda o problema que los usuarios pudieran enfrentar durante las primeras semanas de implementación. Esto incluyó la disponibilidad de recursos técnicos para responder preguntas y resolver problemas rápidamente.

3.3.1.4.7. *Evaluación del Desempeño del Sistema*

Después del lanzamiento, se monitoreó el desempeño del sistema en un entorno de producción real para evaluar su estabilidad, rendimiento y la eficacia general de las soluciones implementadas.

Este lanzamiento no solo representó un hito importante en el proyecto, sino que también estableció una base sólida para las iteraciones futuras, permitiendo una mejora continua basada en el uso real y la retroalimentación activa de los usuarios. Este proceso continuo de lanzamiento y revisión ayuda a asegurar que el sistema se mantenga relevante, útil y eficiente en el cumplimiento de los objetivos de la empresa.

3.3.2. *Segunda Iteración*

3.3.2.1. **Planificación de Iteración**

Se planeó la implementación del módulo de roles y permisos utilizando CASL en Node.js y Vue.js para manejar autorizaciones, mejorando la seguridad y flexibilidad del sistema.

Se analizó el requerimiento del módulo de Site Survey para una mayor personalización en la recopilación y manejo de información detallada sobre las operaciones en campo.

Tabla 12.

Requerimiento Funcional Site Survey

ID	RFSS-1: Requerimiento Funcional Site Surveys
Elaborado por	Abdon Cristhian
Actores	Técnico

Descripción	Permitir a los técnicos registrar, editar y gestionar los <i>site surveys</i> realizados en el campo, documentando condiciones del sitio y detalles técnicos.
Restricción	Solo los usuarios con permisos asignados pueden realizar acciones de creación, edición y gestión de <i>site surveys</i> .
Precondiciones	El usuario debe estar autenticado en el sistema y contar con los permisos necesarios.
Incluye	Buscador con filtro por cliente o ubicación del <i>site survey</i> .
Frecuencia de uso	Alta
Prioridad	Alta
Nota	Asegurar la captura de información clave para facilitar el análisis y toma de decisiones en futuras instalaciones.

Tabla 13.

Requerimiento Funcional Roles

ID	RFR-1: Requerimiento Funcional Roles
Elaborado por	Abdon Cristhian
Actores	Administrador
Descripción	Permitir la creación, edición y eliminación de roles, asignando diferentes permisos de acceso y funciones a cada rol en el sistema.
Restricción	Solo los usuarios con permisos asignados pueden realizar acciones de creación, edición y eliminación de roles.

3.3.2.1.2. Planificación Técnica

Se optó por utilizar el tipo de datos textlong para los valores de los Site Survey, permitiendo así el uso de datos en formato JSON cuando sea necesario, con el objetivo de proporcionar una mayor flexibilidad en la gestión de la información. Además, se planificó la integración de middlewares en el sistema para gestionar de forma eficiente la autenticación, el control de acceso y el manejo de errores. Estos middlewares se encargan de interceptar y procesar las solicitudes, garantizando la seguridad y la consistencia de los datos, y permitiendo la implementación de funcionalidades adicionales sin afectar la lógica central de la aplicación.

3.3.2.2. Desarrollo y Pruebas

3.3.2.2.1. Desarrollo de Módulos

Se implementó la funcionalidad de optimización de imágenes con Sharp, integrándola en el proceso de carga y manipulación de imágenes para los surveys.

Figura 14.

Compresión de imágenes con Sharp

```
const uploadImages = async (files, prefix) => {
  let arrayImagenes = [];
  for (let i = 0; i < 10; i++) {
    const fileKey = `${prefix}_${i}`; const imageData = files[fileKey];
    if (imageData) {
      const imageName = `${shortid.generate()}.jpg`;
      const imagePath = path.join(uploadPath, imageName);
      const imageBuffer = imageData.data;
      await sharp(imageBuffer, { failOnError: false })
        .resize({ width: 1500, height: 1500, fit: 'inside' })
        .jpeg({ quality: 100 })
        .toFile(imagePath);
      arrayImagenes.push(`/Survey/${imageName}`);
    }
  }
  return arrayImagenes;
};
```

3.3.2.2.2. Pruebas Unitarias

Se implementaron extensas pruebas para cada funcionalidad del módulo, asegurando que cada componente funcionara correctamente bajo diferentes escenarios.

Tabla 14.

Pruebas Unitarias Site Survey

ID	Función	Descripción	Entrada	Resultado
Prueba				Esperado
UT2-001	addSurvey	Registra un nuevo <i>Site Survey</i> con toda la información necesaria.	empresa_id, informacion_general,i nformacion_empresa, informacion_torre, tecnicos, data, archivos[]	Mensaje de confirmación: "Survey registrado" y datos guardados correctamente.
UT2-002	editSurvey	Modifica un <i>Site Survey</i> existente con nuevos datos y actualiza archivos asociados.	id, informacion_general, informacion_empresa, informacion_torre, tecnicos, data, archivos[]	Mensaje de confirmación: "Survey Actualizado" y datos actualizados correctamente.
UT2-003	deletesurve y	Elimina un <i>Site Survey</i> existente y sus archivos.	id	Mensaje de confirmación: "Survey eliminado correctamente".

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT2-004	getSurveyPDF	Genera un PDF del <i>Site Survey</i> especificado.	codigo	PDF generado exitosamente del <i>Site Survey</i> .

Tabla 15.

Pruebas Unitarias Roles

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT2-005	getRoles	Obtiene una lista de roles con sus permisos asociados.	Ninguna	Lista de roles con permisos agrupados.
UT2-006	addRol	Crea un nuevo rol con el nombre especificado.	nombre	Mensaje de confirmación: "Role creado correctamente".
UT2-007	editRol	Actualiza un rol existente, modificando su nombre y permisos.	id, nombre, permissionsSelected	Mensaje de confirmación: "Role actualizado correctamente".

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT2-008	deleteRol	Elimina un rol existente si no tiene dependencias que lo restrinjan.	id	Mensaje de confirmación: "Role eliminado correctamente".

3.3.2.2.3. Pruebas de Integración

Se realizaron pruebas de integración para asegurar la funcionalidad y la interacción correcta entre los nuevos módulos y el sistema existente.

Tabla 16.

Pruebas de Integración Site Survey

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
IT2-001	addSurvey + Empresa	Verifica la creación de un survey y su vinculación correcta con una empresa existente en la base de datos.	empresa_id, informacion_gen eral, tecnicos, informacion_torre, data, archivos	Survey creado y vinculado correctamente a la empresa indicada.
IT2-002	editSurvey + Imágenes	Verifica la actualización de un survey, incluyendo la modificación de imágenes relacionadas.	id, informacion_gen eral, informacion_torre, data, archivos	Survey actualizado correctamente con imágenes modificadas.

ID	Función	Descripción	Entrada	Resultado
Prueba				Esperado
IT2-003	deletesurve y + Check	Verifica la eliminación de un survey y sus archivos asociados, incluyendo el check vinculado.	id	Survey y su check asociado eliminados correctamente.
IT2-004	getSurveys + Check	Verifica que al obtener la lista de surveys, cada uno cargue su checklist correspondiente si existe.	empresa_id, mes, gestion	Lista de surveys con checklist asociado cargado.
IT2-005	getSurveyP DF + Empresa	Verifica la generación del PDF de un survey, asegurando la inclusión de la información de la empresa.	codigo	PDF generado con información del survey y empresa.

Tabla 17.

Pruebas de Integración Roles

ID	Función	Descripción	Entrada	Resultado Esperado
Prueba				
IT2-006	addRol + Permissions	Verifica la creación de un rol y la asignación de permisos seleccionados.	nombre, permission sSelected	Rol creado con permisos vinculados correctamente.

ID	Función	Descripción	Entrada	Resultado Esperado
Prueba				
IT2-007	editRol + Permissions	Verifica la actualización de un rol, incluyendo la modificación de sus permisos.	id, nombre, permission sSelected	Rol actualizado con permisos modificados correctamente.
IT2-008	deleteRol + Usuario	Verifica la eliminación de un rol asignado a usuarios, asegurando que se gestione correctamente.	id	Rol eliminado y los usuarios afectados sin errores.
IT2-009	getRoles + Permissions	Verifica que, al obtener los roles, estos se carguen con sus permisos asociados.	Ninguna	Lista de roles con permisos asociados correctamente.

3.3.2.2.4. Pruebas de Funcionalidad

Se llevaron a cabo pruebas exhaustivas de los flujos de trabajo completos en el módulo de órdenes de trabajo para verificar que todos los procesos se ejecutaran sin interrupciones y que los datos se manejaran correctamente.

Figura 15.

Diagrama de Caso de Uso Site Survey

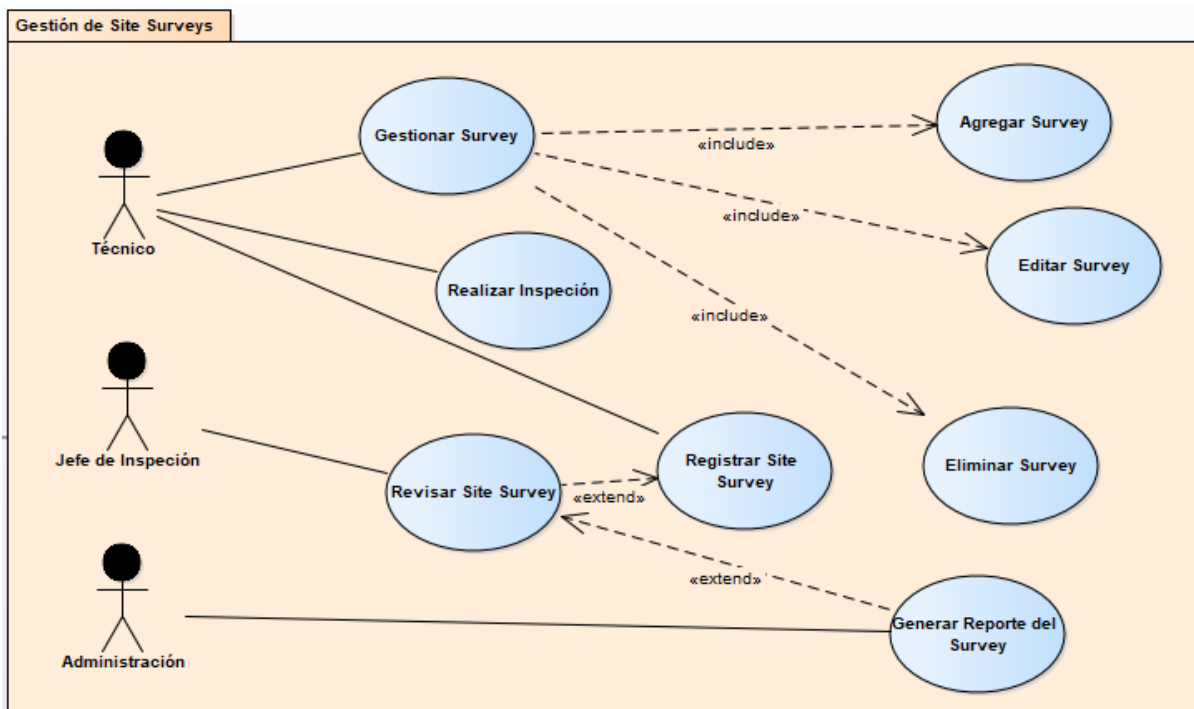


Figura 16.

Diagrama de Caso de uso Roles

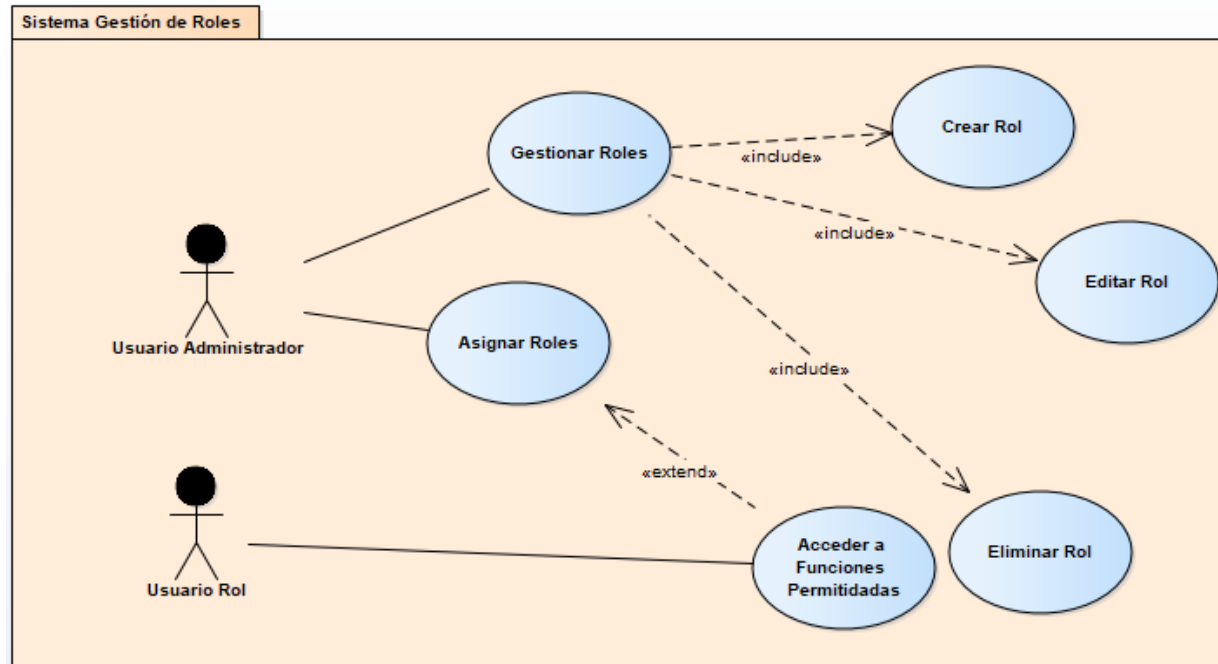


Figura 17.

Formulario dinámico Site Survey

Añadir Survey ✕


Los campos marcados con asterisco * son obligatorios.

< 1. GENERAL 2. EQUIPAMIENTO 3. ACCESO Y SEGURIDAD 4. ENTORNO CIRCUNDANTE 5. ESTADO >

Antenas: 1 −

Descripción −

Imágenes




Agregar +

Equipos de Transmisión o Recepción: 1 −

Descripción −

Imágenes



Agregar +

Cableado y Conexiones: 0 −

Agregar +

Nuevo Tipo de Registro

AGREGAR NUEVO TIPO

CANCELAR GUARDAR

Figura 18.

Formulario general Site Survey

Añadir Survey
✕

Los campos marcados con asterisco * son obligatorios.

<
1. GENERAL
2. EQUIPAMIENTO
3. ACCESO Y SEGURIDAD
4. ENTORNO CIRCUNDANTE
5. ESTADO
>

INFORMACIÓN GENERAL

<p>Tipo de registro*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text" value="SITE SURVEY"/>	<p>Tipo de trabajo*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text"/>
<p>Nivel de dificultad de acceso*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text" value="Fácil"/>	<p>Necesita permiso especial*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text" value="No"/>
<p>Detalles del permiso especial</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text"/>	<p>Fecha*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text" value="dd/mm/aaaa"/>
<p>Observaciones</p> <div style="border: 1px solid #ccc; height: 40px; width: 100%;"></div>	
<p>Latitud*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text" value="-16,47726303673197"/>	<p>Latitud*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text" value="-68,12631726264955"/>

USAR UBICACIÓN ACTUAL (GPS)

INFORMACIÓN DE LA EMPRESA

<p>Empresa*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text"/>	<p>Encargado*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text"/>
<p>Telefono*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text"/>	<p>Correo*</p> <input style="width: 95%; border: 1px solid #ccc; border-radius: 5px;" type="text"/>

Figura 19.*Asignación de permisos*

Editar Rol

Nombre*
Administrador

Usuarios
 Ver Crear Modificar Eliminar

TiposClientes
 Ver Crear Modificar Eliminar

Surveys
 Ver Crear Modificar Eliminar

Servicios
 Ver Crear Modificar Eliminar

3.3.2.3. Revisión y Retroalimentación

3.3.2.3.1. *Recolección de Retroalimentación*

Se realizaron pruebas piloto con usuarios seleccionados para evaluar la usabilidad y efectividad de los nuevos módulos, especialmente en el contexto de los roles y permisos y la recopilación de datos de Site Survey.

3.3.2.3.2. *Análisis y Ajuste*

Se ajustaron los módulos basándose en la retroalimentación recibida, refinando las funcionalidades y mejorando la integración de datos.

3.3.2.4. Lanzamiento de Versión (1.1.0)

Se lanzó oficialmente la versión 1.1.0 del sistema, que incluía todos los nuevos módulos y mejoras.

Esta segunda iteración busca no solo expandir las capacidades del sistema con nuevas funcionalidades críticas sino también optimizar el rendimiento y la usabilidad del sistema, asegurando que DAIFO TELECOM S.R.L. pueda gestionar sus operaciones de manera más eficiente y segura.

3.3.3. Tercera Iteración

3.3.3.1. Planificación de Iteración

La tercera iteración del proyecto estuvo enfocada en la planificación y desarrollo de dos módulos clave: el módulo de finanzas y el módulo de asistencias. Estos módulos representan pilares fundamentales para la gestión operativa y administrativa de la empresa, ya que abordan aspectos esenciales como el control financiero y la optimización de los servicios de asistencia técnica.

El módulo de finanzas tiene como objetivo principal centralizar y automatizar la gestión de pagos, cuentas por cobrar, y puntos de cobro, lo que reduce significativamente los errores humanos y mejora la trazabilidad de las transacciones financieras. Por otro lado, el módulo de asistencias se diseñó para optimizar el registro y seguimiento de las actividades del personal técnico en campo, asegurando una mayor eficiencia en la prestación de servicios a los clientes.

Ambos módulos fueron seleccionados para esta iteración debido a su impacto directo en la productividad y competitividad de la empresa, así como su papel crucial en garantizar un flujo de información claro y oportuno entre las áreas operativas y administrativas.

3.3.3.1.1. Definición de Objetivos y Funcionalidades

Implementación del módulo de Asistencias para gestionar visitas técnicas, incluyendo la programación, seguimiento y documentación detallada de cada asistencia.

Desarrollo del módulo de Finanzas para manejar cuentas, pagos, y generar reportes de pagos mensuales, proporcionando una visión clara y precisa de las finanzas relacionadas con los clientes y servicios.

Tabla 18.

Requerimiento Funcional Asistencias

ID	RFA-1: Requerimiento Funcional Asistencias
Elaborado por	Abdon Cristhian
Actores	Usuario
Descripción	<p>Incluye los requerimientos de gestión y asignación de asistencias.</p> <p>Permite la creación, edición, eliminación de registros de asistencias y la asignación de técnicos a cada asistencia. Cada registro puede contener detalles como cliente, tipo de asistencia, fecha de visita, y observaciones.</p>
Restricción	Solo usuarios con permisos específicos pueden gestionar y asignar asistencias.
Precondiciones	El usuario debe estar autenticado y contar con los permisos necesarios.
Incluye	Filtro para buscar asistencias por cliente, tipo o estado. Listado de técnicos para asignación.

Frecuencia de uso	Alta
--------------------------	------

Prioridad	Media
------------------	-------

Tabla 19.

Requerimiento Funcional Finanzas

ID	RFF-1: Requerimiento Funcional Finanzas
-----------	--

Elaborado por	Abdon Cristhian
----------------------	-----------------

Actores	Administrador, Usuario
----------------	------------------------

Descripción	Incluye los requerimientos de gestión de cuentas y registro de pagos. Permite la creación, edición, y eliminación de cuentas de clientes, así como el registro y modificación de pagos. Los pagos pueden incluir detalles de método de pago, monto, y descuentos.
--------------------	---

Restricción	Solo usuarios con permisos específicos pueden gestionar cuentas y registrar pagos.
--------------------	--

Precondiciones	El usuario debe estar autenticado y contar con los permisos necesarios.
-----------------------	---

Incluye	Filtro para buscar cuentas por cliente, periodo o estado de pago. Opción para cargar comprobantes de pago.
----------------	---

Frecuencia de uso	Alta
--------------------------	------

Prioridad	Alta
------------------	------

Nota	La gestión adecuada de cuentas y pagos asegura la integridad financiera del sistema y facilita el control administrativo.
-------------	---

3.3.3.1.2. Diseño de la Base de Datos

Se adaptaron los modelos de Asistencias, Cuentas, y Pagos para soportar las operaciones necesarias en cada módulo, asegurando que los datos requeridos para operaciones y reportes estén adecuadamente estructurados y sean accesibles.

Figura 20.

Diseño Parcial de la Base de datos (3° iteración)



3.3.3.1.3. Planificación Técnica

Selección de tecnologías adicionales para el procesamiento de reportes y la gestión eficiente de las imágenes y documentos asociados con las asistencias y pagos.

3.3.3.2. Desarrollo y Pruebas

3.3.3.2.1. Desarrollo de Módulos

Se implementarán completamente los módulos de Asistencias y Finanzas, asegurando que se integren sin problemas con el sistema existente.

Figura 21.

Interfaz Gestión de Pagos

The screenshot displays the 'Pagos' (Payments) management interface. At the top, there are three icons: a grid, a printer, and a document. Below these is a dropdown menu showing 'SAN FELIPE DE SEKE'. Underneath, there are two yellow boxes for selecting the month ('Octubre') and the year ('2024'). A search bar with a magnifying glass icon and the text 'Buscar' is present. The main content area shows a payment record with the following details:


- Nro:** 222
- Estado:** Valido
- Cod Cliente:** 1063
- Cliente:** CONCEPCION LUNA CALLE 77773306
- Detalle:**
 - 1. 10-2024 Bs. 175.00
- Descuento:** 0.00
- Total:** Bs. 175.00
- Metodo Pago:** Efectivo
- Referencia:** LIZETH CHIPANA
- Fecha:** 24 oct 2024
- Imagen:**

At the bottom right, there are three icons: a green eye, a blue printer, and a red document with a white 'X'.

Se incorporará la funcionalidad de generación de reportes mensuales de pagos, permitiendo un análisis financiero detallado.

Figura 22.

Comprobante de Pago Digital



DaifoTelecom
NIT: 288188020
LA PAZ - BOLIVIA
Oficina Central: Av. Gral. Juan José Torrez
 N° 444, La Paz
Teléfonos: 2215505 / 76772794
Email: info@daifotelecom.net -
 www.daifotelecom.net

COMPROBANTE DE PAGO: 222
ZONA: SAN FELIPE DE SEKE
CLIENTE: CONCEPCION LUNA CALLE
C.I.:
PLAN: PLAN ESTUDIANTE (30 MBPS)
TELÉFONO: 77773306 / 68170313

SERVICIO DE INTERNET

NRO	PERIODO	MONTO
1	10-2024	Bs. 175.00
SUBTOTAL		Bs. 175.00
DESCUENTO		Bs. 0.00
TOTAL		Bs. 175.00

SON: Ciento Setenta y Cinco Pesos
00/100 M.N.

FECHA DE PAGO: 2024-10-24
MÉTODO DE PAGO: Efectivo




Figura 23.

Reporte de Pagos por Periodo



REPORTE DE PAGOS

Oficina Central: Av. Gral. Juan José Torrez N° 444, La Paz

Teléfonos: 2215505 / 76772794

Email: info@daifotelecom.net - www.daifotelecom.net

PERIODO 10-2024

Código	Cliente	Nodo	Detalle	Plan	Método de pago	Referencia	Descuento	Cobrado
16	FRANCISCA IRENE QUISPE QUISPE	SAN FELIPE DE SEKE	10-2024 Bs. 105.00 - Pagado - 02 oct 2024, 13:17	PLAN SOLITO (17 MBPS) Bs. 105.00	Transferencia/Depósito		0.00	Bs. 105.00
17	ESTANISLAO MAMANI QUISPE	SAN FELIPE DE SEKE	10-2024 Bs. 95.00 - Pagado - 02 oct 2024, 13:34	PLAN ECONOMICO G-PON (17 MBPS) Bs. 95.00	Efectivo		0.00	Bs. 95.00
22	MIGUEL ANGEL ROSSI MIRANDA	SAN FELIPE DE SEKE	10-2024 Bs. 150.00 - Pagado - 03 oct 2024, 09:41	PLAN SOLIDARIO (20MBPS) Bs. 150.00	Transferencia/Depósito		0.00	Bs. 150.00
33	MARCELO BENJAMIN KJAPA RADA	SAN FELIPE DE SEKE	10-2024 Bs. 105.00 - Pagado - 04 oct 2024, 08:41	PLAN SOLITO (17 MBPS) Bs. 105.00	Transferencia/Depósito		0.00	Bs. 105.00
35	AYDEE MONICA MARQUEZ MAMANI	SAN FELIPE DE SEKE	10-2024 Bs. 105.00 - Pagado - 04 oct 2024, 09:47	PLAN SOLITO (17 MBPS) Bs. 105.00	Efectivo		0.00	Bs. 105.00
42	ISMAEL PACAJE MAMANI	SAN FELIPE DE SEKE	10-2024 Bs. 150.00 - Pagado - 04 oct 2024, 10:46	PLAN SOLIDARIO (20MBPS) Bs. 150.00	Transferencia/Depósito	LIZETH CHIPANA	0.00	Bs. 150.00

about:blank

1/

Código	Cliente	Nodo	Detalle	Plan	Método de pago	Referencia	Descuento	Cobrado
43	EUSEBIO CHIARA CHURA	SAN FELIPE DE SEKE	10-2024 Bs. 105.00 - Pagado - 04 oct 2024, 11:24	PLAN SOLITO (17 MBPS) Bs. 105.00	Efectivo	LIZETH CHIPANA	0.00	Bs. 105.00
44	ALAN RONALDO QUISPE CORI	SAN FELIPE DE SEKE	10-2024 Bs. 150.00 - Anulado - 04 oct 2024, 11:51	PLAN SOLIDARIO (20MBPS) Bs. 150.00	Transferencia/Depósito	ROSSIO ESCOBAR	0.00	Bs. 0.00

3.3.3.2.2. Pruebas Unitarias

Se implementaron extensas pruebas para cada funcionalidad del módulo, asegurando que cada componente funcionara correctamente bajo diferentes escenarios.

Tabla 20.*Pruebas Unitarias Finanzas*

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT3-001	getClientesCuentas	Retorna la lista de clientes con sus cuentas activas.	empresa_id	Lista de clientes con sus cuentas.
UT3-002	addCuentaLote	Registra cuentas en lote para múltiples clientes, generando notificaciones.	notificar, clientesID, mes, gestion	Cuentas registradas
UT3-003	addCuenta	Registra una nueva cuenta para un cliente específico, con opción de notificación.	notificar, cliente_id, mes, gestion, monto	Cuenta registrada y notificación enviada.
UT3-004	editCuenta	Actualiza los datos de una cuenta, incluyendo el período y el monto.	id, mes, gestion, monto	Cuenta actualizada.
UT3-005	deleteCuenta	Elimina una cuenta específica de un cliente	id	Cuenta eliminada.
UT3-006	addPago	Registra un pago para una cuenta específica.	id, fecha_pago, metodo_pago, referencia_pago	Pago registrado y cuenta marcada como pagada.
UT3-007	getPagos	Obtiene una lista de pagos realizados en un mes y año	mes, gestion	Lista de pagos filtrada por fecha y empresa.

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT3-008	deletePago	Anula un pago existente.	id	Pago y cuentas relacionadas marcadas como anuladas.

Tabla 21.

Pruebas Unitarias Asistencias

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT3-009	asistirUser	Asigna un usuario a una asistencia en estado "En espera".	id de la asistencia	Asistencia asignada al usuario y estado actualizado
UT3-010	getAsistencias	Retorna una lista de asistencias para una empresa específica en un rango de fechas.	mes, gestion	Lista de asistencias filtrada por empresa y fecha
UT3-011	getAsistenciasManager	Obtiene una lista de asistencias.	empresa_id	Lista de asistencias en proceso

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT3-012	addAsistencia	Registra una nueva asistencia para un cliente específico, con detalles de visita y tipo.	cliente_id, descripcion, estado, fecha_hora, tipo_asistencia_id	Asistencia registrada correctamente
UT3-013	editAsistencia	Actualiza los detalles de una asistencia, incluyendo observaciones e imágenes.	id, observaciones, descripcion, estado, fecha_hora, tipo_asistencia_id, imágenes	Asistencia actualizada con nueva información
UT3-014	deleteAsistencia	Elimina una asistencia y sus imágenes asociadas, si existen.	id	Asistencia eliminada

3.3.3.2.3. Pruebas de Integración

Se realizaron pruebas de integración para asegurar la funcionalidad y la interacción correcta entre los nuevos módulos y el sistema existente.

Tabla 22.

Pruebas de Integración Finanzas

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
IT3-001	addCuentaLote() + Cliente	Verifica la creación de cuentas en lote para múltiples clientes y su asociación correcta.	Datos de cuentas y lista de clientesID	Cuentas creadas y asociadas correctamente a cada cliente

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
IT3-002	addPago() + Cuenta	Verifica la creación de un pago y la actualización del estado de la cuenta asociada.	Datos del pago y id de la cuenta	Pago creado y cuenta actualizada a estado "Pagado"
IT3-003	getClientesCuentas() + Plan	Retorna una lista de clientes junto con sus cuentas y el plan correspondiente.	empresa_id	Lista de clientes con cuentas y planes asociados
IT3-004	pagosReporte() + Nodo	Genera un reporte de pagos filtrado por nodos específicos y fechas.	empresa_id, mes, gestion, nodo	Reporte de pagos generado con totales de acuerdo a los nodos
IT3-005	deletePago() + Cuenta	Verifica la eliminación de un pago y la actualización de estado de las cuentas asociadas a "Anulado".	id de pago	Pago eliminado y cuentas asociadas actualizadas a "Anulado"

Tabla 23.

Pruebas de Integración Asistencias

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
IT3-006	asistirUser() + Usuario	Asigna un usuario a una asistencia	id de asistencia, user_id	Asistencia asignada al usuario y estado actualizado

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
IT3-007	getAsistencia() + Cliente	Retorna una lista de asistencias asociadas a clientes específicos	empresa_id, mes, gestion	Lista de asistencias filtradas por cliente y fecha
IT3-008	addAsistencia() + TiposAsistencias	Verifica la creación de una asistencia para un cliente específico.	Datos de la asistencia y tipo de asistencia	Asistencia creada y tipo de asistencia asociado correctamente
IT3-009	editAsistencia() + TiposClientes	Edita una asistencia existente.	Datos de asistencia editados	Asistencia actualizada con asociaciones intactas
IT3-010	deleteAsistencia() + RegistroControl	Elimina una asistencia	id de asistencia	Asistencia eliminada

3.3.3.2.4. Pruebas de Funcionalidad

Se llevaron a cabo pruebas exhaustivas de los flujos de trabajo completos en el módulo de órdenes de trabajo para verificar que todos los procesos se ejecutaran sin interrupciones y que los datos se manejaran correctamente.

Figura 24.

Diagrama de Caso de Uso Finanzas

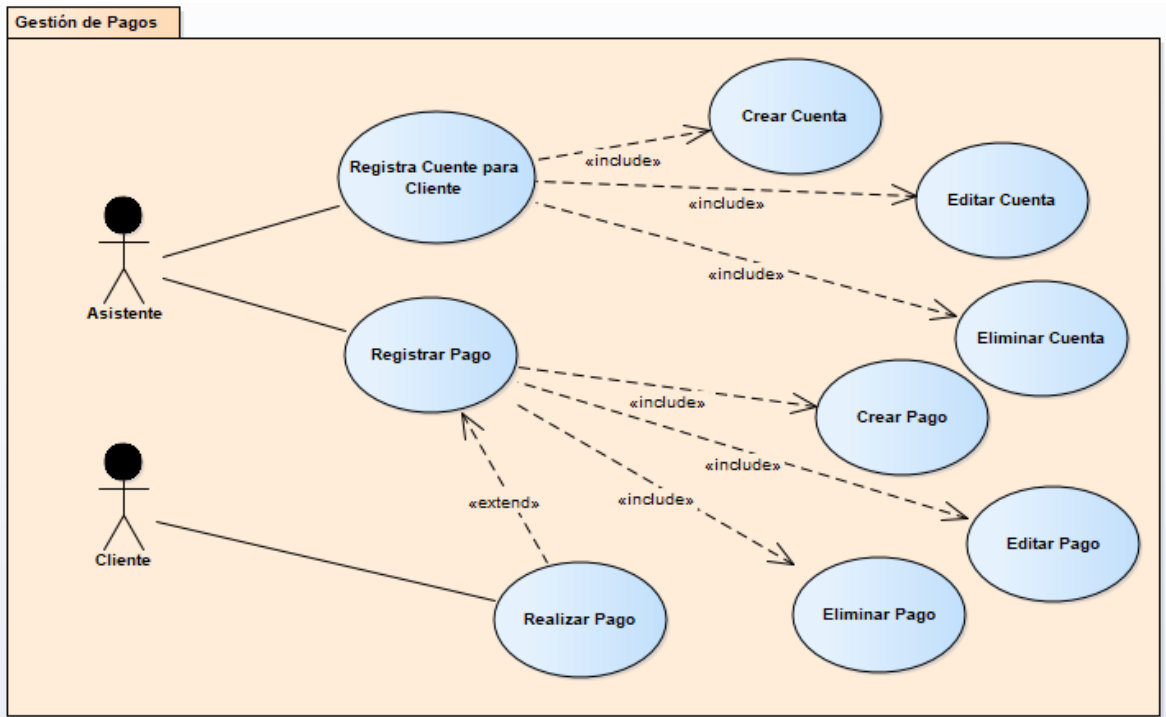
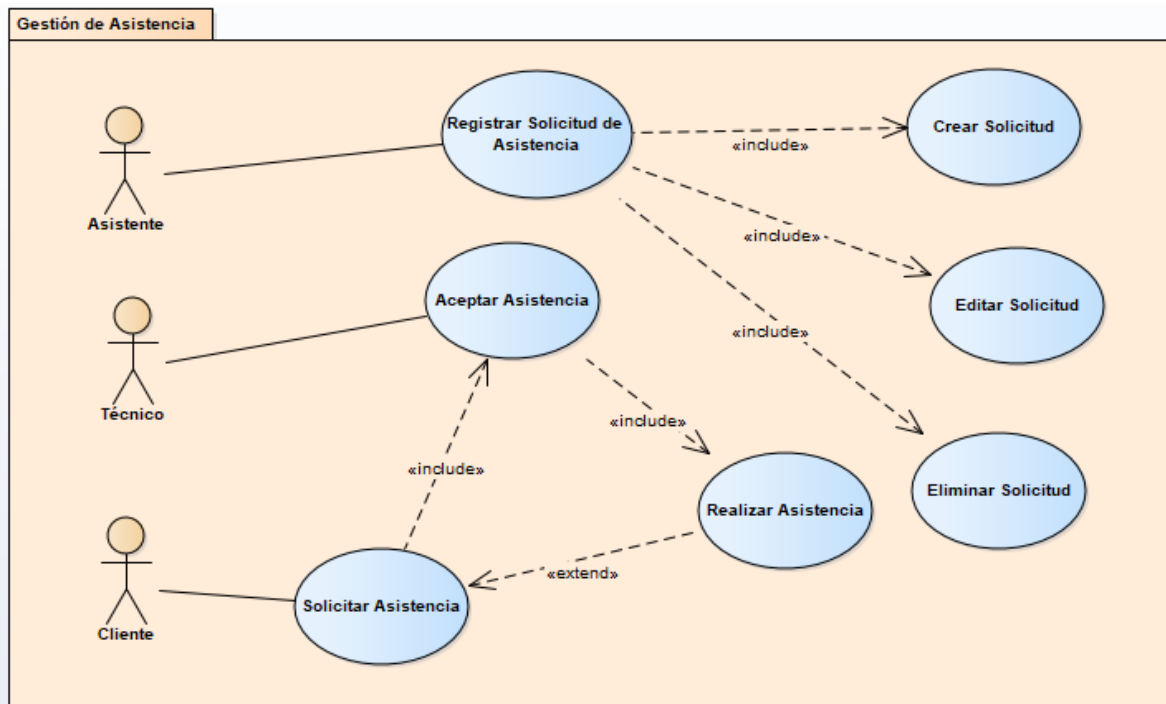


Figura 25.

Diagrama de Caso de Uso Asistencias



3.3.3.3. Revisión y Retroalimentación

3.3.3.3.1. *Recolección y Análisis de Retroalimentación*

Evaluación directa con usuarios finales para recoger comentarios específicos sobre la usabilidad y funcionalidad de los nuevos módulos.

Ajustes basados en la retroalimentación para refinar la experiencia del usuario y mejorar la eficiencia operativa.

3.3.3.4. Lanzamiento de Versión (1.2.0)

3.3.3.4.1. *Preparación para el Lanzamiento*

Se realizó una revisión final y optimización del sistema basada en los resultados de las pruebas y la retroalimentación obtenida. Se llevaron a cabo los preparativos necesarios para desplegar las nuevas funcionalidades en el entorno de producción, asegurando una configuración adecuada para un rendimiento óptimo. Además, se implementaron los módulos de Asistencias y Finanzas en el servidor de producción.

Se brindó una capacitación intensiva a los usuarios para garantizar un uso eficaz de las nuevas herramientas y reportes. Por último, se estableció un sistema de soporte continuo para atender cualquier consulta o problema que pueda surgir tras el lanzamiento.

Esta tercera iteración apunta a expandir significativamente la funcionalidad del sistema, introduciendo herramientas cruciales para la gestión de asistencias y finanzas, lo que permitirá a DAIFO TELECOM S.R.L. mejorar la eficiencia operativa y la precisión en la gestión financiera.

3.3.4. Cuarta Iteración

La cuarta iteración del proyecto se centró en el desarrollo de funcionalidades clave que impactan directamente en la interacción con los clientes y la eficiencia de los procesos

administrativos: el sistema de notificaciones por WhatsApp y la impresión de comprobantes de pago con impresora térmica.

Estas funcionalidades fueron seleccionadas debido a su importancia para mejorar la comunicación y optimizar los flujos de trabajo relacionados con las transacciones financieras y el servicio al cliente. La incorporación de notificaciones por WhatsApp permite una comunicación ágil, asegurando que los clientes reciban información relevante sobre pagos. Por su parte, la impresión de comprobantes de pago con impresora térmica agiliza el proceso de emisión de comprobantes, reduciendo tiempos de espera y aumentando la eficiencia en puntos de atención al cliente.

3.3.4.1. Planificación de Iteración

Durante la planificación de esta iteración, se definieron los metas y se priorizaron las funcionalidades necesarias para implementar de manera efectiva el sistema de notificaciones y la impresión de comprobantes.

3.3.4.1.1. Definición de Objetivos y Funcionalidades

- Notificaciones por WhatsApp: Implementación de un sistema de notificación multi-cuenta usando WhatsAppWeb JS para enviar notificaciones periódicas a clientes.
- Integración de Impresora Térmica: Desarrollo de funcionalidades para imprimir recibos y reportes directamente desde el sistema utilizando comandos ESC/POS, compatible con impresoras térmicas.
- Diseño Técnico: Adaptación del backend para manejar la gestión de notificaciones y del fronted para realizar las impresiones de los comprobantes de pagos.

Tabla 24.*Requerimiento Funciona Notificaciones*

ID	RFN-1: Requerimiento de Notificaciones por WhatsApp
Elaborado por	Abdon Cristhian
Actores	Usuario, Administrador
Descripción	Permite la implementación de un sistema de notificación multi-cuenta para enviar mensajes a clientes a través de WhatsAppWeb JS. Incluye notificaciones periódicas para avisos de cuenta, recordatorios, y otros eventos configurables en el sistema.
Restricción	Requiere la vinculación de una cuenta de WhatsApp activa con el sistema y que los números destinatarios estén registrados en WhatsApp.
Precondiciones	La cuenta de WhatsApp debe estar vinculada con el sistema, y los números de los destinatarios deben estar registrados en WhatsApp.
Incluye	Envío de notificaciones de cuenta y recordatorios de vencimiento. Filtro para seleccionar clientes específicos o grupos de clientes.
Frecuencia de uso	Moderada
Prioridad	Alta
Nota	La integración con WhatsApp permite una comunicación efectiva con los clientes, mejorando la eficiencia en los recordatorios y notificaciones.

Tabla 25.

Requerimiento Funcional Impresión de Comprobantes

ID	RFI-1: Requerimiento de Integración de Impresora Térmica
Elaborado por	Abdon Cristhian
Actores	Usuario, Administrador
Descripción	Desarrolla la funcionalidad para imprimir recibos y reportes directamente desde el sistema utilizando comandos ESC/POS, compatible con impresoras térmicas. Los reportes incluyen detalles de transacciones, recibos de pago, y otros documentos de uso frecuente.
Restricción	La impresora térmica debe estar configurada correctamente y contar con papel térmico.
Precondiciones	La impresora debe estar conectada al sistema y configurada para aceptar comandos ESC/POS.
Incluye	Función de impresión directa de recibos y reportes financieros. Configuración de formato y compatibilidad con múltiples impresoras térmicas.
Frecuencia de uso	Alta
Prioridad	Media
Nota	La integración con impresoras térmicas facilita la generación de documentos físicos, mejorando el servicio al cliente y la gestión administrativa.

3.3.4.2. Desarrollo y Pruebas

Se implementó el sistema de notificaciones utilizando WhatsAppWeb JS, para lo cual se creó un controlador dedicado que gestiona eficazmente el envío de notificaciones, permitiendo la configuración y administración de múltiples cuentas.

Figura 26.

Métodos de Notificaciones

```
const WhatsController = {
  getStatusWhatsApp: async (io, empresa_id) => {...
  },
  logoutWhatsApp: async (io, empresa_id) => {...
  },
  sendMessageWhatsApp: async (io, empresa_id, celular) => {...
  },

  enviarNotificacionCuenta: async (io, clientes) => {...
  }
};

module.exports = WhatsController;
```

Figura 27.

Vinculación de WhatsApp



Figura 28.

Selección de Impresora Térmica

Impresora Termica

AnyDesk Printer ▼

IMPRIMIR TEST

CONTROLADOR

Figura 29.

Comprobante de Pago Impreso

DaifoTelecom
 NIT: 288188020
 DIR: Av. Gral. Juan José Torrez N° 444, La Paz
 TELÉFONOS: 2215505 / 76772794
 WEB: www.daifotelecom.net

NRO COMPROBANTE: 327
 ZONA: SAN FELIPE DE SEKE 2
 CLIENTE: DORIS ROXANA CAZU LDAYZA
 C?DIGO: 1186
 PLAN: PLAN SOLITO (17 MBPS)
 TELÉFONO: 79536462

SERVICIO DE INTERNET		
NRO.	PERIODO	MONTO
1	11-2024	Bs. 105.00

SUBTOTAL: Bs. 105.00
 DESCUENTO: Bs. 0.00
 TOTAL: Bs. 105.00

SON: Ciento Cinco BOLIVIANOS 00/100
 FECHA DE PAGO: 2024-11-06
 MÉTODO DE PAGO: Transferencia/Depósito




Asimismo, se llevó a cabo la integración y configuración de la impresora térmica, permitiendo impresiones directas desde el software. Esto incluyó la generación de imágenes, como logotipos y códigos QR, necesarios para los comprobantes, garantizando así la personalización y profesionalismo en la documentación emitida.

3.3.4.2.1. Pruebas Unitarias

Se implementaron extensas pruebas para cada funcionalidad del módulo, asegurando que cada componente funcionara correctamente bajo diferentes escenarios.

Tabla 26.

Pruebas Unitarias Notificaciones

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT4-001	getStatusWhatsApp()	Verifica si el estado de WhatsApp se obtiene correctamente y envía el estado al cliente.	empresa_id	Estado de conexión enviado al cliente.
UT4-002	logoutWhatsApp()	Verifica el cierre de sesión de la cuenta de WhatsApp.	empresa_id	Estado de desconexión enviado al cliente.
UT4-003	sendMessageWhatsApp()	Envía un mensaje de prueba a un número de cliente registrado en WhatsApp.	empresa_id, celular	Mensaje enviado con éxito.

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT4-004	enviarNotificacion Cuenta()	Envía notificación de cuenta a una lista de clientes con cuentas activas.	io, clientes	Notificación enviada a clientes activos.

Tabla 27.

Pruebas Unitarias Impresión de Comprobantes

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
UT4-005	addPagoPos()	Registra un pago en el sistema y genera el comprobante.	Datos del pago y cuentas seleccionadas	Pago registrado y comprobante generado.
UT4-006	imprimirComprobante()	Imprime el comprobante en una impresora térmica usando el conector ESC/POS.	item (datos del pago)	Comprobante impreso correctamente en la impresora.

3.3.4.2.2. Pruebas de Integración

Se realizaron pruebas de integración para asegurar la funcionalidad y la interacción correcta entre los nuevos módulos y el sistema existente.

Tabla 28.*Pruebas de Integración Notificaciones por WhatsApp*

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
IT4-001	sendMessageWhatsApp()	Verifica que se envíe una notificación a un número.	empresa_id, celular	Mensaje enviado exitosamente.
IT4-002	enviarNotificacionnCuenta() + Cliente + Cuenta	Envía una notificación a los clientes	io, clientes	Notificación enviada a todos los clientes válidos.
IT4-003	getStatusWhatsApp()	Verifica el estado de WhatsApp	empresa_id	Estado de conexión obtenido.

Tabla 29.*Pruebas de Integración Impresora Térmica*

ID Prueba	Función	Descripción	Entrada	Resultado Esperado
IT4-004	addPagoPos() + Cuenta + Cliente	Verifica el registro de pago y la generación de comprobante, incluyendo datos de cuenta y cliente.	Datos del pago, cuentas seleccionadas	Pago registrado y comprobante generado con datos correctos.
IT4-005	imprimirComprobante() + Empresa	Imprime el comprobante de pago en la impresora térmica con los datos de la empresa y cliente correctamente.	item (datos del pago)	Comprobante impreso correctamente en la impresora.

En cuanto a la impresión, se configuró correctamente una impresora térmica de 80 mm con sus respectivos controladores y su integración al sistema. Además, se añadió una interfaz que permite seleccionar la impresora deseada, facilitando la personalización y el control de las impresiones dentro del sistema. También se verificó el correcto funcionamiento de las impresiones, incluyendo la generación de comprobantes con logotipos y códigos QR para una mejor presentación.

3.3.4.2.3. Pruebas de Funcionalidad

Se llevaron a cabo pruebas exhaustivas de los flujos de trabajo completos en el módulo de órdenes de trabajo para verificar que todos los procesos se ejecutaran sin interrupciones y que los datos se manejaran correctamente.

Figura 30.

Diagrama de Caso de Uso Notificaciones

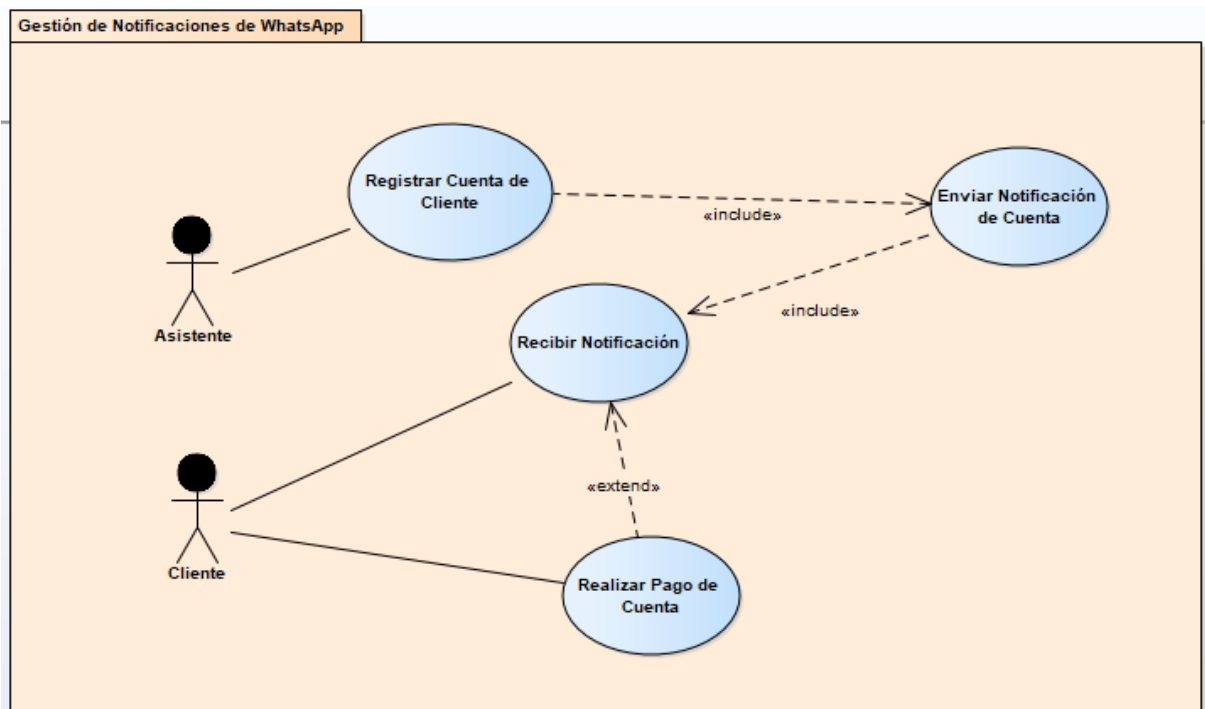
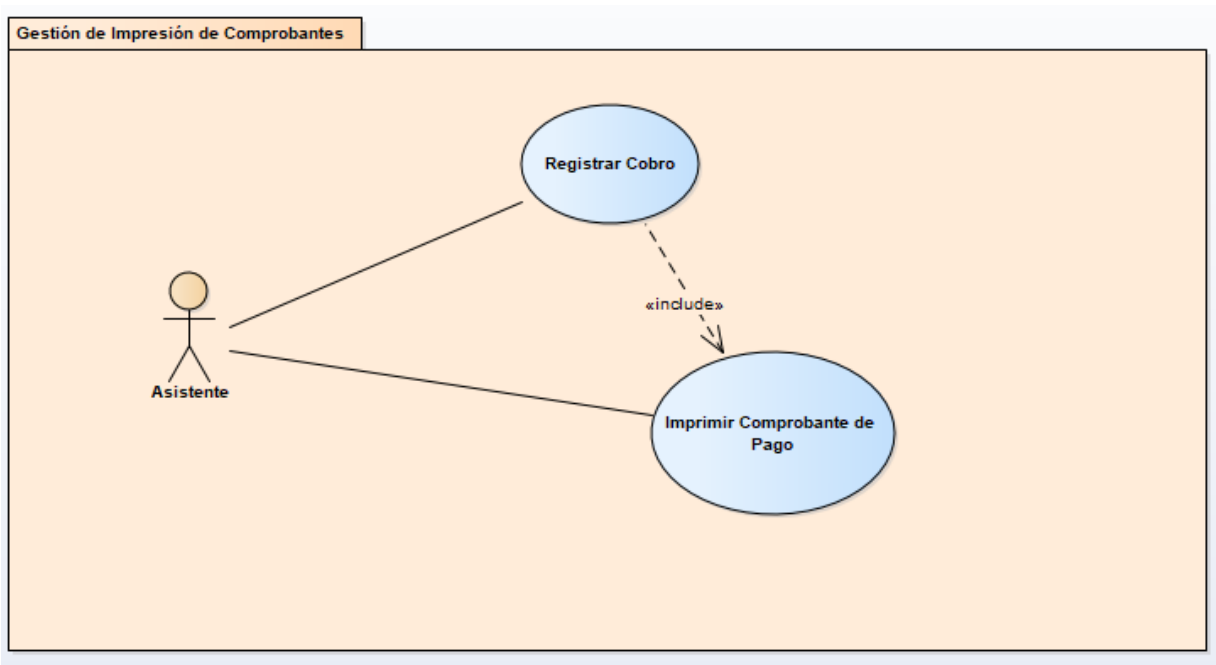


Figura 31.

Diagrama de Caso de Uso Impresión de Comprobantes



3.3.4.3. Revisión y Retroalimentación

Se llevó a cabo un proceso de recolección y análisis de retroalimentación, centrado en evaluar la efectividad del sistema de notificación. También se realizó una evaluación detallada del rendimiento y la usabilidad de la impresión térmica dentro de los flujos operativos del negocio.

En base a la retroalimentación recibida y los resultados de las pruebas, se realizaron ajustes y mejoras en las integraciones del sistema, optimizando su funcionamiento para garantizar una experiencia más fluida y eficiente para los usuarios.

3.3.4.4. Lanzamiento de Versión (1.3.0)

3.3.4.4.1. Preparativos para el Lanzamiento

Se realizó una revisión final y optimización de las nuevas integraciones para asegurar su correcto funcionamiento. Además, se preparó la documentación y los recursos de soporte necesarios para el lanzamiento.

3.3.4.4.2. Lanzamiento de la Versión 1.3.0

Se desplegó la versión actualizada del sistema, que incluye nuevas funcionalidades para la gestión de notificaciones a través de WhatsApp y la impresión térmica. Se llevó a cabo una capacitación integral y se brindó soporte a los usuarios para facilitar una transición suave y efectiva hacia las nuevas herramientas.

3.3.4.4.3. Documentación y Recursos Adicionales

Se incluyeron detalles técnicos relevantes sobre las nuevas funcionalidades implementadas, así como instrucciones para su configuración y uso, con el fin de facilitar la adopción por parte de los usuarios y el mantenimiento del sistema.

Esta cuarta iteración introduce mejoras significativas que optimizan la interacción con el cliente y la eficiencia operativa, incrementando la capacidad del sistema para manejar tareas críticas y condiciones específicas en tiempo real.

3.3.5. Quinta Iteración

3.3.5.1. Planificación de Iteración

En esta iteración, se evaluó la necesidad de migrar de un entorno completamente web a la implementación de aplicaciones nativas para Windows y Android, con el objetivo de mejorar la experiencia del usuario y la eficiencia en las operaciones.

3.3.5.1.1. Definición de Objetivos y Funcionalidades

El objetivo principal de esta iteración fue desarrollar aplicaciones nativas que permitan al personal de DAIFO TELECOM S.R.L. tener acceso completo a las funcionalidades del sistema, tanto en dispositivos de escritorio con Windows como en dispositivos móviles con Android, sin depender exclusivamente de la versión web.

Tabla 30.*Requerimiento Funcional Aplicación Windows*

ID	RFON-1: Requerimiento Aplicación Windows
Elaborado por	Abdon Cristhian
Actores	Personal de DAIFO TELECOM S.R.L.
Descripción	Desarrollar una aplicación nativa para Windows que permita el acceso completo a las funcionalidades del sistema.
Restricción	Solo usuarios con credenciales pueden acceder a la aplicación.
Precondiciones	La aplicación debe estar instalada en el sistema operativo Windows.
Incluye	Funcionalidades completas del sistema, acceso a bases de datos y gestión de usuarios.
Frecuencia de uso	Alta
Prioridad	Alta

Tabla 31.*Requerimiento Funcional Aplicación Android*

ID	RFON-2: Requerimiento Aplicación Android
Elaborado por	Abdon Cristhian
Actores	Personal de DAIFO TELECOM S.R.L.
Descripción	Desarrollar una aplicación nativa para Android que permita el acceso completo a las funcionalidades del sistema desde dispositivos móviles.

Restricción	Solo usuarios con credenciales válidas pueden acceder a la aplicación.
Precondiciones	La aplicación debe estar instalada en un dispositivo con sistema operativo Android.
Incluye	Funcionalidades completas del sistema, acceso a bases de datos y gestión de usuarios.
Frecuencia de uso	Alta
Prioridad	Alta

3.3.5.2. Desarrollo y Pruebas

Para la plataforma Windows, se utilizó Electron, un framework que permite crear aplicaciones de escritorio multiplataforma utilizando tecnologías web como HTML, CSS y JavaScript. Con Electron, se generó un instalable completo del sistema, lo cual incluyó la integración de todas las funcionalidades del sistema, desde la gestión de órdenes de trabajo hasta el envío de notificaciones y la impresión de comprobantes. El instalable se diseñó para ofrecer una experiencia de usuario consistente y optimizada. Además, se realizaron pruebas exhaustivas para garantizar la estabilidad y el rendimiento del instalable en diferentes entornos de Windows.

Figura 32.

Aplicativo Escorp Windows

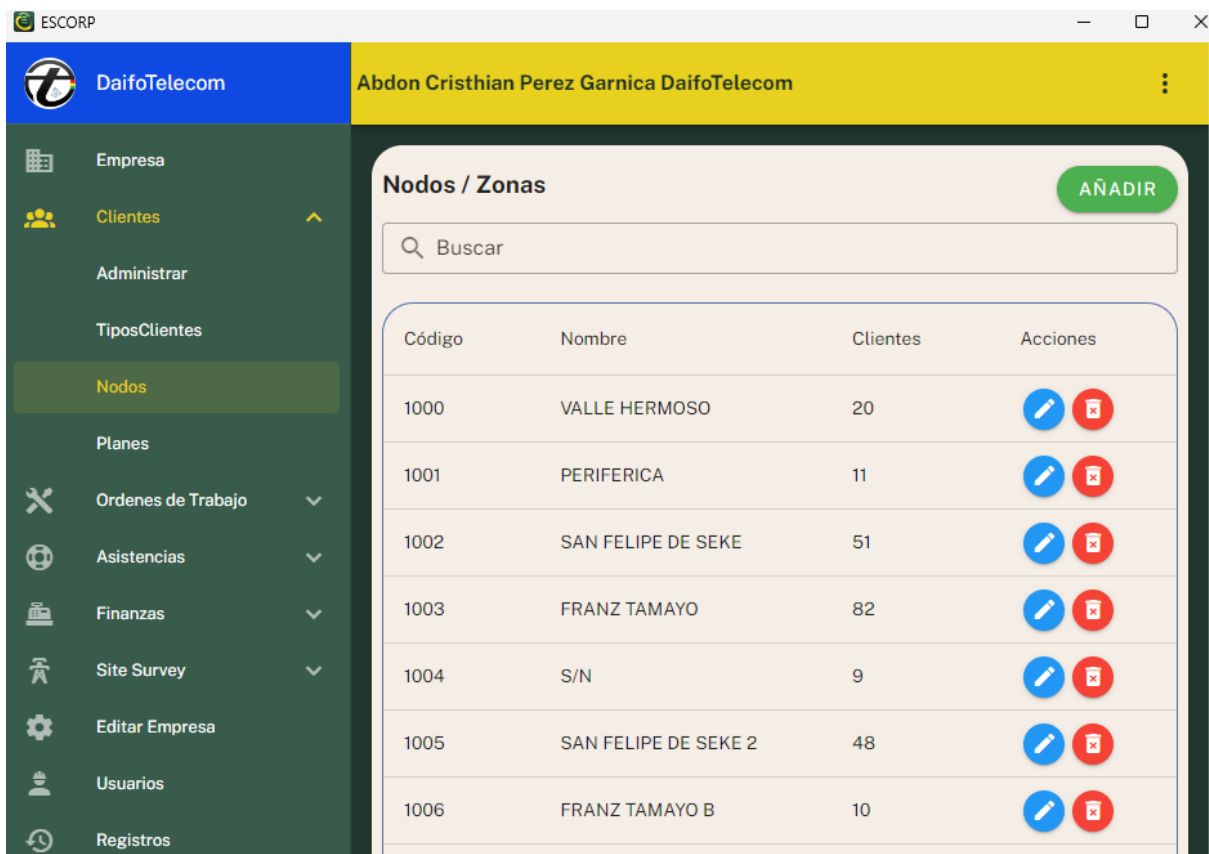
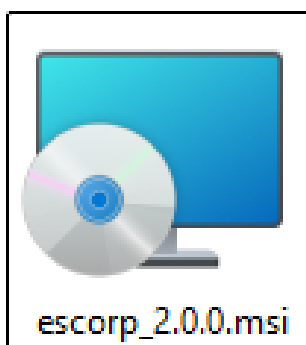


Figura 33.

Instalador Escorp Windows

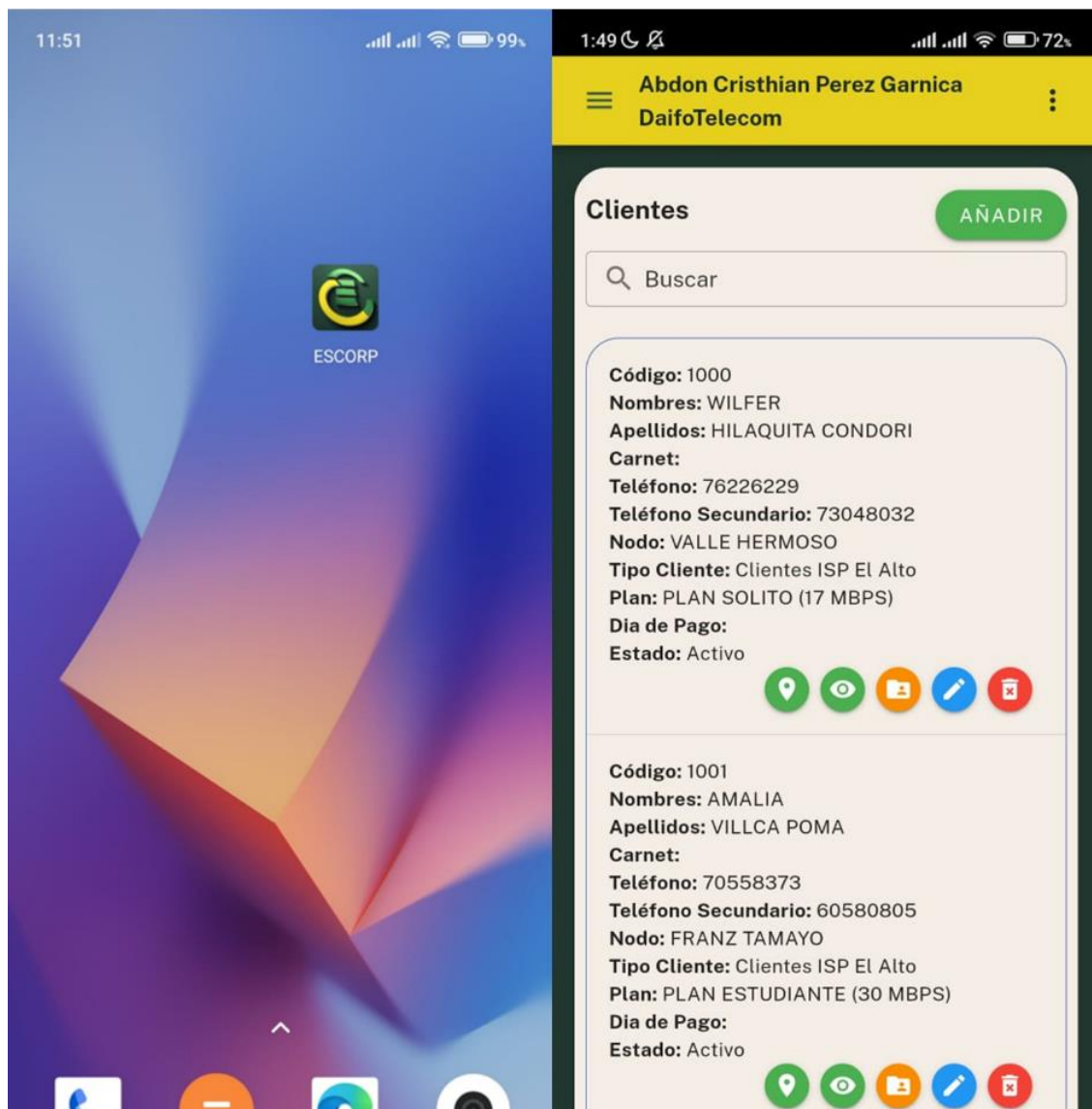


En cuanto a la plataforma Android, se desarrolló una aplicación móvil utilizando Apache Cordova, una herramienta que permite crear aplicaciones híbridas mediante tecnologías web. La aplicación de Android integró librerías específicas para gestionar características propias de los dispositivos móviles, como los permisos de ubicación y acceso a la cámara. Se generó un

archivo APK instalable, que se distribuyó al personal operativo, permitiéndoles acceder al sistema y realizar sus tareas en campo de manera ágil y eficaz. La aplicación se sometió a rigurosas pruebas en diversos dispositivos Android para garantizar su compatibilidad y estabilidad, y se verificó la correcta administración de los permisos para asegurar un funcionamiento seguro y conforme a las políticas de la empresa.

Figura 34.

Aplicativo Escorp Android



3.3.5.3. Revisión y Retroalimentación

Se llevó a cabo un proceso de revisión integral de ambas aplicaciones, recopilando retroalimentación del personal operativo y administrativo. Esta fase incluyó pruebas de usabilidad para asegurar que la experiencia de usuario fuera intuitiva y alineada con los flujos de trabajo de la empresa. También se verificó que las aplicaciones cumplieran con los objetivos de la iteración, optimizando las funcionalidades según la retroalimentación recibida.

3.3.5.4. Lanzamiento de Versión (2.0.0)

Dado que esta iteración implicó un cambio significativo en la arquitectura del sistema, se decidió avanzar a la versión 2.0.0. Con este lanzamiento, DAIFO TELECOM S.R.L. ahora cuenta con dos aplicaciones nativas (para Windows y Android) y un sistema web auxiliar, diseñado para servir como respaldo en caso de que algún miembro del personal no tenga acceso a los aplicativos móviles o de escritorio.

Figura 35.

Código raíz aplicativo Android

```
<name>ESCORP</name>
<content src="index.html" />
<allow-intent href="http://*/*" />
<allow-intent href="https://*/*" />

<preference name="AutoHideSplashScreen" value="true" />
<preference name="SplashScreen" value="none" />
<preference name="exit-on-suspend" value="false" />
<preference name="SplashScreenDelay" value="0" />
<platform name="android">

    <preference name="AndroidWindowSplashScreenAnimatedIcon"
value="res/imagen.png" />
    <preference name="AndroidWindowSplashScreenBackground"
value="#395b4b" />
    <preference name="android-minSdkVersion" value="21" />
</platform>
```

3.3.6. Sexta Iteración

3.3.6.1. Planificación de Iteración

En esta iteración, el objetivo principal fue implementar un sistema de monitoreo en tiempo real para las torres de comunicación, permitiendo observar el estado de las antenas mediante la captura y transmisión de datos de varios sensores.

3.3.6.1.1. Definición de Objetivos y Funcionalidades

Monitoreo de parámetros críticos como la temperatura, humedad, niveles de lluvia y el estado estructural (medido con un giroscopio y un acelerómetro) de las torres de comunicación.

Implementación de un sistema de transmisión de datos desde las torres de comunicación a través de un ESP32 y sensores, con visualización de los datos en tiempo real en el sistema frontend utilizando WebSockets.

Tabla 32.

Requerimiento Funcional Monitoreo de Torres

ID	RFMON-1: Requerimiento Funcional Monitoreo de Torres
Elaborado por	Abdon Cristhian
Actores	Personal de Monitoreo y Mantenimiento de DAIFO TELECOM S.R.L.
Descripción	Implementar un sistema de monitoreo en tiempo real para las torres de comunicación, capturando datos de sensores en las antenas. Monitorear parámetros críticos como temperatura, humedad, niveles de lluvia, y estado estructural mediante giroscopio y acelerómetro.

Restricción	Solo usuarios autorizados pueden acceder al monitoreo en tiempo real y modificar parámetros de alerta.
Precondiciones	Las torres deben estar equipadas con ESP32 y sensores adecuados. Los datos deben estar disponibles desde los sensores en las torres y ser transmitidos al sistema.
Incluye	Visualización en tiempo real en el frontend usando WebSockets, configuración de límites críticos, y sistema de alertas automáticas.
Frecuencia de uso	Permanente durante el monitoreo de las torres, con prioridad en condiciones climáticas adversas o eventos sísmicos.
Prioridad	Alta
Nota	El sistema debe alertar al personal ante valores críticos y guardar un historial de datos para cada parámetro crítico monitoreado.

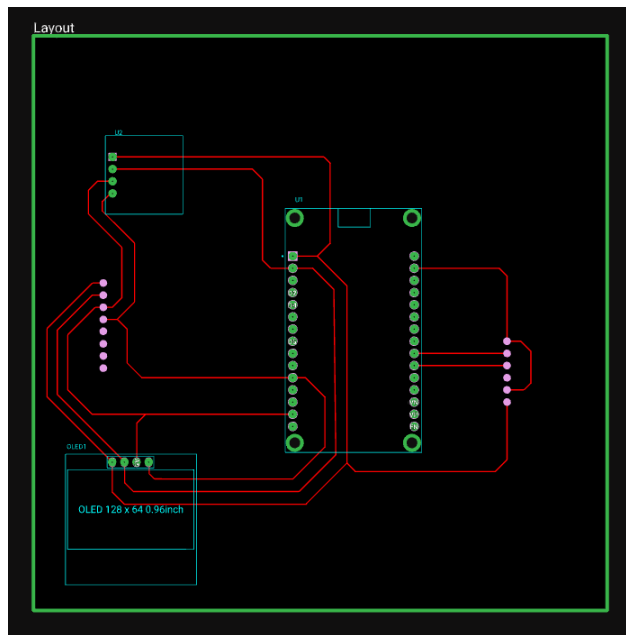
3.3.6.2. Desarrollo y Pruebas

3.3.6.2.1. Implementación de sensores al módulo ESP32

Se implementaron los sensores BME280 para la medición de temperatura, humedad y presión atmosférica, el MPU-6050 para la medición de la inclinación y vibraciones utilizando un giroscopio y un acelerómetro, y el YL-83 para la detección de la presencia de lluvia. Como microcontrolador central, se utilizó un ESP32 para recoger los datos de estos sensores y transmitirlos mediante el módulo HTTP a un endpoint del backend desarrollado en Node.js. La transmisión de datos se programó para enviarse cada 5 segundos, y el ESP32 fue alimentado con un adaptador de 12 V y 5 amperios, proporcionando una fuente de energía estable para el correcto funcionamiento de todos los componentes.

Figura 36.

Conexión de sensores al microcontrolador

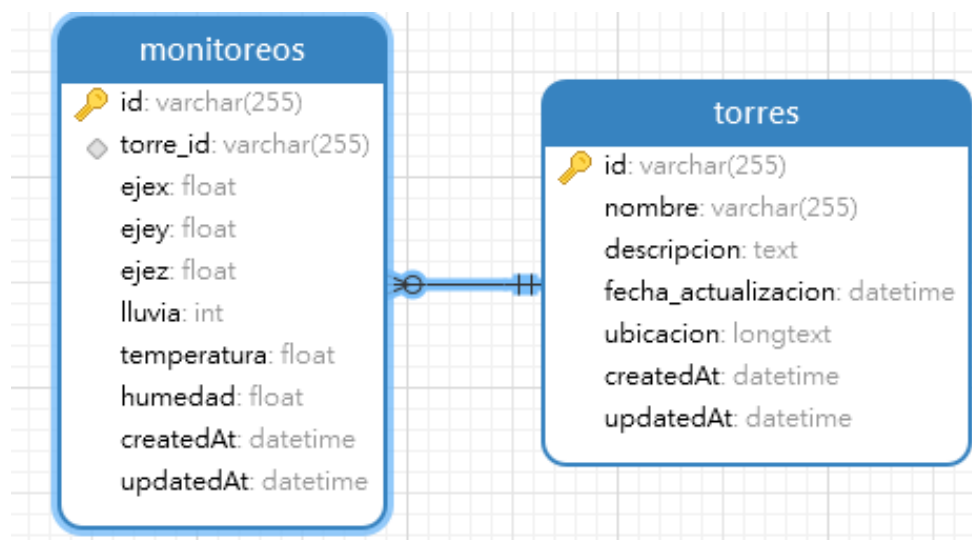


3.3.6.2.2. Recepción y Almacenamiento de Datos

En el backend, se creó un endpoint para recibir los datos de los sensores. Estos datos fueron actualizados en una tabla de la base de datos que incluye los siguientes campos:

Figura 37.

Diseño de Base de Datos Parcial (6° iteración)



3.3.6.2.3. Visualización en Tiempo Real

Para el frontend, se utilizaron WebSockets para establecer una conexión bidireccional entre el servidor y el cliente. Esto permitió la actualización en tiempo real de los datos visualizados en la interfaz, brindando al personal técnico de DAIFO TELECOM S.R.L. la posibilidad de monitorear continuamente el estado de las torres y reaccionar de manera proactiva ante cualquier irregularidad detectada.

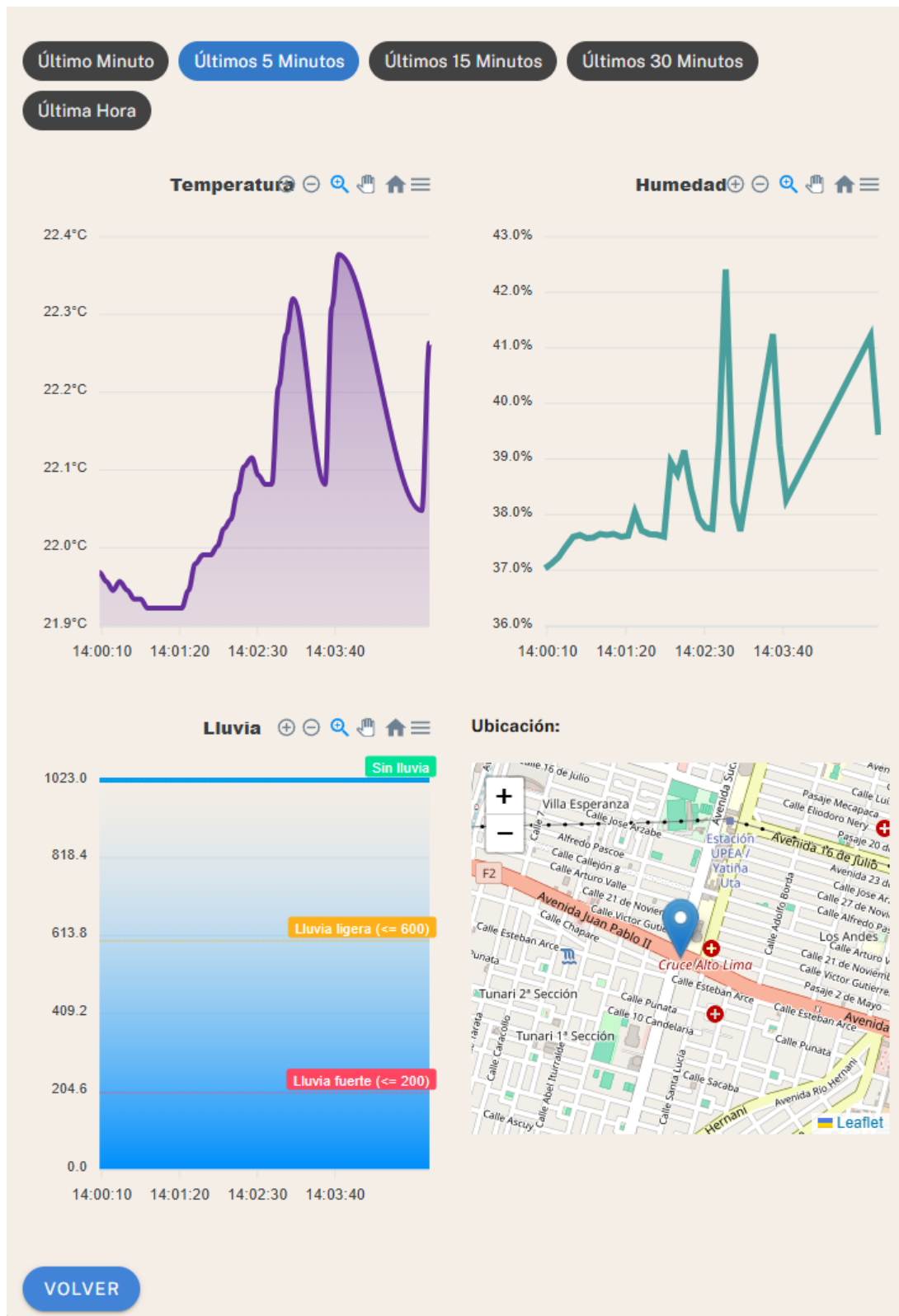
Figura 38.

Interfaz de Monitoreo en Tiempo Real (Estructural)



Figura 39.

Interfaz de Monitoreo en Tiempo Real (Ambiental)



3.3.6.3. Revisión y Retroalimentación

Se realizaron pruebas intensivas del sistema para validar la precisión de los datos recopilados y la estabilidad de la transmisión en tiempo real. Además, se solicitó retroalimentación del equipo técnico encargado del monitoreo para ajustar la interfaz y mejorar la presentación de los datos críticos.

3.3.6.4. Lanzamiento de Versión (2.1.0)

En esta iteración, se lanzó la versión 2.1.0, que incluye la funcionalidad de monitoreo en tiempo real de las torres de comunicación. Con esta nueva versión, el sistema permite no solo gestionar las operaciones y asistencias, sino también vigilar el estado de la infraestructura de manera constante, optimizando la seguridad y el rendimiento del servicio.

3.4. Revisión de Versión Final

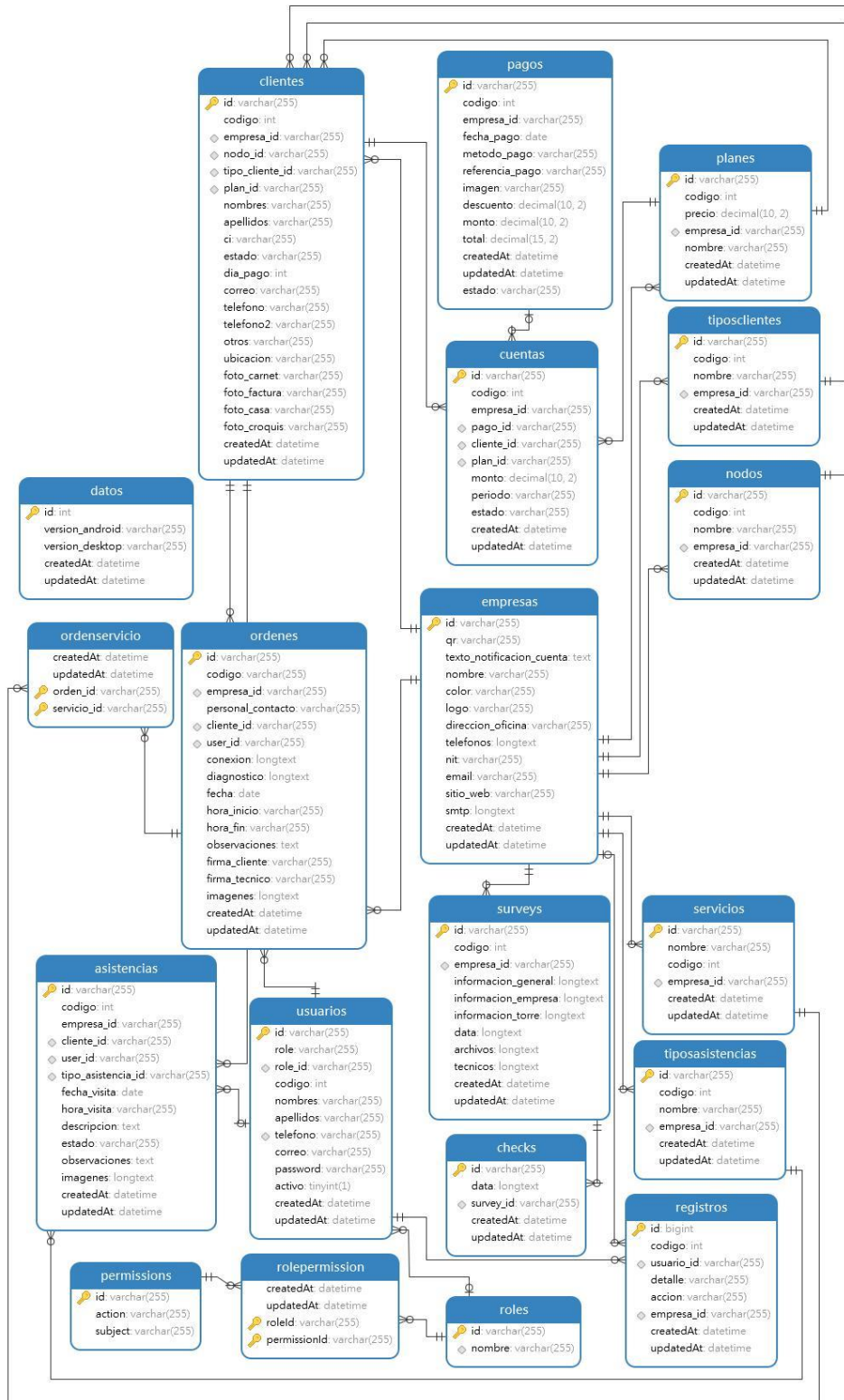
Con esta sección, se concluye el desarrollo del sistema siguiendo la Metodología Iterativa e Incremental, la cual permitió dividir el proyecto en múltiples iteraciones que facilitaron la integración progresiva de todas las funcionalidades clave. Cada iteración se centró en objetivos específicos, abordando de manera ordenada el desarrollo, prueba y refinamiento de cada módulo del sistema. Esta metodología no solo permitió una mayor flexibilidad para adaptarse a nuevos requisitos y cambios, sino que también posibilitó realizar ajustes continuos basados en la retroalimentación obtenida, garantizando así la estabilidad y funcionalidad del sistema a medida que se completaban las iteraciones.

3.4.1. Estructura de la Base de Datos

A continuación, se detalla la estructura final de la base de datos, mostrando las relaciones y campos clave para la gestión de usuarios, asistencias, órdenes de trabajo, monitoreo de torres, roles, entre otros.

Figura 40.

Diagrama de Base de Datos Final

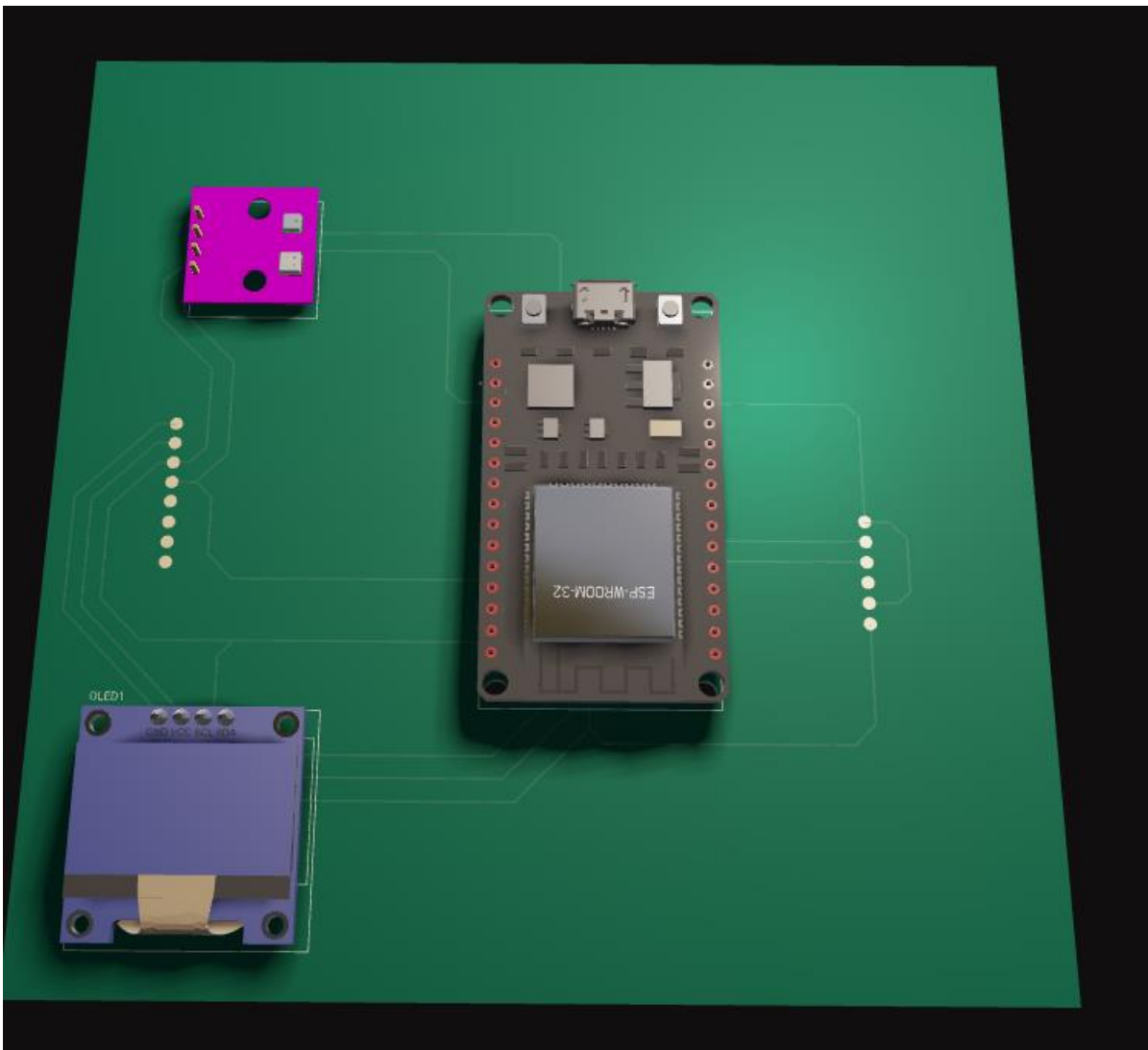


3.4.2. Diagrama de Conexión del Sistema de Monitoreo

Se proporciona también un diagrama de conexión que ilustra cómo los diferentes sensores, el ESP32 y el backend trabajan en conjunto para recoger, procesar y visualizar los datos de monitoreo en tiempo real.

Figura 41.

Modelado 3D del diseño de PBC



Con esta revisión final, se concluye satisfactoriamente el capítulo dedicado al desarrollo del sistema.

CAPÍTULO IV

CALIDAD, COSTO Y SEGURIDAD

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



4. CALIDAD, COSTO Y SEGURIDAD

4.1. Introducción

En este capítulo se abordarán los aspectos fundamentales relacionados con la calidad, el costo, la seguridad y las pruebas del sistema desarrollado para DAIFO TELECOM S.R.L. A lo largo del proceso de desarrollo, se han implementado una serie de metodologías y métricas con el objetivo de asegurar que el sistema cumpla con los estándares de calidad requeridos, sea seguro en su operación y ofrezca un rendimiento eficiente y confiable.

La calidad del sistema se evaluará mediante el análisis de las métricas clave que permiten medir la efectividad y el cumplimiento de los objetivos establecidos. Se detallarán los costos asociados al proyecto, incluyendo tanto los recursos humanos como las tecnologías y componentes utilizados para su implementación. Además, se describirán las estrategias y medidas de seguridad implementadas para proteger los datos y la información sensible, tanto en el almacenamiento como en la transmisión, garantizando la confidencialidad e integridad del sistema.

Por último, se presentarán los resultados de las pruebas realizadas a lo largo de cada iteración. Estas pruebas incluyeron verificaciones funcionales, de integración y de rendimiento, así como pruebas específicas para evaluar la fiabilidad del sistema bajo diferentes condiciones. Con este enfoque integral, se busca garantizar que el sistema desarrollado no solo cumpla con los requerimientos técnicos, sino que también se ajuste a las expectativas operativas y de seguridad de DAIFO TELECOM S.R.L.

4.2. Métricas de Calidad

En el desarrollo de un sistema de gestión para **DAIFO TELECOM S.R.L.**, las métricas de calidad desempeñan un papel fundamental para asegurar que el software cumpla con los objetivos y requerimientos específicos del negocio. Las métricas proporcionan criterios

objetivos para evaluar la efectividad del sistema en diferentes aspectos clave, como su precisión, robustez, facilidad de uso, eficiencia en el manejo de recursos, mantenimiento y capacidad de adaptación en distintos entornos. De acuerdo con la norma **ISO/IEC 25010**, se han seleccionado y aplicado diversas métricas de calidad relevantes al proyecto.

4.2.1. Funcionalidad

La funcionalidad del sistema desarrollado para DAIFO TELECOM S.R.L. se evaluó mediante la Cobertura de Requisitos, que mide el porcentaje de requisitos implementados respecto al total planificado. Esta métrica garantiza que el sistema cumpla con las funcionalidades clave especificadas para la gestión operativa y administrativa.

A continuación, se presenta una tabla simplificada que agrupa los requisitos funcionales del sistema por módulo:

Tabla 33.

Requisitos Funcionales

Módulo	Descripción
Gestión de Usuarios	Creación, lectura, actualización y eliminación de usuarios con asignación de roles en etapas posteriores.
Gestión de Empresas	Administración de empresas, incluyendo creación, edición, eliminación y asociación con clientes y servicios.
Gestión de Planes	Creación, modificación y eliminación de planes, con asociación a empresas y clientes.
Gestión de Nodos	Gestión de nodos de red vinculados a empresas y clientes según la cobertura.
Gestión de Servicios	Administración de servicios ofrecidos a clientes y empresas.

Tipos de Clientes	Gestión de tipos de clientes como residencial o empresarial.
Gestión de Clientes	Creación, modificación y eliminación de clientes, asignándoles nodos, planes y tipos de cliente.
Órdenes de Trabajo	Creación, seguimiento y finalización de órdenes de trabajo, con asignación de servicios y clientes.
Site Surveys	Registro, edición y gestión de encuestas de sitio en el campo.
Roles y Permisos	Creación, edición y eliminación de roles y asignación de permisos.
Asistencias	Gestión y asignación de asistencias, permitiendo la creación, edición y eliminación de registros de asistencia.
Finanzas	Gestión de cuentas y registro de pagos de clientes.
Notificaciones	Sistema de notificación por WhatsApp para avisos de cuenta y recordatorios.
Impresora Térmica	Funcionalidad para imprimir recibos y reportes desde el sistema usando impresoras térmicas.
Aplicaciones Nativas	Acceso a funcionalidades en dispositivos de escritorio (Windows) y móviles (Android).
Monitoreo en Tiempo Real	Monitoreo de parámetros críticos en torres de comunicación, visualizado en tiempo real.

Para calcular la cobertura de requisitos, se utilizó la fórmula:

$$\text{Cobertura de Requisitos} = \frac{\text{Número de Requisitos Implementados}}{\text{Número Total de Requisitos}} \times 100 \quad (8)$$

$$= \frac{18}{18} \times 100 = 100\%$$

La Cobertura de Requisitos del 100% demuestra que el sistema cumple con todos los requisitos funcionales planificados, garantizando una funcionalidad robusta y completa en cada módulo, alineada con los objetivos operativos de DAIFO TELECOM S.R.L.

4.2.2. Confiabilidad

La confiabilidad es un aspecto crucial en sectores como las telecomunicaciones, donde los sistemas deben garantizar un servicio ininterrumpido y confiable. Para evaluar la confiabilidad de nuestro sistema, se empleará la métrica de Tasa de Fallos.

$$\text{Tasa de Fallos} = \frac{\text{Número de Fallos Reportados}}{\text{Tiempo de Operación Total (días)}} = \frac{5}{132} = 0.037 \quad (9)$$

$$\begin{aligned} \text{Confiabilidad\%} &= 100 - (\text{Tasa de Fallos} \times 100) = 100 - (0.037 \times 100) = 100 - 3.7 \quad (10) \\ &= 96.3\% \end{aligned}$$

En el periodo desde el inicio del sistema el 2024-06-21 hasta finales de octubre 2024, con un total de 132 días de operación y 5 fallos reportados, hemos determinado las siguientes métricas:

- Tasa de Fallos: 0.037
- Tiempo Medio entre Fallos (MTBF): 15 días
- Tiempo Medio de Reparación (MTTR): 12 horas

Estas métricas indican que, en promedio, el sistema experimenta un fallo cada 15 días, con un tiempo de reparación de 12 horas por incidente.

La evaluación de la confiabilidad es fundamental para tomar decisiones informadas sobre la implementación de mejoras y redundancias. Mejorar estas métricas no solo aumentará la confianza de los usuarios en la estabilidad del servicio, sino que también contribuirá a reducir los costos operativos asociados con las reparaciones y el tiempo de inactividad del sistema.

4.2.3. Usabilidad

La usabilidad se refiere a la facilidad con la que los usuarios pueden interactuar con el sistema y completar las tareas requeridas. Es un componente esencial para la experiencia del usuario, especialmente en sistemas de gestión como el desarrollado para DAIFO TELECOM S.R.L., donde los usuarios necesitan ejecutar sus actividades con rapidez y precisión. La métrica de usabilidad se calcula de la siguiente manera:

$$\text{Tasa de Error} = \frac{\text{Número de Errores Cometidos}}{\text{Total de Acciones Realizadas}} = \frac{4}{1559} = 0.0025 \quad (11)$$

$$\text{Usabilidad\%} = 100 - (\text{Tasa de Error} \times 100) = 100 - 0.25 = 99.75\% \quad (12)$$

Con un total de 1559 acciones realizadas y solo 4 errores reportados, la Tasa de Error del Usuario es aproximadamente 0.25%. Este bajo porcentaje indica una alta usabilidad del sistema, permitiendo a los usuarios realizar sus tareas con mínima interrupción debido a errores. Estos errores incluyen desviaciones del flujo correcto de datos y el uso de caracteres no permitidos en algunos campos, lo que sugiere áreas específicas para mejorar la validación de entrada y la guía del usuario dentro del sistema.

4.2.4. Eficiencia

La eficiencia del sistema evalúa cómo se utilizan los recursos del hardware, como la CPU y la memoria, para realizar tareas específicas. En un entorno donde el rendimiento y la estabilidad son cruciales, es esencial que el software minimice el consumo de estos recursos para evitar sobrecargas y mantener un rendimiento óptimo.

El servidor que aloja el sistema opera con un procesador AMD EPYC-Milan de 2 núcleos y está gestionado por Plesk Obsidian v18.0.64 sobre Ubuntu 22.04.5 LTS. Hasta la fecha, el sistema ha mantenido un tiempo de actividad de 76 días, indicando estabilidad y resistencia operacional.

4.2.4.1. Uso de CPU

La eficiencia en el uso de la CPU se monitorea a través de la carga promedio, que indica el número de procesos en estado de ejecución o esperando activamente por la CPU.

Los valores recientes promedio son:

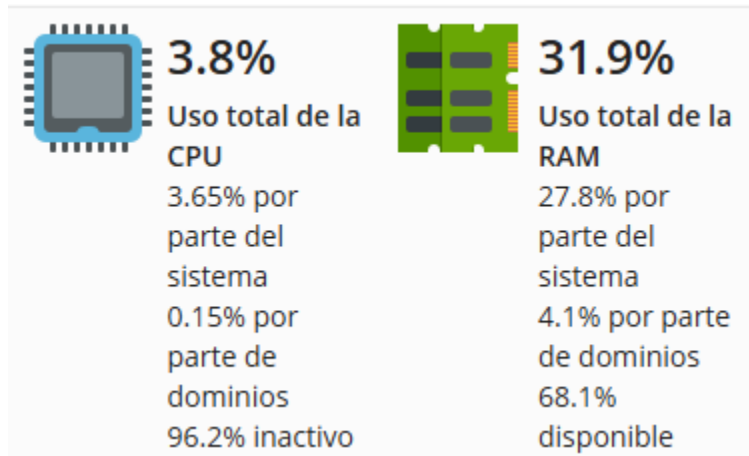
- Último minuto: 0.00%
- Últimos 5 minutos: 0.07%
- Últimos 15 minutos: 0.14%

Estos porcentajes reflejan un uso extremadamente bajo de la CPU, lo que sugiere que la CPU está manejando las tareas asignadas sin dificultades significativas, manteniendo una eficiencia alta bajo condiciones operativas normales.

4.2.4.2. Uso de Memoria

En cuanto al uso de memoria, el sistema tiene un total de 3.79 GB disponibles, de los cuales el 35.20% está actualmente en uso. Este nivel de uso de memoria indica una buena gestión de los recursos, permitiendo suficiente margen para el manejo de aplicaciones sin recurrir a la paginación excesiva, lo cual podría ralentizar el sistema.

La combinación de bajo uso de CPU y uso moderado de memoria demuestra la capacidad del sistema para manejar múltiples tareas simultáneamente sin comprometer el rendimiento. Estas métricas no solo aseguran que el sistema es capaz de mantener un alto rendimiento bajo diversas cargas operativas, sino que también subrayan la eficacia de la configuración actual del servidor para maximizar la eficiencia de los recursos disponibles.

Figura 42.*Uso de CPU y RAM del Servidor*

4.2.5. **Mantenibilidad**

La mantenibilidad mide la facilidad con la que se pueden realizar modificaciones y actualizaciones en el sistema. Esta característica es esencial para asegurar la evolución y mejora del sistema a lo largo del tiempo. Una métrica clave en este aspecto es la Densidad de Errores, que permite evaluar la calidad del código fuente. Se define como:

$$\text{Densidad de Errores} = \frac{\text{Número de Errores Detectados}}{\text{Líneas de Código Fuente}} \times 1000 = \frac{7}{6835} \times 1000 \quad (13)$$

$$= 1.02$$

$$\text{Mantenibilidad\%} = 100 - \frac{\text{Densidad de Errores}}{\text{Densidad de Errores Permitida}} \times 100 \quad (14)$$

$$= 100 - \frac{1.02}{14} \times 100 = 92.71\%$$

El sistema, con un total de 6,835 líneas de código y 7 errores detectados, la Densidad de Errores resulta en aproximadamente 1.02 errores por cada 1,000 líneas de código. Esta métrica proporciona una idea clara sobre la estabilidad y mantenibilidad del sistema, indicando

que la calidad del código es relativamente alta y que las áreas que requieren atención para refactorización o mejora son mínimas.

La utilización de un sistema de control de versiones con Git facilita la gestión de cambios y la colaboración, elementos cruciales para mantener y mejorar continuamente el código y la arquitectura del sistema. Esto no solo ayuda a mantener el sistema actualizable y adaptable a nuevas necesidades, sino que también asegura que las mejoras se puedan implementar de manera eficiente y efectiva.

4.2.6. Portabilidad

La portabilidad mide la capacidad del sistema para ejecutarse en diferentes entornos sin requerir modificaciones significativas. Para DAIFO TELECOM S.R.L., la portabilidad es esencial, ya que el sistema debe ser accesible en múltiples plataformas, incluyendo Android, Windows y navegadores web. Esta capacidad permite a los usuarios acceder al sistema desde dispositivos móviles, equipos de escritorio y otros dispositivos con acceso a internet, brindando flexibilidad y accesibilidad.

La métrica de Compatibilidad de Plataforma se calcula como:

$$\begin{aligned} \text{Compatibilidad de Plataforma} &= \frac{\text{Número de Plataformas Compatibles}}{\text{Número Total de Plataformas Objetivo}} \times 100 \quad (15) \\ &= \frac{3}{3} \times 100 = 100\% \end{aligned}$$

Actualmente, el sistema es compatible con las tres plataformas objetivo: Android, una aplicación para Windows y navegadores web. Esto da como resultado una Compatibilidad de Plataforma del 100%, lo cual asegura que el sistema cumple con los requisitos de accesibilidad y operatividad en diversos entornos.

Este nivel de portabilidad proporciona al sistema una alta adaptabilidad, permitiendo su uso en dispositivos y sistemas operativos variados, lo cual mejora la experiencia del usuario y maximiza el alcance de la solución.

4.2.7. Conclusión de las Métricas de Calidad

Las métricas de calidad implementadas para el sistema desarrollado para DAIFO TELECOM S.R.L. proporcionan un análisis exhaustivo y cuantitativo del desempeño del software, destacando fortalezas clave y áreas de oportunidad. Los resultados obtenidos reflejan un alto cumplimiento de los estándares establecidos, consolidando al sistema como una herramienta robusta y eficiente para las operaciones de la empresa.

- **Funcionalidad:** Se obtuvo una cobertura del 100% de los requisitos, lo que demuestra que el sistema cumple plenamente con las necesidades específicas del negocio.
- **Confiabilidad:** Con un 96.3% de tiempo operativo, el sistema asegura una disponibilidad constante y minimiza interrupciones en las operaciones críticas.
- **Usabilidad:** Un 99.75% refleja una experiencia de usuario altamente intuitiva y eficiente, permitiendo que los usuarios interactúen con el sistema de forma fluida.
- **Eficiencia:** Según el monitoreo, el uso de la CPU es del 3.8% y el uso de la RAM es del 31.9%, lo que indica un manejo óptimo de recursos, dejando un amplio margen para futuras expansiones o cargas adicionales.
- **Mantenibilidad:** Con un 92.71%, el sistema se caracteriza por una estructura de código clara y fácil de mantener, lo que facilita futuras actualizaciones y correcciones.
- **Portabilidad:** El 100% de compatibilidad asegura que el sistema puede operar sin problemas en todas las plataformas objetivo, garantizando su adaptabilidad en diferentes entornos.

La implementación de estas métricas no solo valida el desempeño actual del sistema, sino que también permite un monitoreo continuo y una mejora iterativa. Esto asegura que el sistema

puede adaptarse a los cambios en los requisitos del negocio y en las expectativas de los usuarios, promoviendo un enfoque ágil y sostenible para su desarrollo. En conjunto, los resultados obtenidos confirman que el sistema cumple con altos estándares de calidad, posicionándolo como una solución estratégica y confiable para las operaciones de DAIFO TELECOM S.R.L

4.3. Costos

En la gestión de proyectos de software, el control de costos es fundamental para asegurar una asignación eficiente de recursos y una correcta administración del presupuesto a lo largo del ciclo de vida del desarrollo. En el caso del sistema desarrollado para DAIFO TELECOM S.R.L., se ha utilizado el Modelo de Análisis de Puntos de Función (FPA) como la técnica principal para estimar los costos del proyecto. Este enfoque permite medir el tamaño del software de manera objetiva y precisa, cuantificando las funcionalidades proporcionadas al usuario final.

4.3.1. Identificación de Funcionalidades

La primera etapa en la estimación de costos utilizando el modelo FPA consiste en la identificación detallada de las funcionalidades requeridas por el sistema. En el sistema desarrollado para DAIFO TELECOM S.R.L., se han identificado y clasificado cinco tipos de funciones fundamentales:

- Entradas: Datos capturados a través de formularios u otros mecanismos, como el registro de órdenes de trabajo, datos de clientes y solicitudes de asistencia. Total de Entradas: 42.
- Salidas: Información generada por el sistema, como reportes de asistencias, resultados de site surveys y comprobantes de pago. Total de Salidas: 12.

- Consultas: Funciones que permiten a los usuarios realizar búsquedas específicas, como consultas de datos de clientes, servicios y reportes históricos. Total de Consultas: 42.
- Archivos Internos: Estructuras de almacenamiento de datos persistentes, incluyendo bases de datos de clientes, órdenes de trabajo y registros de asistencias. Total de Archivos Internos: 8.
- Interfaces Externas: Comunicación con sistemas externos, como notificaciones automáticas mediante WhatsApp Web JS y la conexión de monitoreo de sensores mediante ESP32. Total de Interfaces Externas: 2.

Puntos de Función

(16)

$$= \sum(\text{Entradas} + \text{Salidas} + \text{Consultas} + \text{Archivos Internos} + \text{Interfaces Externas}) = 42 + 12 + 42 + 8 + 2 = 106$$

Sumando estas funcionalidades, el sistema alcanza un total de 106 puntos de función, lo cual proporciona una medida objetiva del tamaño del software y facilita la estimación de los recursos necesarios para el desarrollo.

La identificación precisa de estas funcionalidades permite establecer un tamaño del software basado en criterios objetivos, facilitando una evaluación inicial de los recursos necesarios para su desarrollo.

4.3.2. Asignación de Complejidad

Una vez identificadas las funcionalidades del sistema, el siguiente paso en el modelo FPA es asignar un nivel de complejidad a cada función. La complejidad de cada función puede clasificarse como baja, media o alta, y esta clasificación se basa en las características específicas de cada operación en el sistema:

- Complejidad Baja: Incluye funciones que realizan operaciones básicas de captura o presentación de datos, como el registro de información estándar de clientes y asistencias, y consultas simples de datos. Este nivel representa aproximadamente 45 funciones en el sistema.
- Complejidad Media: Comprende funciones que requieren validaciones de datos o interacciones más complejas, como la gestión de permisos y roles y la autenticación de usuarios. Estas operaciones involucran una mayor lógica de negocio y representan aproximadamente 35 funciones en el sistema.
- Complejidad Alta: Corresponde a funciones avanzadas con operaciones complejas, como el cálculo automático de reportes financieros, generación de comprobantes QR, y la integración de sensores de monitoreo en tiempo real mediante ESP32. Estas funciones representan aproximadamente 20 operaciones en el sistema.

Cada nivel de complejidad influye en la ponderación de los puntos de función y, por lo tanto, en la estimación final del esfuerzo y los costos del proyecto. La asignación de complejidad es esencial para obtener una estimación realista, evitando tanto subestimaciones como sobrestimaciones, y permitiendo una planificación precisa de los recursos necesarios para el desarrollo y mantenimiento del sistema.

4.3.3. Cálculo de Esfuerzo

La productividad representa el tiempo estimado en horas-hombre que se necesita para desarrollar cada punto de función. Para este proyecto, se ha estimado que la productividad es de 3 horas-hombre por punto de función, teniendo en cuenta factores como la complejidad del sistema, el nivel de validación de datos y la experiencia del equipo.

Con un total de 106 puntos de función identificados en el sistema, el cálculo del esfuerzo es el siguiente:

$$\text{Esfuerzo (HH)} = 106 \times 3 = 318 \text{ horas/hombre} \quad (17)$$

Esta estimación de esfuerzo proporciona una medida realista del tiempo necesario para completar el proyecto. Cada nivel de complejidad asignado a las funcionalidades influye en esta estimación, ayudando a evitar subestimaciones o sobreestimaciones que podrían afectar la asignación de recursos y la planificación del proyecto.

Una vez calculado el esfuerzo, se determina el costo total multiplicando las horas-hombre por la tarifa por hora establecida para el equipo de desarrollo en Bolivia, que es de 70 Bs. por hora. Por lo tanto, el costo total estimado para el desarrollo es:

$$\text{Costo Total} = 318 \times 70 \text{ Bs} = 22,260 \text{ Bs} \quad (18)$$

4.3.4. Conclusiones sobre los Costos

La aplicación del Modelo de Análisis de Puntos de Función (FPA) ha demostrado ser una herramienta eficaz para estimar y gestionar los costos del sistema desarrollado para DAIFO TELECOM S.R.L. Al proporcionar una medición objetiva del tamaño del software y relacionarla con la productividad del equipo de desarrollo, este enfoque permite una planificación financiera precisa y un control riguroso del presupuesto, alcanzando un costo total estimado de 22,260 Bs.

Este cálculo no solo facilita una proyección clara de los recursos necesarios para el desarrollo, sino que también permite realizar comparaciones de costos entre los diferentes módulos del sistema. Esta capacidad de desglosar costos es especialmente útil para identificar módulos que, por su complejidad o por la integración de tecnologías avanzadas, requieren una mayor asignación de recursos. Así, el FPA se convierte en una herramienta estratégica que ayuda a los gestores del proyecto a tomar decisiones informadas sobre la asignación de

presupuesto, asegurando que los módulos críticos tengan el soporte adecuado y que los recursos se utilicen de manera óptima.

En conclusión, la implementación de un enfoque basado en puntos de función proporciona un marco sólido para la gestión de costos en el desarrollo de software. En proyectos como el de DAIFO TELECOM S.R.L., donde la complejidad y la variedad de funcionalidades son significativas, esta metodología es especialmente útil para mantener el control financiero y asegurar el éxito del proyecto a lo largo de su ciclo de vida, permitiendo una evolución planificada y eficiente del sistema.

4.4. Seguridad

La seguridad es un pilar fundamental en el desarrollo de sistemas de información, especialmente en sectores sensibles como el de las telecomunicaciones, donde se manejan datos críticos y confidenciales. Para el sistema de DAIFO TELECOM S.R.L., se han implementado prácticas de seguridad alineadas con los estándares de la norma ISO/IEC 27001, enfocadas en la protección de la confidencialidad, integridad y disponibilidad de la información.

4.4.1.1. Autenticación y Contraseñas

Se han implementado políticas de seguridad para contraseñas, que incluyen una longitud mínima, el uso de caracteres especiales y la renovación periódica. Las contraseñas están cifradas utilizando el algoritmo bcrypt, lo cual cumple con los estándares de seguridad recomendados para evitar accesos no autorizados y proteger contra ataques de fuerza bruta.

4.4.1.2. Encriptación y Tokens

Para proteger la información en tránsito y en reposo, se utilizan técnicas de encriptación simétrica y asimétrica, garantizando que las comunicaciones entre el cliente y el servidor estén aseguradas. Los JSON Web Tokens (JWT) gestionan las sesiones de los usuarios,

almacenando los tokens de manera segura en el navegador mediante localStorage, previniendo accesos no autorizados y validando la identidad de los usuarios durante las sesiones.

4.4.1.3. Captura de Firmas

Para la validación de aprobaciones y conformidades, se utiliza la captura de firmas electrónicas mediante VueSignature, una herramienta que genera una imagen con marca de tiempo. Este procedimiento asegura la autenticidad y rastreabilidad de las firmas electrónicas, que se almacenan de manera segura y solo se utilizan en el contexto aprobado.

4.4.1.4. Validación de Peticiones

El sistema emplea controles de CORS (Cross-Origin Resource Sharing) para restringir el acceso a solicitudes provenientes de orígenes no confiables. Adicionalmente, la verificación mediante tokens JWT garantiza que solo los usuarios autorizados puedan realizar operaciones críticas dentro del sistema, alineándose con los controles de acceso recomendados por ISO/IEC 27001.

4.4.1.5. Copias de Seguridad y Versionado

Para asegurar la disponibilidad de la información y la capacidad de recuperación ante fallos, se realizan copias de seguridad incrementales en un servidor dedicado y seguro. Los datos están cifrados para evitar accesos no autorizados, y se guardan versiones anteriores en un directorio del servidor, permitiendo restauraciones de manera rápida y segura.

4.4.1.6. Firma de Aplicaciones

Los archivos de la aplicación, como los APK de Android, están firmados digitalmente para asegurar la autenticidad del software y proteger a los usuarios de posibles modificaciones maliciosas. La firma se realiza con una clave privada en el entorno de desarrollo, garantizando la integridad del software distribuido.

4.4.1.7. Almacenamiento de Archivos con UUID

Para proteger la integridad de los archivos y evitar accesos directos no autorizados, se utilizan UUIDs como nombres de archivo. Esto minimiza el riesgo de colisiones y accesos directos a los archivos, reforzando la seguridad de documentos generados por el sistema, como reportes y comprobantes.

4.4.1.8. Conclusiones sobre Seguridad

En conclusión, el enfoque de seguridad implementado en el sistema de DAIFO TELECOM S.R.L. cumple con los lineamientos de ISO/IEC 27001, asegurando que las prácticas adoptadas brinden una sólida protección en los ámbitos de autenticación, cifrado, almacenamiento seguro y gestión de usuarios.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



5. CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

La implementación del sistema multiplataforma de gestión de operaciones en el sector de telecomunicaciones para DAIFO TELECOM S.R.L. ha culminado satisfactoriamente, logrando cumplir con los objetivos propuestos y respondiendo a las necesidades operativas y de gestión de la empresa. Durante el proceso de desarrollo, se implementó una metodología Iterativa e Incremental, lo que permitió llevar a cabo mejoras progresivas y adaptarse a los cambios en los requisitos y condiciones del entorno de trabajo. Cada iteración se enfocó en la creación de módulos funcionales y completos, optimizando las funcionalidades existentes y añadiendo nuevas capacidades al sistema.

Se destaca como uno de los logros más importantes la implementación de un sistema multiplataforma, que brinda acceso tanto desde dispositivos móviles como desde aplicaciones de escritorio, facilitando la labor de los técnicos y administrativos. Esta característica ha contribuido significativamente a mejorar la eficiencia operativa de la empresa, permitiendo la gestión centralizada de órdenes de trabajo, asistencias, monitoreo de infraestructura, y manejo de pagos.

El sistema de monitoreo en tiempo real utilizando sensores y microcontroladores ha demostrado ser una solución efectiva para la supervisión del estado de las torres de comunicación. Los datos recogidos por los sensores de temperatura, humedad, inclinación y vibraciones son enviados periódicamente al backend, donde se almacenan y se procesan para ser visualizados de forma dinámica en el frontend mediante WebSockets. Este enfoque ha permitido incrementar la capacidad de la empresa para anticiparse a problemas en la infraestructura, garantizando así una continuidad operativa de los servicios.

En términos de seguridad, el sistema ha adoptado medidas avanzadas como la encriptación de contraseñas mediante bcrypt, el uso de JSON Web Tokens (JWT) para la

autenticación de sesiones, y la validación de peticiones de origen para prevenir ataques. Además, se implementaron mecanismos para la captura de firmas electrónicas con marcas de tiempo, asegurando la validez de las conformidades de servicio.

Otro aspecto a destacar es la optimización del sistema para trabajar con notificaciones automáticas mediante WhatsApp, lo cual ha mejorado significativamente la comunicación con los clientes y técnicos. La posibilidad de configurar cuentas y enviar notificaciones automáticas ha permitido mantener a todos los usuarios informados sobre el estado de sus órdenes de trabajo, asistencias y pagos, lo que ha incrementado la satisfacción del cliente.

La finalización de cada iteración fue precedida de un proceso riguroso de pruebas que incluyó pruebas unitarias, pruebas de integración y pruebas de aceptación del usuario (UAT), lo que permitió validar la robustez y la usabilidad del sistema en cada fase del desarrollo. La retroalimentación constante de los usuarios y la evaluación del sistema en situaciones reales de operación jugaron un papel clave para refinar y ajustar las funcionalidades.

En resumen, el sistema desarrollado ha logrado:

- Optimizar la gestión de órdenes de trabajo, asistencias, pagos y monitoreo de infraestructura, reduciendo los tiempos de procesamiento y aumentando la precisión de la información.
- Facilitar la toma de decisiones mediante reportes en tiempo real y notificaciones automáticas.
- Mejorar la seguridad y la integridad de los datos a través de técnicas de encriptación y validación de peticiones.
- Incrementar la satisfacción del cliente al proporcionar información oportuna y confiable sobre sus servicios.

Este proyecto marca un avance significativo en la infraestructura tecnológica de DAIFO TELECOM S.R.L., brindando una base sólida para futuras ampliaciones y mejoras del sistema.

En conclusión, el sistema desarrollado para DAIFO TELECOM S.R.L. ha logrado optimizar de manera significativa los procesos internos y la gestión operativa de la empresa. Con un enfoque en la calidad, seguridad y eficiencia, el proyecto ha sentado las bases para el crecimiento continuo y la adaptación a las demandas cambiantes del sector de telecomunicaciones. Siguiendo las recomendaciones planteadas, DAIFO TELECOM S.R.L. podrá mantener y mejorar este sistema, asegurando una operación fluida y una gestión eficiente de sus servicios.

5.2. Recomendaciones

Para asegurar la operación continua y el crecimiento del sistema, se recomiendan las siguientes acciones y consideraciones:

- **Revisión periódica del almacenamiento disponible:** Dado que el sistema maneja una cantidad significativa de datos, incluyendo imágenes, firmas electrónicas, registros de sensores y archivos de órdenes de trabajo, es crucial monitorizar el espacio de almacenamiento en el servidor. Se sugiere realizar copias de seguridad incrementales y considerar la implementación de estrategias de compresión y limpieza de datos antiguos para optimizar el uso del espacio.
- **Mantenimiento de las conexiones de los sensores y microcontroladores:** La infraestructura de monitoreo en tiempo real depende de la correcta operación de los microcontroladores ESP32 y los sensores. Se recomienda verificar regularmente las conexiones físicas y revisar el firmware de los dispositivos para garantizar la continuidad del monitoreo y la precisión de los datos recolectados.
- **Actualización de librerías y dependencias:** El sistema se ha desarrollado utilizando diversas tecnologías como Vue.js, Node.js, Apache Cordova y Electron. Dado que

estas herramientas se actualizan de manera constante, es fundamental mantener las librerías y dependencias actualizadas para garantizar la seguridad y la compatibilidad del sistema con nuevas versiones.

- Optimización del rendimiento del backend: A medida que se incrementa el número de usuarios y dispositivos conectados, es importante evaluar y optimizar la eficiencia del backend. Se recomienda implementar pruebas de carga periódicas para identificar cuellos de botella y optimizar el uso de recursos en el servidor.
- Monitoreo continuo de la seguridad del sistema: La seguridad es un aspecto crítico, especialmente considerando que el sistema maneja datos sensibles de clientes y operaciones. Se recomienda implementar auditorías de seguridad periódicas y evaluar nuevas medidas de protección para mantener la confidencialidad, integridad y disponibilidad de la información.
- Capacitación y soporte continuo a los usuarios: La introducción de nuevas funcionalidades y mejoras en el sistema requiere una capacitación constante para los usuarios finales. Se sugiere desarrollar un plan de formación que incluya tutoriales en línea, manuales y sesiones de entrenamiento para asegurar un uso óptimo de las herramientas disponibles.
- Mejoras en la experiencia de usuario: Aunque el sistema ha sido diseñado para ser intuitivo, es recomendable realizar encuestas de satisfacción y sesiones de retroalimentación regulares con los usuarios para identificar áreas de mejora en la interfaz y en la experiencia de usuario.
- Planificación para futuras expansiones del sistema: El éxito de este proyecto abre la puerta para futuras ampliaciones y mejoras en las capacidades del sistema. Se sugiere explorar nuevas funcionalidades, como la implementación de módulos adicionales para la gestión de inventarios o el control avanzado de infraestructura mediante técnicas de machine learning.

BIBLIOGRAFÍA

- Albrecht, A. J., y Gaffney, J. E. (1983). Software function, source lines of code, and development effort prediction: A software science validation. IEEE Transactions on Software Engineering.
- Apache Software Foundation. (2023). Apache Cordova. Recuperado de <https://cordova.apache.org/>
- Cabo Tinoso. (2023). Instrucciones para el sensor BME280 con Arduino: presión, temperatura, humedad. Recuperado de <https://www.cabotinoso.es/instrucciones-para-el-sensor-bme280-con-arduino-presion-temperatura-humedad/>
- Espressif Systems. (2023). Introducción al ESP32. Recuperado de <https://www.espressif.com/>
- Evan You. (2014). Vue.js. Recuperado de <https://vuejs.org/>
- Fernández, C., y Díaz, M. (2018). Mapas interactivos con Leaflet: Aplicaciones y casos de uso. Editorial Académica.
- Fernández, L., y Ramírez, C. (2020). Sistemas operativos y gestión de datos en aplicaciones empresariales. Ediciones Díaz de Santos.
- García, J., y Luna, R. (2019). Gestión de la información y sistemas integrados en telecomunicaciones. Editorial UOC.
- García, M., y Rodríguez, J. (2015). Seguridad de la información y protección de datos. Ediciones Díaz de Santos.
- García, M., y Rodríguez, J. (2019). Gestión de la información y metodologías ágiles en telecomunicaciones. Ediciones Díaz de Santos.
- GitHub. (2023). Electron: Build cross-platform desktop apps with JavaScript, HTML, and CSS. Recuperado de <https://www.electronjs.org/>
- Gómez, F., y Martínez, L. (2017). Firmas electrónicas y su validez jurídica. Editorial Jurídica.

- Gómez, M. (2021). Sistemas de información empresariales: Estrategias y aplicaciones. Ediciones Díaz de Santos.
- González, A., y Flores, J. (2020). Automatización y gestión de operaciones de campo en telecomunicaciones. Editorial Tecnológica.
- González, P., y Flores, J. (2020). Desarrollo de aplicaciones web escalables con Node.js y tecnologías de tiempo real. Editorial Tecnológica.
- Google. (2023). Puppeteer. Recuperado de <https://pptr.dev/>
- Heizer, J., Render, B., y Munson, C. (2021). Principios de administración de operaciones. Pearson.
- ISO/IEC. (2011). ISO/IEC 25010:2011 - Systems and software engineering. International Organization for Standardization.
- ISO/IEC. (2013). ISO/IEC 27001:2013 - Information security management. International Organization for Standardization.
- Laudon, K. C., y Laudon, J. P. (2021). Sistemas de información gerencial. Pearson.
- Leaflet. (2023). Leaflet. Recuperado de <https://leafletjs.com/>
- Llamas, L. (2020). Sensor ambiental con Arduino y BME280. Recuperado de <https://www.luisllamas.es/sensor-ambiental-arduino-bme280/>
- Lovell, A. (2023). Sharp: High performance image processing in Node.js. Recuperado de <https://sharp.pixelplumbing.com/>
- Martínez, S., y Sánchez, L. (2019). Gestión de operaciones en la industria de telecomunicaciones. Ediciones Díaz de Santos.
- Medina, J. (2010). Tecnología y productividad en las organizaciones. Revista de Ciencias Sociales, 10(1), 47-58.

- Monreal, C. (2018, julio 17). Ciclos de vida iterativo e incremental, ¿Qué son? Curso Dirección Proyectos. <https://www.cursodireccionproyectos.com/2018/07/ciclos-de-vida-iterativo-e-incremental-que-son/>
- MySQL AB. (2023). MySQL: Open source database. Recuperado de <https://www.mysql.com/>
- Naylamp Mechatronics. (2018). Tutorial MPU6050: Acelerómetro y Giroscopio. Recuperado de https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html
- Nielsen, J. (2012). Usabilidad: Diseño de sitios web con el usuario en mente. McGraw-Hill.
- Nodemailer Team. (2023). Nodemailer. Recuperado de <https://nodemailer.com/>
- Núñez, S. (2020). Comunicación en tiempo real con WebSockets: Aplicaciones en la industria de telecomunicaciones. Editorial UOC.
- OpenJS Foundation. (2023). Node.js. Recuperado de <https://nodejs.org/>
- Patterson, D. A., Hennessy, J. L., y Arpaci-Dusseau, A. C. (2018). Computer organization and design MIPS edition: The hardware/software interface. Elsevier.
- Pérez, L., y García, M. (2021). Automatización de procesos en la atención al cliente. Ediciones Académicas.
- Pressman, R. S. (2010). Ingeniería del software: Un enfoque práctico (7ma ed.). McGraw-Hill.
- Rodríguez, A., y Luna, L. (2018). Metodologías de desarrollo de software: Enfoques ágiles e incrementales. Editorial UOC.
- Rodríguez, L., y Pérez, M. (2018). Desarrollo de aplicaciones móviles con Android: Un enfoque práctico. Editorial Académica.
- Rodríguez, M., y Ramírez, E. (2019). Vue.js: El framework progresivo para interfaces web modernas. Ediciones Tecnológicas.

- Rojas, C., y Pérez, S. (2018). Sistemas multiplataforma y accesibilidad en el desarrollo de software. Editorial Universitaria.
- Rojas, S., y Ramírez, A. (2019). Aplicaciones de escritorio y multiplataforma: Desarrollo con Electron. Ediciones Díaz de Santos.
- Sommerville, I. (2011). Ingeniería de software (9na ed.). Addison-Wesley.
- Soto, C., y Villalobos, A. (2021). Sequelize ORM: Simplificación de la gestión de bases de datos relacionales en Node.js. Ediciones Díaz de Santos.
- SpainLabs. (2017). Guía ESP8266 (NodeMCU) + sensor BME280 + MQTT. Recuperado de <https://www.spainlabs.com/foros/showthread.php?tid=6346>
- Tanenbaum, A. S., y Van Steen, M. (2017). Distributed systems: Principles and paradigms. Pearson.
- Villalobos, L., y López, M. (2014). Gestión y estimación de proyectos de software mediante puntos de función. Ediciones Académicas.

ANEXOS

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO

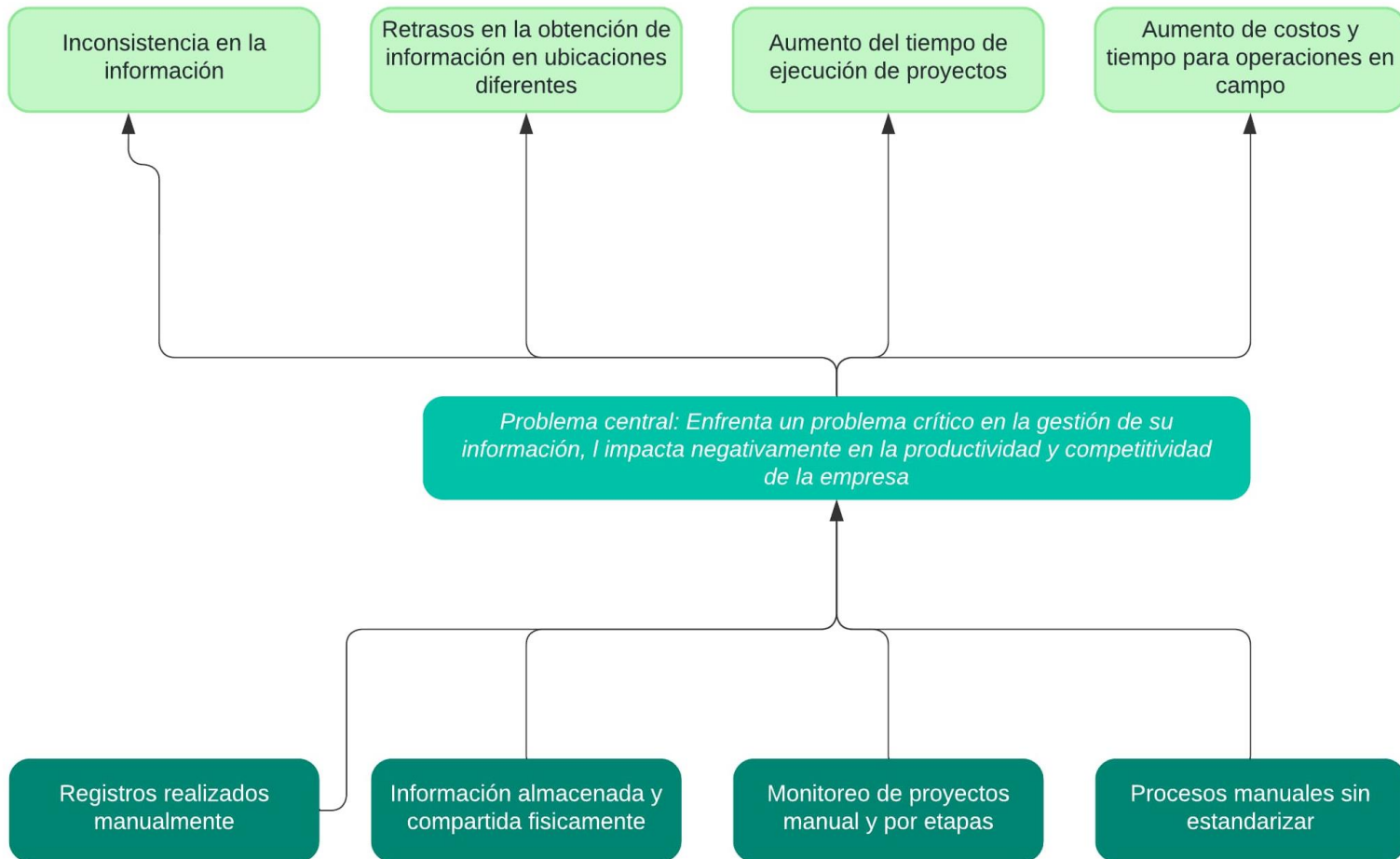


ANEXO A

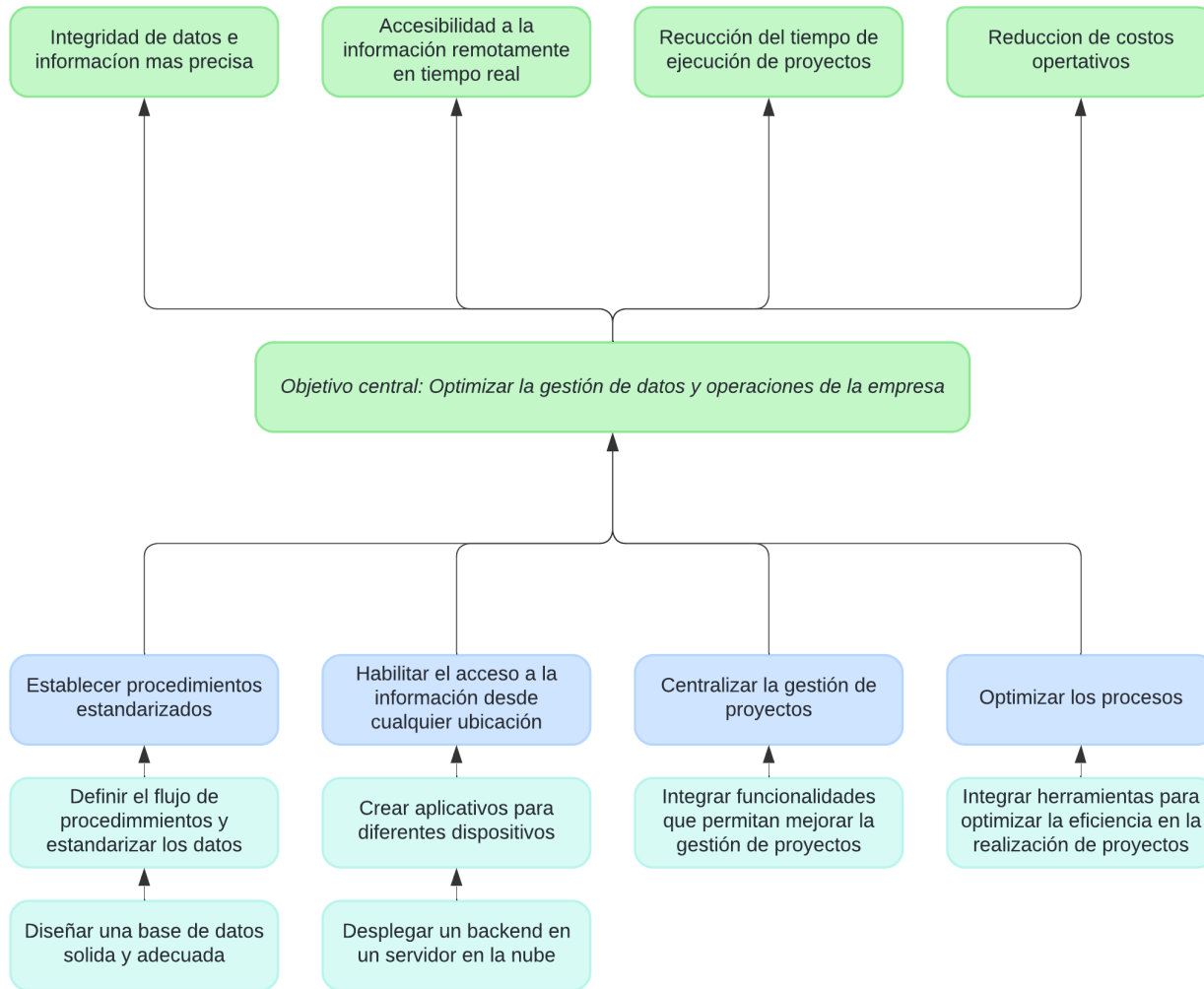
**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



ÁRBOL DE PROBLEMAS



ÁRBOL DE OBJETIVOS

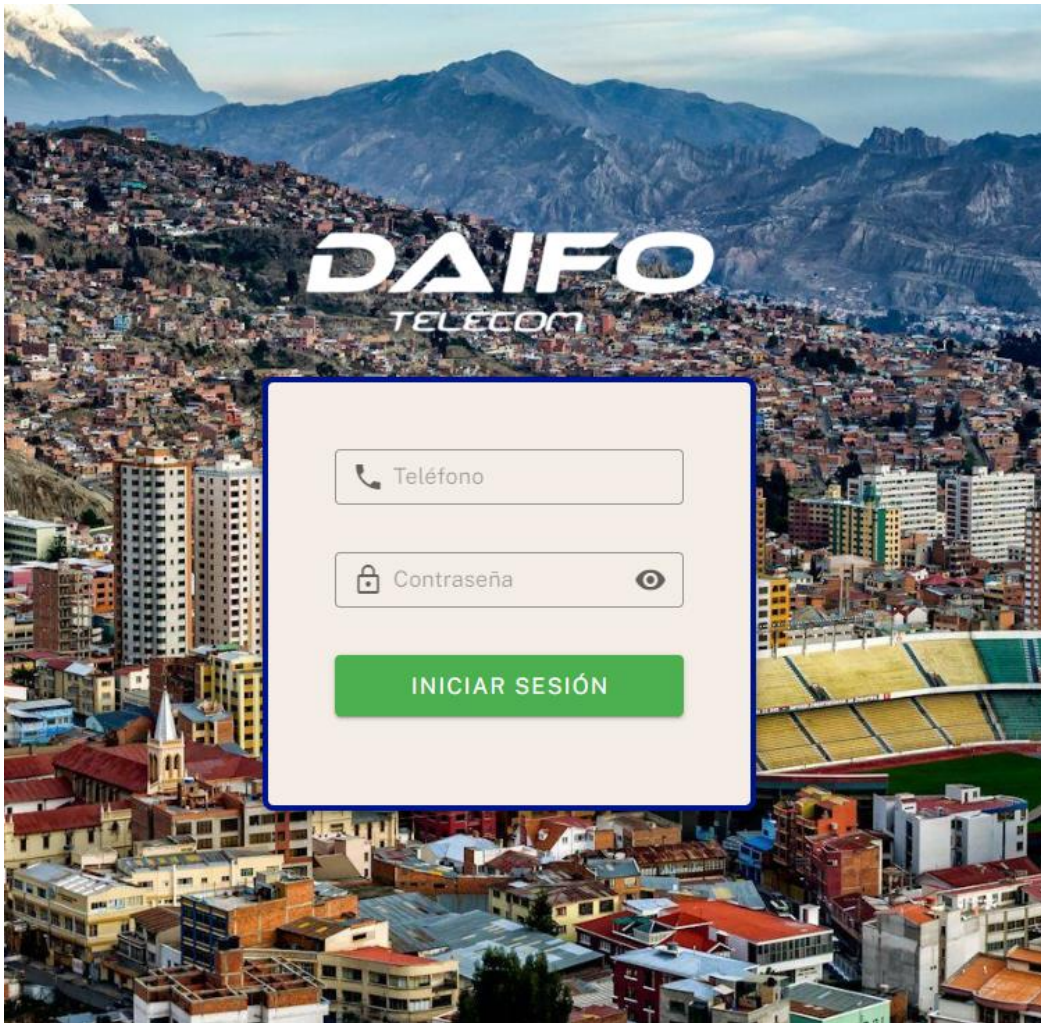


ANEXO B

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



MANUAL OPERATIVO V1



Usuarios

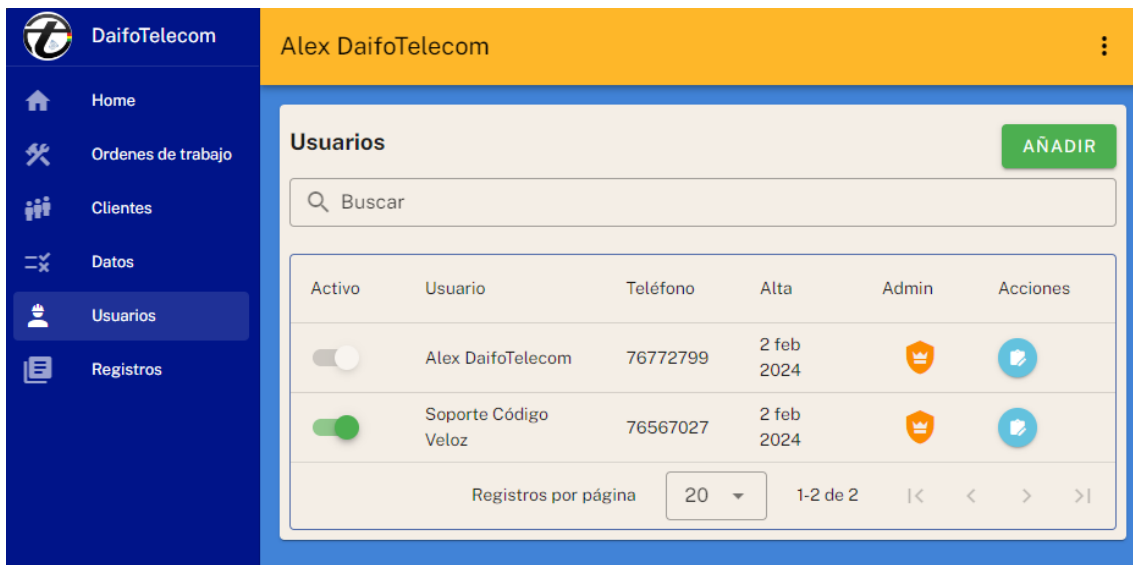
Descripción: Los usuarios serán el personal operativo y administrativo de Daifo Telecom que requiera acceder a las Ordenes de trabajo o a los datos de los clientes.

Creación: Los administradores pueden crear cuentas de usuarios para el personal (se usa el teléfono y contraseña para iniciar sesión)

Modificación: Los administradores pueden modificar a los usuarios y también pueden restablecer la contraseña de los usuarios

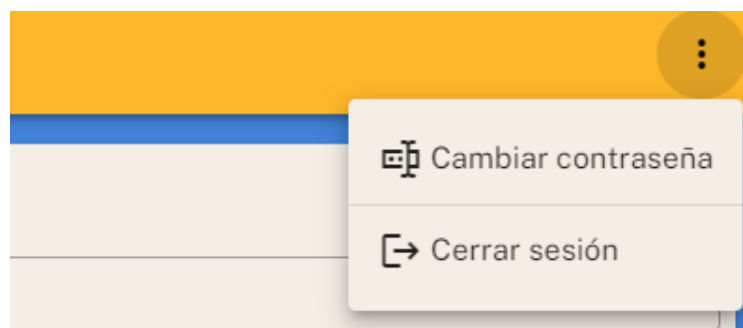
Restricción: Los administradores pueden “desactivar” a los usuarios, con esta acción el usuario inactivo será deslogueado y no podrá ingresar al sistema

Rol: Los administradores pueden definir a usuarios como administradores del sistema



Cambio de contraseña

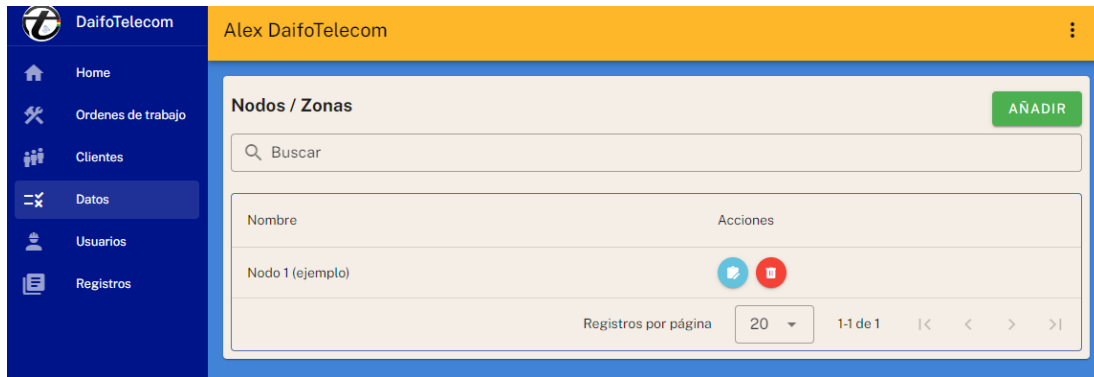
Descripción: Cualquier usuario logueado puede cambiar su contraseña actual a una



nueva (se recomienda cambiar la contraseña la primera vez que inicia sesión).

Datos

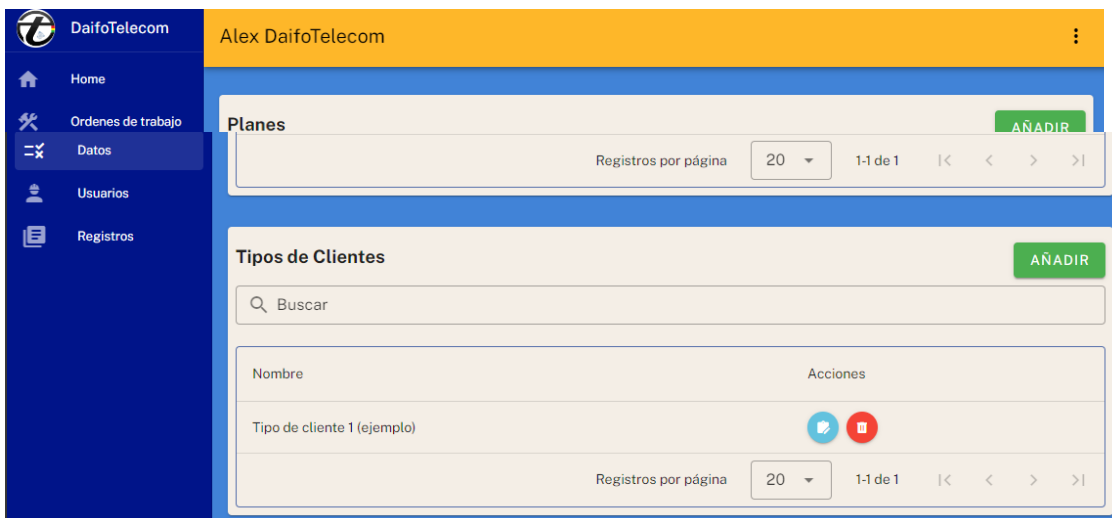
Nodos / Zonas: Son registros que serán usados para la creación y edición de clientes, considerando que cada cliente pertenece a un nodo o zona. Los administradores pueden:



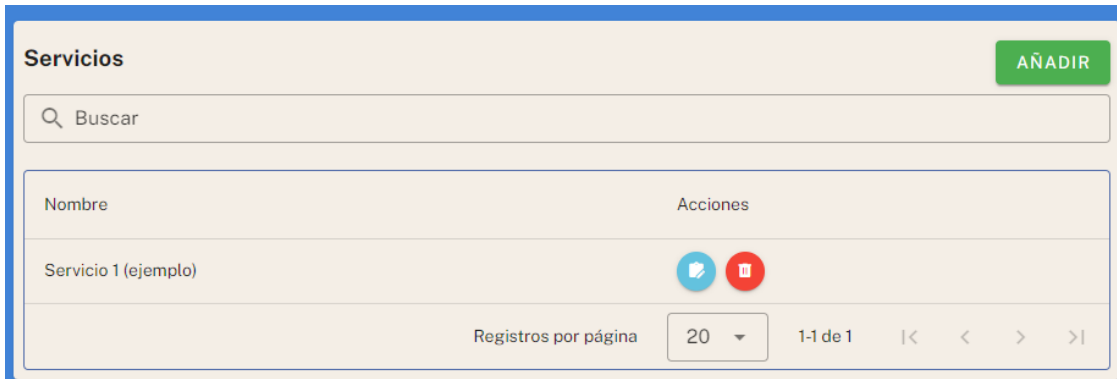
añadir, editar y eliminar estos registros.

Planes: Son registros que serán usados para la creación y edición de clientes, considerando que cada cliente puede o no tener un plan, ejemplo (Plan económico, Plan estudiante, etc) Los administradores pueden: añadir, editar y eliminar estos registros.

Tipos de clientes: Son registros que serán usados para la creación y edición de clientes, considerando que cada cliente pertenece a un tipo, ejemplo (Portador, Internet, etc.). Los administradores pueden: añadir, editar y eliminar estos registros.



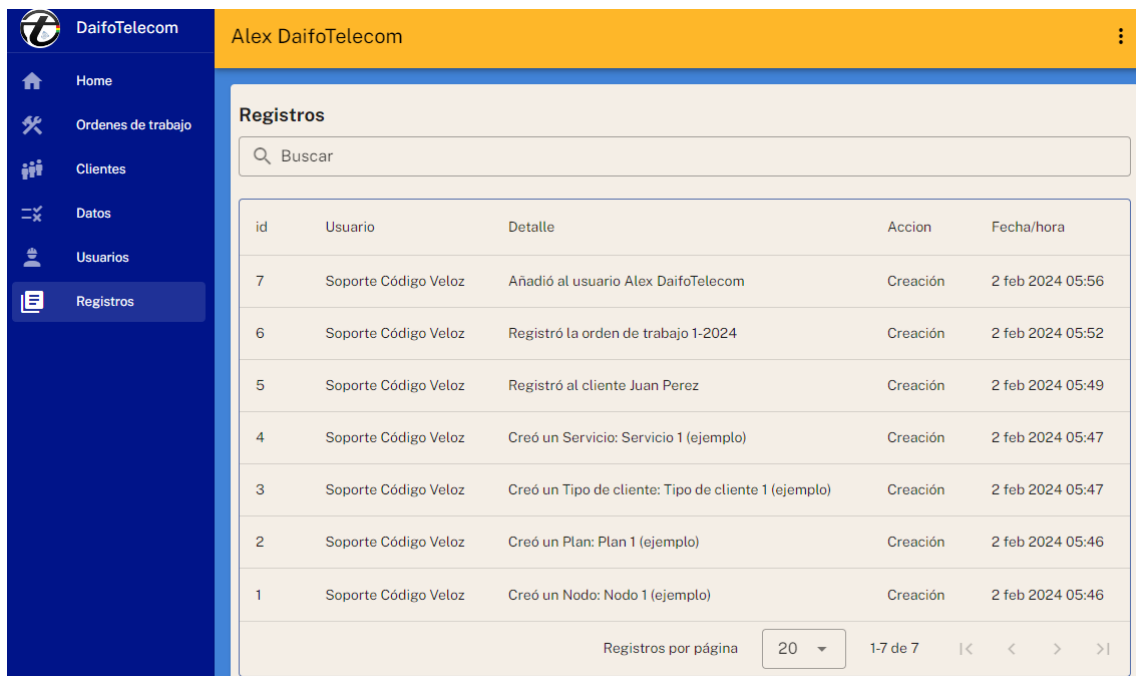
Servicios: Son registros que serán usados para la creación de ordenes de trabajo, considerando que cada orden de trabajo puede tener varios servicios, ejemplo



(Mantenimiento de red, Instalación de fibra óptica Internet, etc.). Los administradores pueden: añadir, editar y eliminar estos registros.

Registros




Descripción: Con cada creación, modificación y eliminación de cualquier dato del sistema se guarda un registro detallado con fecha y hora de la actividad realizada, (su uso es para fines de control de datos, solo los administradores pueden ver el registro).



Cientes

Descripción: Registro de clientes

The screenshot shows the 'Clientes' page in the DaifoTelecom system. The interface includes a sidebar with navigation options: Home, Ordenes de trabajo, Cientes (selected), Datos, Usuarios, and Registros. The main content area has a yellow header with the user name 'Alex DaifoTelecom' and a green 'AÑADIR CLIENTE' button. Below the header is a search bar labeled 'Buscar'. A table displays client information with columns: Cliente, Nodo, Tipo, Alta, and Acciones. The table contains one entry for 'Juan Perez' with details: 'Nodo 1 (ejemplo)', 'Tipo de cliente 1 (ejemplo)', and '2 feb 2024'. The 'Acciones' column for this entry contains three icons: a location pin, a camera, and a document. At the bottom of the table, there is a pagination control showing 'Registros por página' set to 20, '1-1 de 1', and navigation arrows.



Cliente	Nodo	Tipo	Alta	Acciones
Juan Perez	Nodo 1 (ejemplo)	Tipo de cliente 1 (ejemplo)	2 feb 2024	  

Los datos de los clientes incluyen fotocopia de carnet, factura de servicios básicos, croquis, ubicación (coordenadas geográficas)

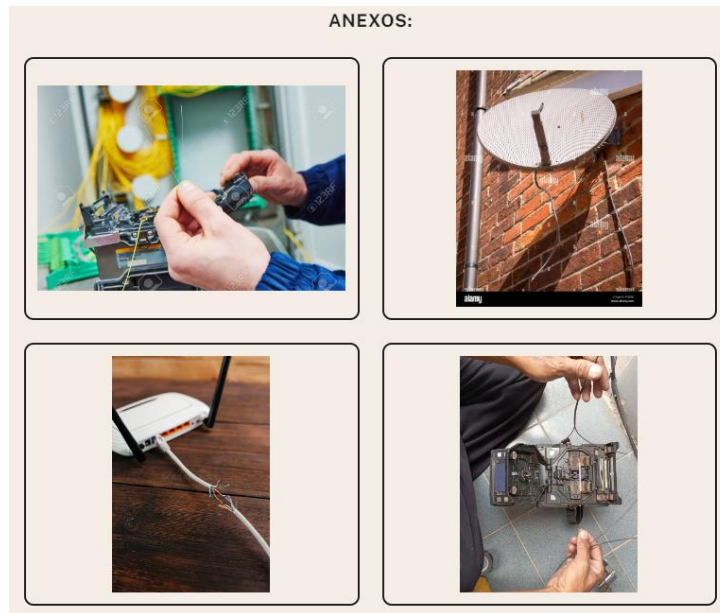
Orden de Trabajo

Descripción: Registro de ordenes de trabajos

The screenshot shows the 'Ordenes de Trabajo' page in the DaifoTelecom system. The interface includes a sidebar with navigation options: Home, Ordenes de trabajo (selected), Cientes, Datos, Usuarios, and Registros. The main content area has a yellow header with the user name 'Alex DaifoTelecom' and a green 'AÑADIR' button. Below the header is a search bar labeled 'Buscar'. A table displays work order information with columns: Nro, Tipo, Cliente, Técnico, Alta, and Acciones. The table contains one entry with details: '1/2024', 'Tipo de cliente 1 (ejemplo)', 'Juan Perez', 'Soporte Código Veloz', and '2 feb 2024'. The 'Acciones' column for this entry contains two icons: a camera and a location pin. At the bottom of the table, there is a pagination control showing 'Registros por página' set to 20, '1-1 de 1', and navigation arrows.

Nro	Tipo	Cliente	Técnico	Alta	Acciones
1/2024	Tipo de cliente 1 (ejemplo)	Juan Perez	Soporte Código Veloz	2 feb 2024	 


Los datos de las ordenes de trabajo incluyen hasta un máximo de 4 imágenes para anexas





Detalles del cliente

Nombres: Juan
Apellidos: Perez
Nodo: Nodo 1 (ejemplo)
Tipo de cliente: Tipo de cliente 1 (ejemplo)

Plan: Plan 1 (ejemplo)
Correo: juan@gmail.com
Teléfono:
Ubicación: [Ver ubicación en Google](#)

Fotocopia de carnet:


Factura de luz o agua:


Croquis:


CERRAR

Agregar OT



Tipo*

Personal de contacto

Cliente*

Técnico Asignado

Alex DaifoTelecom

Categoría de Servicio*

Tipo de conexión

Dirección origen

Dirección destino

Observaciones

Diagnóstico

Descripción del trabajo realizado*

Sugerencias

Equipos y accesorios instalados

Hora de inicio

12:00



Hora de finalización

12:00



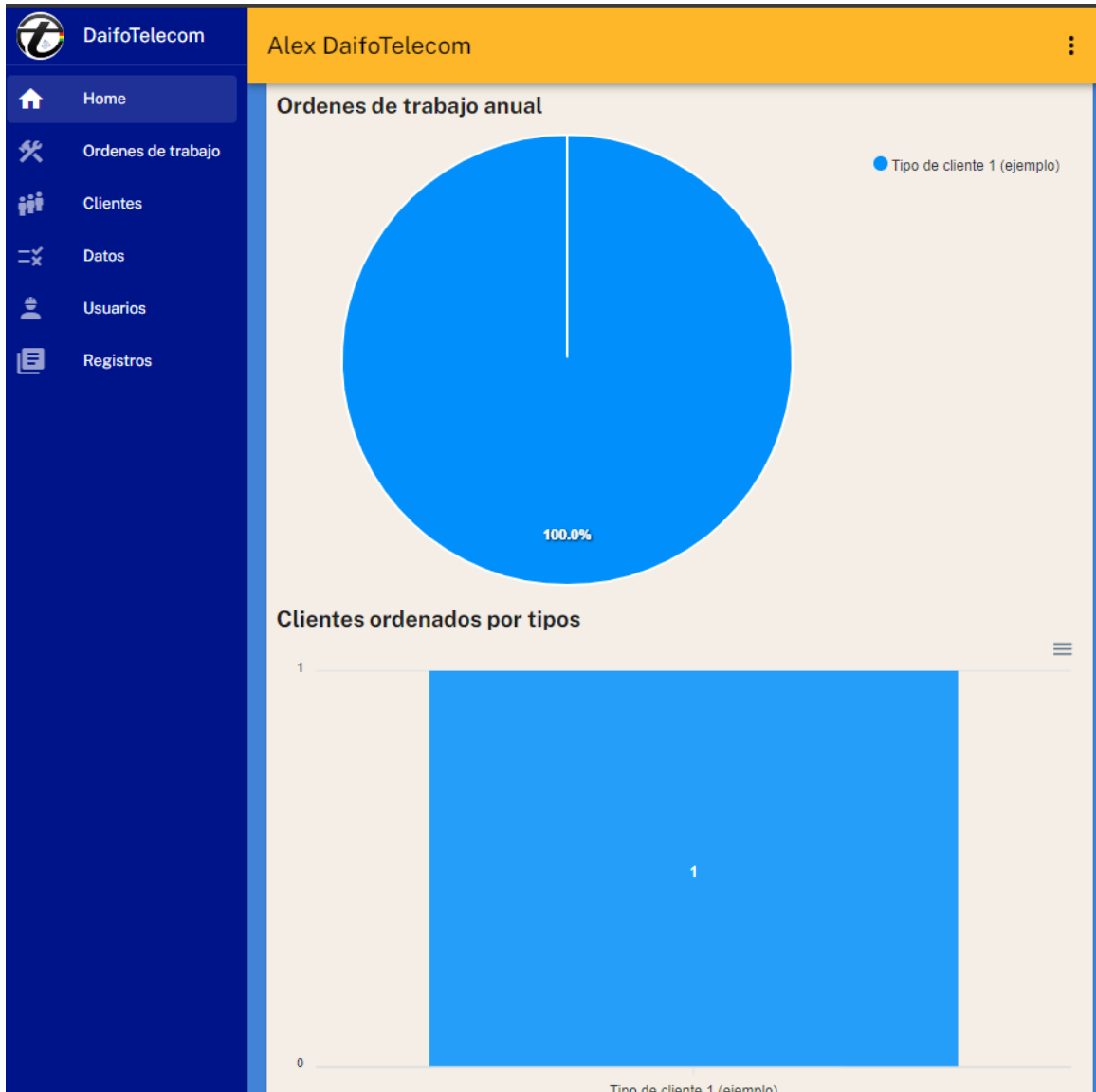
Observaciones

Imágenes



Gráficos

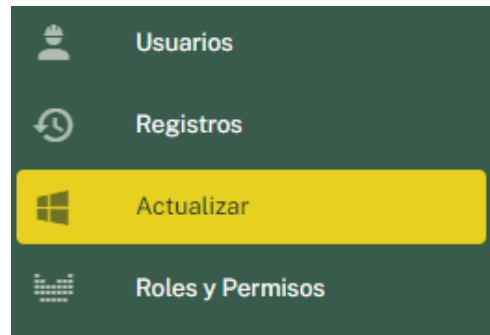
Descripción: El primer gráfico muestra las ordenes de trabajo total anula agrupándolos por tipo de cliente, el segundo gráfico muestra el total de clientes registrados agrupados por tipo



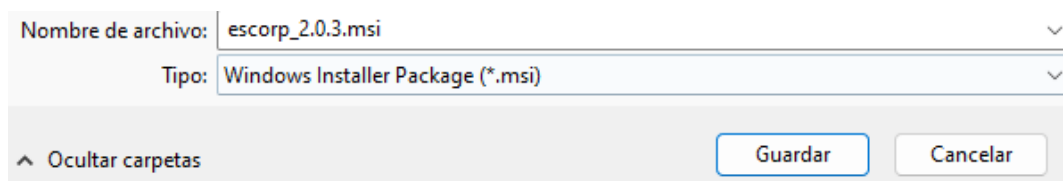
MANUAL OPERATIVO V2

ACTUALIZACIÓN VERSIÓN DESKTOP (WINDOWS)

Si existe una versión nueva se mostrará el siguiente botón de actualizar en la barra de navegación



Procedemos a elegir la ruta de la descargar



Cerramos la ventana blanca y esperamos unos 2 a 3 minutos a que se complete la descarga. Nota: la descarga no tiene una interfaz gráfica para ver el proceso, se debe esperar a que se complete, normalmente unos 3 minutos, luego de ese lapso el archivo estará descargado y podrá ejecutarlo.

VINCULACIÓN DE WHATSAPP



Pulsar “verificar WhatsApp” y si no existe ninguna cuenta vinculada se mostrará el QR para emparejar. Si ya se vinculó una cuenta o ya estaba previamente vinculada se mostrará “conectado”



Una vez conectado podremos enviar un mensaje de prueba a un número cualquiera

Estado: MENSAJE ENVIADO

Si el mensaje fue enviado la vinculación de la cuenta de WhatsApp fue exitosa



CREACIÓN DE CUENTAS DE PAGO

Configurar correctamente los precios de los planes

The screenshot displays a management interface for payment plans. It features a search bar with the text "Buscar" and a green "AÑADIR" button. Below is a table with columns for "Código", "Nombre", "Precio", "Clientes", and "Acciones". A single plan is listed with the code "1000", name "Hogar-100 (30MB)", price "180.00", and "2" clients. The "Acciones" column contains edit and delete icons. At the bottom, there is a pagination control showing "Registros por página" set to "20", "1-1 de 1", and navigation arrows.

Código ↑	Nombre	Precio	Clientes	Acciones
1000	Hogar-100 (30MB)	180.00	2	 

Seleccionar el nodo o zona y procedemos a crear cuentas

Código	Cliente	Paga cada	Cuentas	Acciones
1000	Juan Perez 76567027	5		+ ✎
1001	Mario Torrez 76567027			+ ✎

Registros por página: 20 | 1-2 de 2

Masivamente: Presionando el botón añadir en lote, donde podremos ver la cantidad de clientes y seleccionar el periodo de sus cuentas, indicando mes y gestión. Se puede elegir si notificar a los clientes o no.

Añadir Cuenta ✕

Clientes: 2
Nodo: Max Paredes

Mes* 09 **Gestión*** 2024

Notificar al cliente por WhatsApp

CERRAR GUARDAR

Administración de cuentas.

Código	Cliente	Paga cada	Cuentas	Acciones
1000	Juan Perez 76567027	5	09-2024 Bs. 180.00	+ ✎
1001	Mario Torrez 77786114		09-2024 Bs. 180.00	+ ✎

Si el cliente tiene registrado el valor “día de pago” se validará la fecha de pago, si supera la fecha se mostrará en rojo.

Se pueden añadir varias cuentas a un cliente. Para eliminar o modificar alguna cuenta basta con pulsar el botón del lápiz.

Editar Cuentas

Cliente: Juan Perez **Teléfono:** 76567027

Plan: Hogar-100 (30MB) **Precio:** Bs. 180.00

Cuenta

09-2024 Bs. 180.00

Mes* 09 **Gestión*** 2024

Monto* 180.00

ELIMINAR

CERRAR GUARDAR

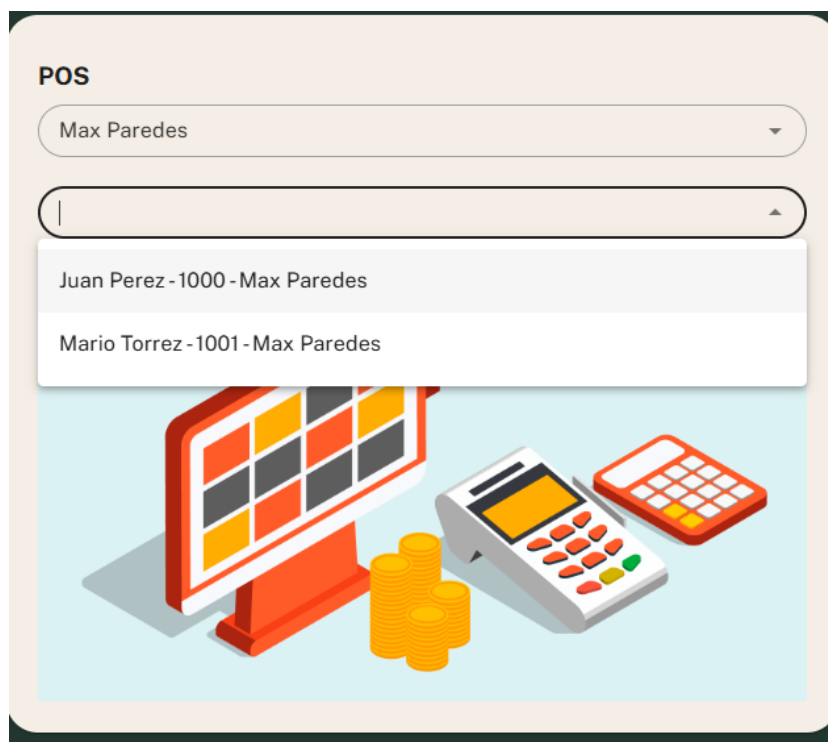
Las cuentas pendientes de pago se mostrarán en “cuentas por cobrar”, teniendo la opción de registrar su pago

Código	Periodo	Cliente	Paga cada	Monto	Acciones
8	09-2024	Juan Perez 76567027	5	Bs. 180.00	
9	09-2024	Mario Torrez 77786114		Bs. 180.00	

Registros por página: 20 | 1-2 de 2

REGISTRO DE PAGOS

Seleccionamos el nodo o zona, y buscamos al cliente que realizará o realizó el pago (solo estarán disponibles clientes con pagos pendientes).



Al seleccionar al cliente se nos mostrará la información de sus cuentas pendientes, y debemos seleccionar la cuenta que pagará. También podremos establecer la fecha de pago, el método de pago y opcionalmente podremos cargar una imagen que podría ser el comprobante de una transferencia o similares.

POS

Max Paredes

Juan Perez - 1000 - Max Paredes

Cliente: Juan Perez **Teléfono:** 76567027 22886225

Nodo: Max Paredes **Plan:** Hogar-100 (30MB)

PAGAR	PERIODO	MONTO
<input type="checkbox"/>	09-2024	Bs. 180.00

Total Parcial: **Bs 0.00**

Descuento:

Total General: **Bs 0.00**

Fecha de Pago* **Método de Pago***

16/09/2024 Efectivo

Referencia



Imprimir al finalizar

Descargar al finalizar

LIMPIAR **REGISTRAR**

Imprimir al finalizar: Si tenemos configurada una impresora térmica al marcar esta opción se imprimirá el respectivo comprobante



DaifoTelecom
NIT: 288188020
LA PAZ - BOLIVIA
OFICINA: Av. Gral. Juan José Torrez N° 444, La Paz
TELÉFONOS: 2215505 / 2215505
EMAIL: info@daifotelecom.net
WEB: www.daifotelecom.net

COMPROBANTE DE PAGO: 1

ZONA: S/N
CLIENTE: Prueba .
C.I.:
PLAN: S/N
TELÉFONO: 76595428

SERVICIO DE INTERNET		
NRO.	PERIODO	MONTO
1	09-2024	Bs. 150.

SUBTOTAL: Bs. 150.00
DESCUENTO: Bs. 15.00
TOTAL: Bs. 135.00
SON: Ciento Treinta Y Cinco BOLIVIANOS 00/100

FECHA DE PAGO: 2024-09-13
MÉTODO DE PAGO: Efectivo



Descargar al finalizar: Con esta opción obtendremos el comprobante en formato digital pdf, el cual puede ser almacenado o enviado al cliente de manera manual



DaifoTelecom

NIT: 288188020

LA PAZ - BOLIVIA

Oficina Central: Av. Gral. Juan José Torrez
N° 444, La Paz

Teléfonos: 2215505 / 2215505

Email: info@daifotelecom.net -
www.daifotelecom.net

COMPROBANTE DE PAGO: 1

ZONA: S/N

CLIENTE: Prueba .

C.I.:

PLAN: S/N

TELÉFONO: 76595428

SERVICIO DE INTERNET

NRO	PERIODO	MONTO
1	09-2024	Bs. 150.00
SUBTOTAL		Bs. 150.00
DESCUENTO		Bs. 15.00
TOTAL		Bs. 135.00

SON: Ciento Treinta y Cinco Pesos
00/100 M.N.



FECHA DE PAGO: 2024-09-13

MÉTODO DE PAGO: Efectivo



GESTIÓN DE PAGOS

Pagos






Max Paredes

Septiembre 2024

Buscar

Código: 5
Estado: Valido
Cliente: Juan Perez 76567027
Detalle:
1. 09-2024 Bs. 180.00
Descuento: 0.00
Total: Bs. 180.00
Metodo Pago: Efectivo
Referencia:
Fecha: 16 sep 2024
Imagen:



Los pagos pueden obtenerse en formato digital o imprimirlo en impresoras térmicas si está configurada, además los pagos pueden anularse. El botón verde del pago nos muestra el comprobante en pdf. El botón azul del pago imprime directamente el comprobante si la impresora está configurada. El botón rojo del pago anula el pago.

Reporte de pagos: Para los reportes existe la opción de hacerlo por nodos o zonas y también obtener el reporte de todas las zonas, basta con seleccionar esa opción y el periodo

Reporte por clientes: Lista a todos los clientes filtrados por nodo o todos dependiendo a los filtros seleccionados. Muestra los pagos de cada cliente en el periodo seleccionado, también muestra las cuentas pendientes del periodo.



REPORTE DE PAGOS

Oficina Central: Av. Gral. Juan José Torrez N° 444, La Paz

Teléfonos: 2215505 / 76772794

Email: info@daifotelecom.net - www.daifotelecom.net

PERIODO 11-2024

Código	Cliente	Nodo	Detalle	Plan	Método de pago	Referencia	Descuento	Cobrado
268	PRUDENCIO CONDORI QUENALLATA	PERIFERICA	11-2024 Bs. 105.00 - Pagado - 04 nov 2024, 16:38	PLAN SOLITO (17 MBPS) Bs. 105.00	Efectivo	CRISTIAN NINA	0.00	Bs. 105.00
299	OMAR BURGOA MAGNE	PERIFERICA	11-2024 Bs. 175.00 - Pagado - 05 nov 2024, 11:05	PLAN ESTUDIANTE (30 MBPS) Bs. 175.00	Efectivo	CRISTIAN NINA	0.00	Bs. 175.00
329	MERI ZURITA ALVAREZ	PERIFERICA	11-2024 Bs. 80.00 - Pagado - 06 nov 2024, 08:29	PLAN CONVENIO Bs. 80.00	Efectivo		0.00	Bs. 80.00
331	DIEGO GAVINCHA	PERIFERICA	11-2024 Bs. 80.00 - Pagado - 06 nov 2024, 09:53	PLAN CONVENIO Bs. 80.00	Efectivo	CRISTIAN NINA	0.00	Bs. 80.00

Reporte por pagos por cliente: Muestra solamente los pagos realizados en el periodo seleccionado de los clientes con los filtros seleccionados



REPORTE DE PAGOS POR CLIENTES

Oficina Central: Av. Gral. Juan José Torrez N° 444, La Paz

Teléfonos: 2215505 / 76772794

Email: info@daifotelecom.net - www.daifotelecom.net

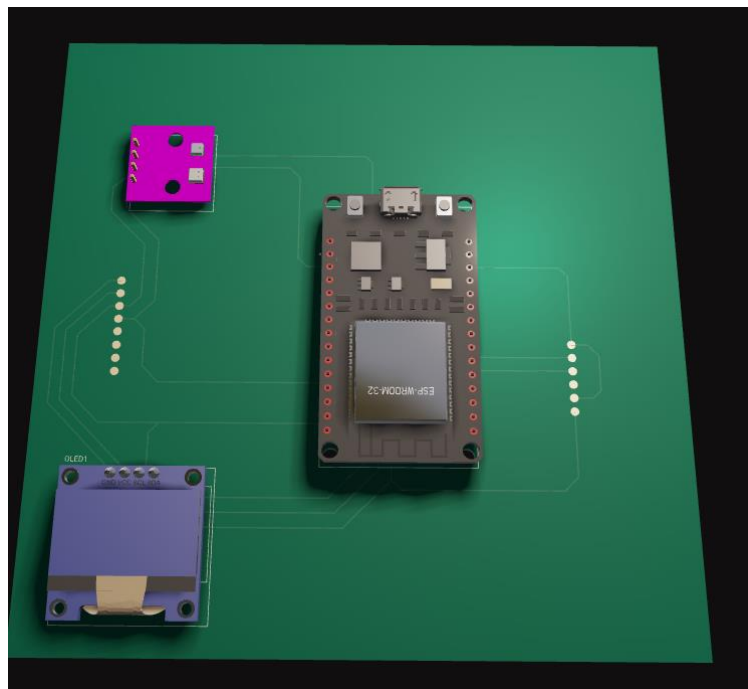
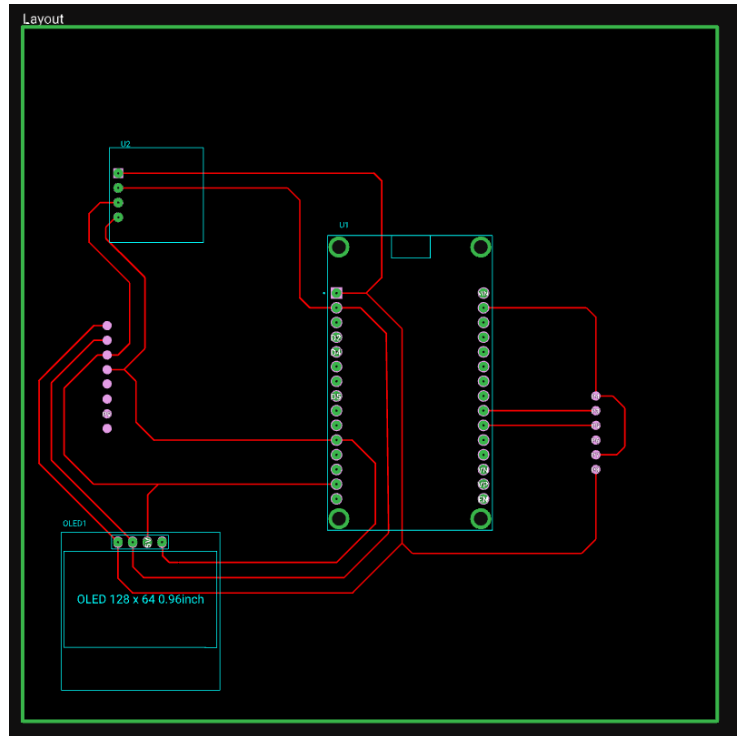
PERIODO 11-2024

Código	Cliente	Teléfono	Día de pago	Nodo	Plan	Detalle	Descuento	Cobrado
1020	MERI ZURITA ALVAREZ	73287282 76534728		PERIFERICA	PLAN CONVENIO Bs. 80.00	11-2024 Bs. 80.00 - Pagado - 06 nov 2024, 08:29 10-2024 Bs. 80.00 - Pagado - 07 oct 2024, 08:28 09-2024 Bs. 80.00 - Anulado - 24 sept 2024, 15:34	Bs. 0.00	Bs. 80.00
1035	DIEGO GAVINCHA	70152710		PERIFERICA	PLAN CONVENIO Bs. 80.00	11-2024 Bs. 80.00 - Pagado - 06 nov 2024, 09:53 10-2024 Bs. 80.00 - Pagado - 14 oct 2024, 16:34	Bs. 0.00	Bs. 80.00
1215	OMAR BURGOA MAGNE	69731799 74919870		PERIFERICA	PLAN ESTUDIANTE (30 MBPS) Bs. 175.00		Bs. 0.00	Bs. 0.00
1109	JUAN PACOSILLO HUANTO	73592011 68078182		PERIFERICA	PLAN ESTUDIANTE (30 MBPS) Bs. 175.00	11-2024 Bs. 175.00 - Pagado - 08 nov 2024, 10:50	Bs. 0.00	Bs. 175.00

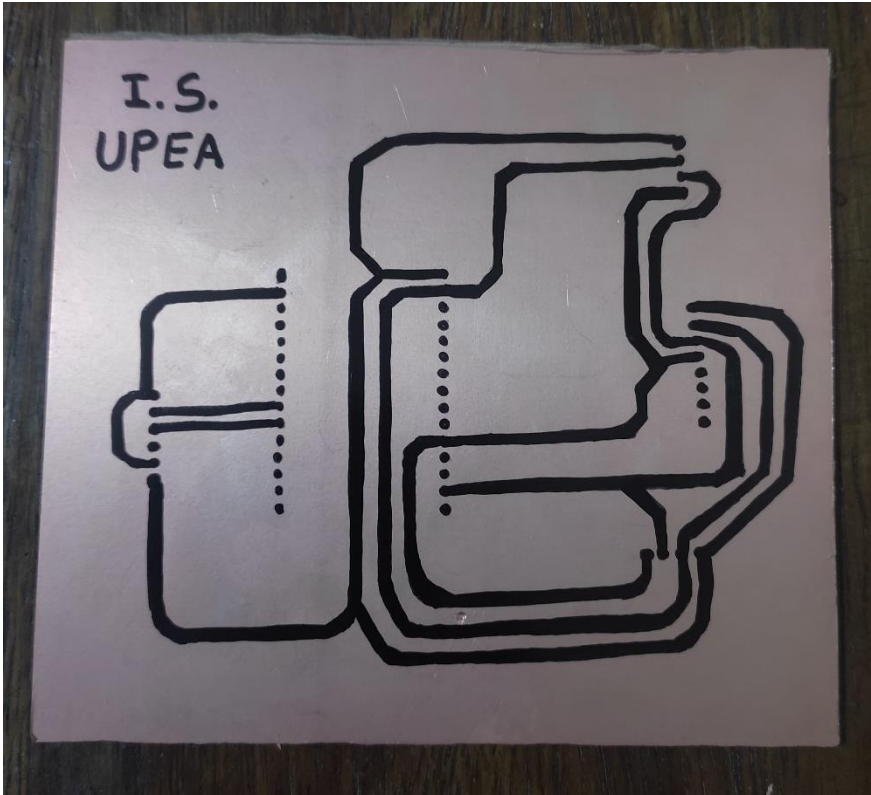
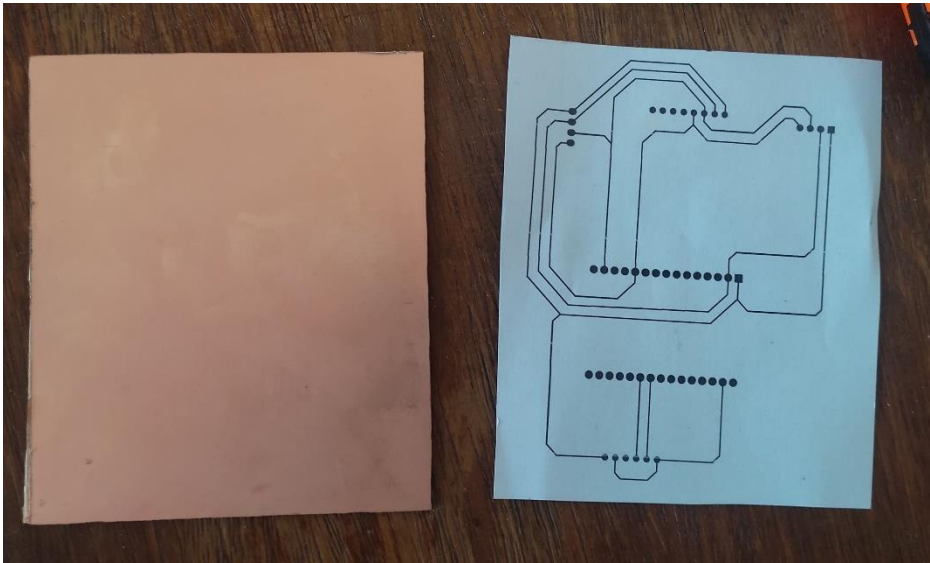
Nota: el primer reporte muestra a todos los clientes y sus pagos, la segunda muestra a todos los clientes que realizaron su pago del periodo

ELABORACIÓN DEL SISTEMA DE MONITOREO

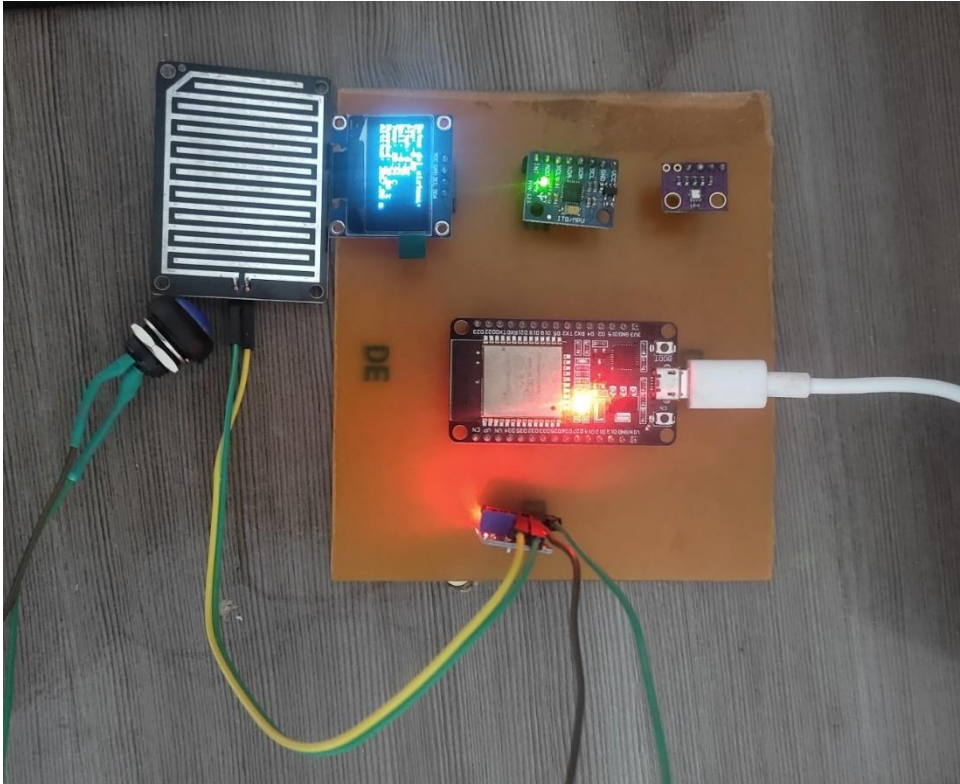
Diseño de la placa



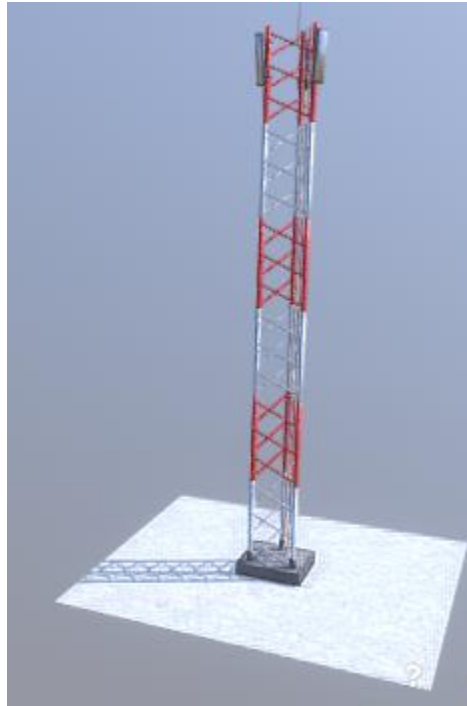
Diseño y quemado de la Placa



Armado de los componentes



Uso de modelo 3D

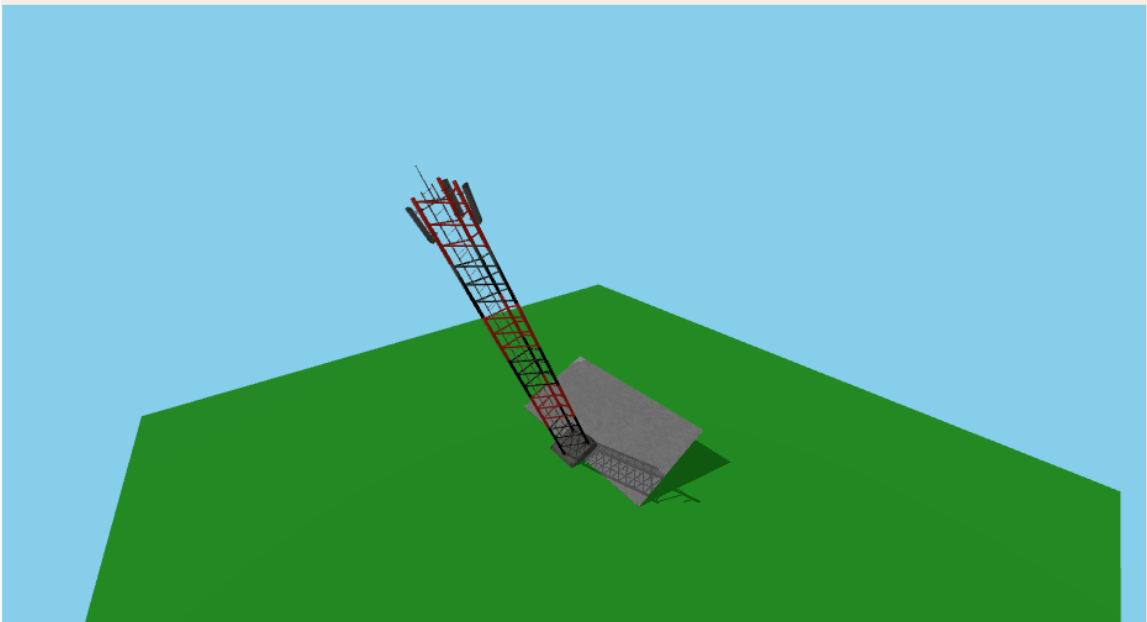


Lectura de datos para visualizar el Roll y Pitch de la Torre

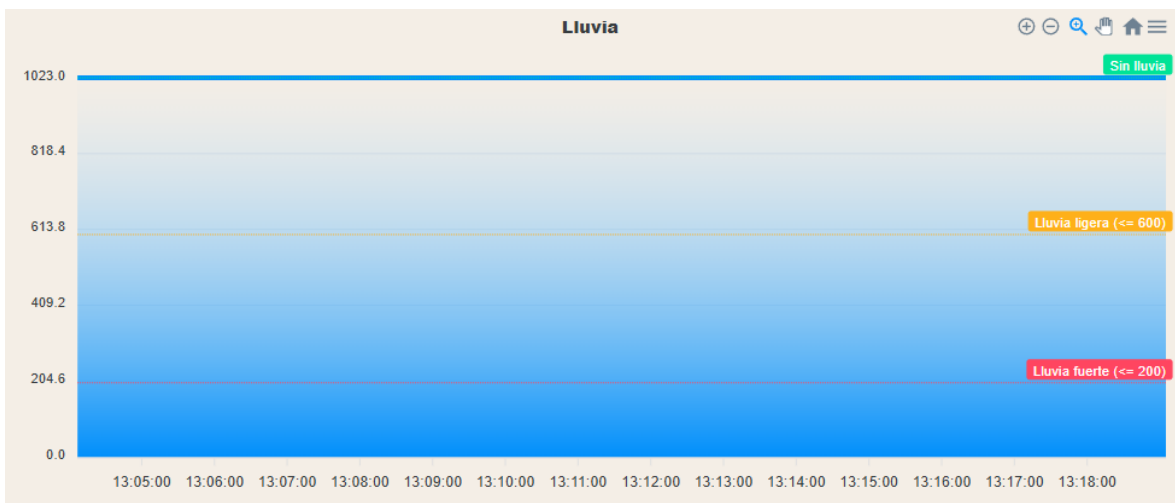
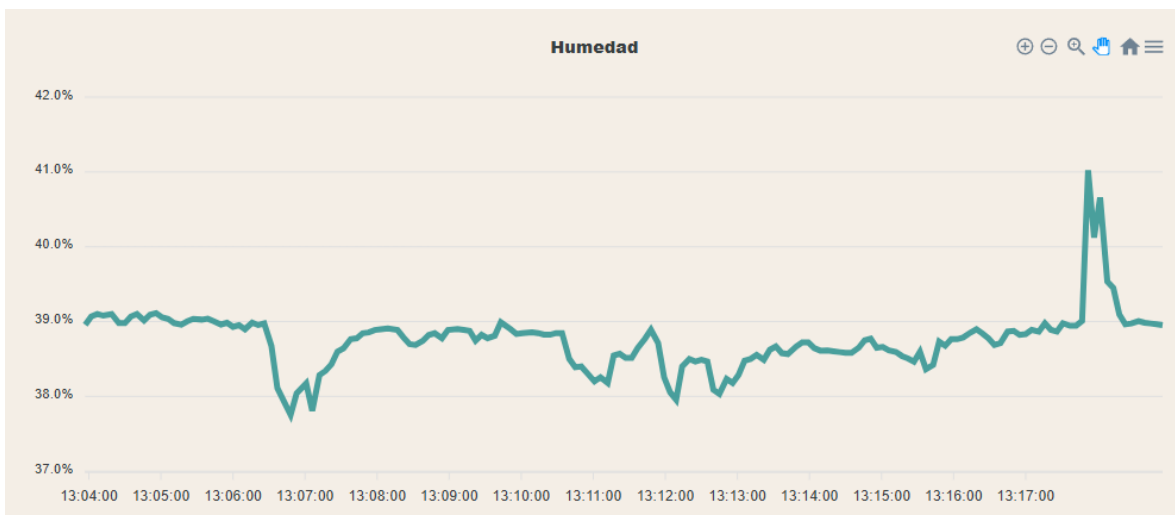
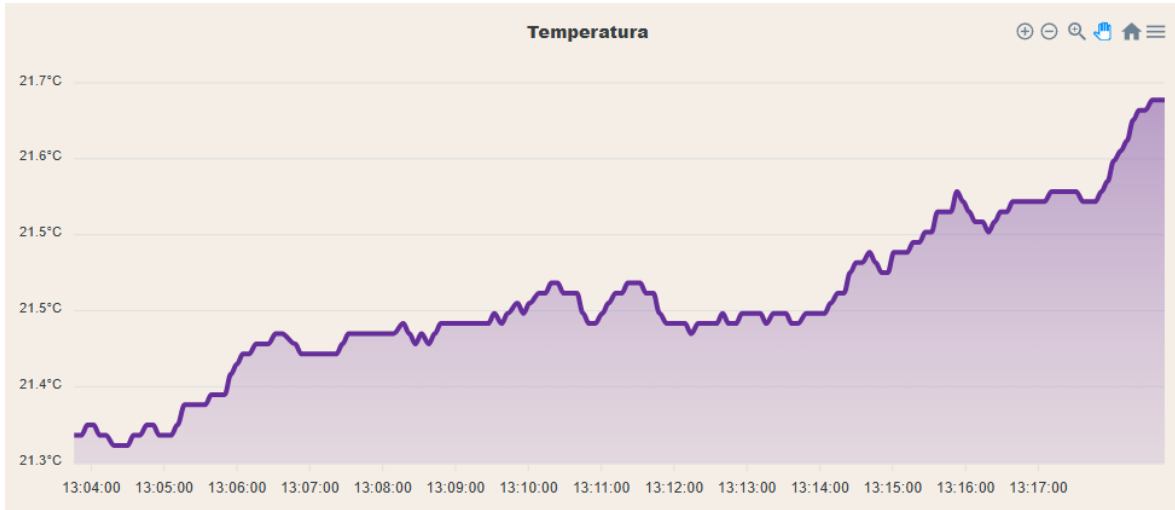
Última Conexión: hace menos de un minuto

Roll: -1.2

Pitch: -25.5



Lectura e interpretación de los datos de Temperatura, Humedad y Lluvia



Diccionario de Datos

Asistencias

Comentarios de la tabla: *Registra la información de las asistencias técnicas*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único de la asistencia
codigo	int	No	0	Código numérico de la asistencia
empresa_id	varchar(100)	No		Identificador de la empresa asociada
cliente_id	varchar(100)	No		Identificador del cliente asociado
user_id	varchar(100)	Sí	NULL	Identificador del usuario asignado (opcional)
tipo_asistencia_id	varchar(100)	No		Identificador del tipo de asistencia
fecha_visita	date	No		Fecha programada para la visita
hora_visita	varchar(100)	No		Hora programada para la visita
descripcion	Text	No		Descripción detallada de la asistencia
estado	varchar(100)	No		Estado actual de la asistencia
observaciones	text	Sí	NULL	Notas adicionales sobre la asistencia
imagenes	longtext	Sí	NULL	Imágenes relacionadas con la asistencia
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última actualización

Checks

Comentarios de la tabla: *Almacena los datos de checks relacionados con inspecciones*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único del check
data	longtext	No		Datos relacionados con el check
survey_id	varchar(100)	No		Identificador de la inspección asociada
createdAt	datetime	No		Fecha y hora de la creación del registro
updatedAt	datetime	No		Fecha y hora de la última actualización

Cientes

Comentarios de la tabla: *Contiene la información detallada de los clientes*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único del cliente
codigo	int	No	0	Código numérico del cliente
empresa_id	varchar(100)	No		Identificador de la empresa asociada
nodo_id	varchar(100)	No		Identificador del nodo asociado al cliente
tipo_cliente_id	varchar(100)	No		Identificador del tipo de cliente
plan_id	varchar(100)	No		Identificador del plan asociado
nombres	varchar(100)	No		Nombres del cliente
apellidos	varchar(100)	No		Apellidos del cliente
ci	varchar(100)	Sí	NULL	Número de documento de identificación (opcional)
estado	varchar(100)	No	Activo	Estado del cliente
dia_pago	int	Sí	NULL	Día del mes asignado para el pago
correo	varchar(100)	Sí	NULL	Correo electrónico del cliente
telefono	varchar(100)	No		Teléfono principal del cliente
telefono2	varchar(100)	Sí	NULL	Teléfono secundario del cliente
otros	varchar(100)	Sí	NULL	Otros datos adicionales (opcional)
ubicacion	varchar(100)	No		Dirección del cliente
foto_carnet	varchar(100)	Sí	NULL	Imagen del documento de identidad (opcional)
foto_factura	varchar(100)	Sí	NULL	Imagen de factura de servicios básicos
foto_casa	varchar(100)	Sí	NULL	Imagen de la vivienda del cliente (opcional)
foto_croquis	varchar(100)	Sí	NULL	Imagen del croquis del cliente
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última actualización

Cuentas

Comentarios de la tabla: *Contiene la información relacionada con las cuentas de los clientes*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único de la cuenta
codigo	int	No	0	Código interno de la cuenta
empresa_id	varchar(100)	No		Identificador de la empresa asociada
pago_ia	varchar(100)	Sí	NULL	Identificador del pago asociado
cliente_id	varchar(100)	No		Identificador del cliente relacionado con la cuenta
plan_id	varchar(100)	No		Identificador del plan asociado a la cuenta
monto	decimal(10,2)	No		Monto total asociado a la cuenta
periodo	varchar(100)	No		Periodo
estado	varchar(100)	No		Estado de la cuenta (por ejemplo, activa o cerrada)
createdAt	datetime	No		Fecha y hora de la creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Datos

Comentarios de la tabla: *Registra las versiones de las aplicaciones utilizadas*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	int	No		
versión_android	varchar(100)	No		Versión actual de la app
version_desktop	varchar(100)	No		Versión actual de la aplicación escritorio
createdAt	datetime	No		Fecha y hora de creación
updatedAt	datetime	No		Fecha y hora de la última modificación

Empresa

Comentarios de la tabla: *Almacena la información general de las empresas registradas*

Columna	Tipo	Nulo	Prede	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único de la empresa
qr	varchar(100)	Sí	Null	Código qr para cobros
texto_notificacion_cuenta	text	Sí	NULL	Texto personalizado para notificaciones relacionadas con cuentas

Columna	Tipo	Nulo	Prede	Comentarios
nombre	varchar(100)	No		Nombre de la empresa
color	varchar(100)	No		Color corporativo asociado a la empresa
logo	varchar(100)	No		Logo de la empresa
direccion_oficina	varchar(100)	No		Dirección de la oficina principal de la empresa
teléfonos	longtext	No		Teléfonos de contacto de la empresa
nit	varchar(100)	Sí	NULL	Número de Identificación Tributaria (opcional)
email	varchar(100)	No		Correo electrónico de la empresa
sitio_web	varchar(100)	No		Sitio web oficial de la empresa
smtpt	longtext	No		Credenciales SMTP
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Monitoreos

Comentarios de la tabla: *Registra las mediciones y monitoreos realizados en las torres*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único del monitoreo
torre_id	varchar(100)	No		Identificador de la torre
ejex	float	Sí	NULL	Valor del eje X
ejeY	float	Sí	NULL	Valor del eje Y
ejeZ	float	Sí	NULL	Valor del eje Z
lluvia	int	Sí	NULL	Cantidad de lluvia
temperatura	float	Sí	NULL	Temperatura registrada (opcional)
humedad	float	Sí	NULL	Humedad registrada
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Nodos

Comentarios de la tabla: *Registra los nodos asociados a las empresas*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único del nodo
codigo	int	No	0	Código interno del nodo
nombre	varchar(100)	No		Nombre descriptivo del nodo
empresa_id	varchar(100)	No		Identificador de la empresa asociada
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Ordenes

Comentarios de la tabla: *Registra las órdenes de servicio generadas*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único de la orden
codigo	varchar(100)	No		Código interno de la orden
empresa_id	varchar(100)	No		Identificador de la empresa asociada
personal_contacto	varchar(100)	No		Nombre del personal de contacto
cliente_id	varchar(100)	No		Identificador del cliente relacionado con la orden
user_id	varchar(100)	No		Identificador del usuario asignado a la orden
conexion	longtext	Sí	NULL	Detalles técnicos sobre la conexión del cliente
diagnostico	longtext	Sí	NULL	Diagnóstico del trabajo
fecha	date	No		Fecha de emisión de la orden
hora_inicio	varchar(100)	No		Hora de inicio del trabajo
hora_fin	varchar(100)	No		Hora de finalización del trabajo
observaciones	text	Sí	NULL	Notas adicionales
firma_cliente	varchar(100)	Sí	NULL	Firma digital del cliente (opcional)
firma_tecnico	varchar(100)	Sí	NULL	Firma digital del técnico (opcional)
imagenes	longtext	Sí	NULL	Imágenes relacionadas con la orden
createdAt	datetime	No		Fecha y hora de la creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Ordenservicio

Comentarios de la tabla: *Relaciona órdenes con servicios específicos realizados*

Columna	Tipo	Nulo	Predet	Comentarios
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación
orden_id (Primaria)	varchar(100)	No		Identificador de la orden asociada
servicio_id (Primaria)	varchar(100)	No		Identificador del servicio asociado

Pagos

Comentarios de la tabla: *Registra los pagos realizados por los clientes*

Columna	Tipo	Nulo	Prede	Comentarios
id (Primaria)	varchar(100)	No		Identificador único del pago
codigo	int	No	0	Código interno del pago
empresa_id	varchar(100)	No		Identificador de la empresa asociada al pago
fecha_pago	date	No		Fecha de pago
metodo_pago	varchar(100)	No		Método de pago utilizado (por ejemplo, transferencia, efectivo)
referencia_pago	varchar(100)	Sí	NULL	Referencia del pago
imagen	varchar(100)	Sí	NULL	Imagen del comprobante de pago (opcional)
descuento	decimal(10,2)	No		Monto de descuento
monto	decimal(10,2)	No		Monto base del pago antes de descuentos
total	decimal(10,2)	No		Monto total
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación
estado	varchar(100)	No	Valido	Estado del pago (por ejemplo, Válido o Anuado)

Permissions

Comentarios de la tabla: *Registra los permisos disponibles en el sistema*

Columna	Tipo	Nulo	Predet	Comentarios
id (Primaria)	varchar(100)	No		Identificador único del permiso

Columna	Tipo	Nulo	Predet	Comentarios
action	varchar(100)	No		Acción permitida
subject	varchar(100)	No		Recurso sobre el cual se aplica el permiso

Planes

Comentarios de la tabla: *Registra los planes disponibles para los clientes*

Columna	Tipo	Nulo	Predet	Comentarios
id (Primaria)	varchar(100)	No		Identificador único del plan
codigo	int	No	0	Código interno del plan
precio	decimal(10,2)	No		Precio del plan
empresa_id	varchar(100)	No		Identificador de la empresa
nombre	varchar(100)	No		Nombre del plan
createdAt	datetime	No		Fecha y hora de la creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Registros

Comentarios de la tabla: *Almacena los registros de actividad realizados en el sistema*

Columna	Tipo	Nulo	Predet	Comentarios
id (Primaria)	bigint	No		
codigo	int	No	0	Código interno del registro
usuario_id	varchar(100)	No		Identificador del usuario que generó el registro
detalle	varchar(100)	No		Descripción breve del registro
accion	varchar(100)	No		Acción realizada (por ejemplo, insertar, actualizar)
empresa_id	varchar(100)	Sí	NULL	Identificador de la empresa
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Rolepermission

Comentarios de la tabla: *Relaciona roles con los permisos asignados*

Columna	Tipo	Nulo	Predet	Comentarios
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de modificación

Columna	Tipo	Nulo	Predet	Comentarios
roleId (<i>Primaria</i>)	varchar(100)	No		Identificador del rol asociado
permissionId(<i>Primaria</i>)	varchar(100)	No		Identificador del permiso asociado

Roles

Comentarios de la tabla: *Define los roles disponibles en el sistema*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único del rol
nombre	varchar(100)	No		Nombre del rol

Servicios

Comentarios de la tabla: *Registra los servicios disponibles ofrecidos por las empresas*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único del servicio
nombre	varchar(100)	No		Nombre del servicio ofrecido
codigo	int	No	0	Código interno del servicio
empresa_id	varchar(100)	No		Identificador de empresa
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Surveys

Comentarios de la tabla: *Almacena las inspecciones realizadas por las empresas*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único de la inspección
codigo	int	No	0	Código interno de la inspección
empresa_id	varchar(100)	No		Identificador de la empresa asociada a la inspección
Información_general	longtext	No		Información general de la inspección
informacion_empresa	longtext	No		Información sobre la empresa recopilada en la inspección
información_torre	longtext	No		Información de la torre
data	longtext	No		Datos generales en la inspección

Columna	Tipo	Nulo	Predet	Comentarios
archivos	longtext	No		Archivos adjuntos
tecnicos	longtext	No		Información de los técnicos involucrados en la inspección
createdAt	datetime	No		Fecha y hora de la creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Tiposasistencias

Comentarios de la tabla: *Define los tipos de asistencias técnicas disponibles*

Columna	Tipo	Nulo	Predet	Comentarios
id (Primaria)	varchar(100)	No		Identificador único del tipo de asistencia
codigo	int	No	0	Código interno del tipo de asistencia
nombre	varchar(100)	No		Nombre descriptivo del tipo de asistencia
empresa_id	varchar(100)	No		Identificador de la empresa
createdAt	datetime	No		Fecha y hora de creación del registro
updateAt	datetime	No		Fecha y hora de la última modificación

Tiposclientes

Comentarios de la tabla: *Define los tipos de clientes registrados en el sistema*

Columna	Tipo	Nulo	Predet	Comentarios
id (Primaria)	varchar(100)	No		Identificador único del tipo de cliente
codigo	int	No	0	Código interno del tipo de cliente
nombre	varchar(100)	No		Nombre descriptivo del tipo de cliente
empresa_id	varchar(100)	No		Identificador de la empresa asociada
createdAt	datetime	No		Fecha y hora de creación del registro
updatedAt	datetime	No		Fecha y hora de la última modificación

Torres

Comentarios de la tabla: *Registra la información de las torres gestionadas*

Columna	Tipo	Nulo	Predet	Comentarios
id (Primaria)	varchar(100)	No		Identificador único de la torre
nombre	varchar(100)	No		Nombre descriptivo de la torre
descripción	text	Sí	NULL	Descripción de la torre

fecha_actualización	datetime	Sí	NULL	Fecha de la última actualización de la torre (opcional)
ubicacion	longtext	Sí	NULL	Ubicación de la torre
createdAt	datetime	No		Fecha y hora de creación del registro ccr
updatedAt	datetime	No		Fecha y hora de la última modificación

Usuarios

Comentarios de la tabla: *Registra la información de los usuarios del sistema*

Columna	Tipo	Nulo	Predet	Comentarios
id (<i>Primaria</i>)	varchar(100)	No		Identificador único del usuario
role	varchar(100)	Sí	NULL	Nombre del rol asignado al usuario (opcional)
role_id	varchar(100)	Sí	NULL	Identificador del rol asignado al usuario (opcional)
codigo	int	No	0	Código interno del usuario
nombres	varchar(100)	No		Nombres del usuario
apellido	varchar(100)	No		Apellido del usuario
telefono	varchar(100)	No		Teléfono de contacto del usuario
correo	varchar(100)	Sí	NULL	Correo electrónico del usuario (opcional)
password	varchar(100)	No		Contraseña cifrada del usuario
activo	tiny (1)	No	11	Estado de usuario 1 (activo), 0 (inactivo)
createdAt	datetime	No		Fecha y hora de creación del registro
updateAt	datetime	No		Fecha y hora de la última modificación

INSTALACIÓN DEL SISTEMA

BACKEND NODE

Antes de proceder con la instalación, asegúrese de cumplir con los siguientes requisitos:

Software Necesario:

- Node.js: Versión 18 o superior (se recomienda la última versión estable).
- Gestor de bases de datos: MySQL versión 8.0 o superior.
- NPM: Incluido con Node.js, para gestionar dependencias del proyecto.

Hardware Recomendado:

- Procesador: Mínimo 2 núcleos a 2.0 GHz.
- Memoria RAM: 4 GB o superior.
- Almacenamiento: 10 GB disponibles.
- Sistema operativo: Windows 10/11, macOS, o distribuciones Linux (Ubuntu recomendado).

Pasos de Instalación

Descargar el Código Fuente:

Obtenga el paquete del sistema desde la fuente autorizada (como un archivo comprimido .zip)

Extraiga el contenido en la ubicación deseada en su equipo.

Configurar el Entorno:

Configure las variables de entorno necesarias, como el puerto de la aplicación y la conexión a la base de datos:

DB_NAME=ABC

DB_USER=root

DB_PASSWORD=

RUTA_PUBLICA=http://localhost:3000

ALLOWEDORIGINS=http://localhost:5173

Instalar Dependencias:

Abra una terminal en la carpeta del proyecto y ejecute el siguiente comando para instalar las dependencias:

```
npm install
```

Inicie la aplicación con:

```
npm run dev
```

La aplicación estará disponible en el puerto configurado, por ejemplo: http://localhost:3000.

Probar el Sistema:

Acceda a la interfaz de usuario mediante un navegador compatible.

Use las credenciales predeterminadas para iniciar sesión (si están configuradas).

Aplicativos

El sistema incluye módulos adicionales y aplicativos específicos que serán distribuidos y configurados internamente:

Compartición Interna:

Los aplicativos o módulos se compartirán exclusivamente entre los equipos a través del personal superior inmediato.

Las credenciales o accesos adicionales deben solicitarse al administrador del sistema.

Distribución Interna:

Las actualizaciones o nuevas versiones serán comunicadas oficialmente y distribuidas mediante canales internos autorizados.

Solución de Problemas Comunes

Error al conectar la base de datos:

Verifique las credenciales y que el servicio MySQL esté activo.

Asegúrese de que el archivo .env contenga los valores correctos.

Dependencias no instaladas:

Ejecute npm install nuevamente para asegurarse de que todas las dependencias estén instaladas.

Problemas con Node.js:

Verifique la versión instalada con:

```
node -v
```

Actualice si es necesario.

FRONTEND VUE

Antes de proceder con la configuración del frontend, asegúrese de contar con lo siguiente:

Software Necesario:

- Node.js: Versión 18 o superior (ya instalado en los pasos anteriores).
- NPM: Incluido con Node.js, para instalar dependencias del proyecto.

Entorno Recomendado:

- Editor de código: Visual Studio Code o cualquier editor compatible con JavaScript.
- Navegador: Google Chrome o Mozilla Firefox (se recomienda instalar extensiones como Vue Devtools para depuración).

Pasos de Instalación

Descargar el Código Fuente del Frontend:

Obtenga los archivos del frontend desde la fuente autorizada (un archivo comprimido .zip).

Extraiga el contenido en una carpeta de su equipo.

Configurar el Entorno del Frontend:

Localice el archivo de configuración en src/router/endpoint.js y verifique que las URL del backend estén correctamente configuradas:

```
const url = "http://localhost:3000/"; // API base para rutas protegidas
const urlPublic = "http://localhost:3000"; // API base para rutas públicas
```

Asegúrese de que estas URL coincidan con la dirección del servidor backend donde está corriendo la API.

Instalar Dependencias:

Abra una terminal en la carpeta raíz del proyecto frontend y ejecute:

```
npm install
```

Ejecutar el Servidor de Desarrollo:

Inicie el servidor del frontend para pruebas locales con:

```
npm run dev
```

La aplicación estará disponible en el navegador en <http://localhost:8080> por defecto.

Construir para Producción:

Si desea preparar el frontend para un entorno de producción, ejecute:

Copiar código

```
npm run build
```

Esto generará los archivos optimizados en la carpeta `/dist`, listos para ser desplegados en un servidor web.

Probar el Sistema

Abra un navegador y acceda al URL del servidor frontend (por ejemplo, <http://localhost:8080>).

Inicie sesión o explore las funcionalidades disponibles según la configuración del backend.

Integración con el Backend:

Asegúrese de que las rutas del backend (APIs) estén funcionando correctamente y devuelvan los datos esperados.

ANEXO C

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO

