

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas Carrera de Software

APLICACIÓN MÓVIL PARA MEJORAR LA CALIDAD DE SERVICIO EN LA DISTRIBUCIÓN DE CILINDROS DE GAS DOMÉSTICOS EN LA ZONA URBANA DE LA CIUDAD DE IBARRA.

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

Autores:

Carlosama Carlosama Edgar Diego
Quinatoa Ulcuango Paúl Alexander

Director:

Ing. Jorge Adrián Caraguay Procel MSc.

Ibarra - Ecuador 2023

1. IDENTIFICACIÓN DE LA OBRA

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

DATOS DEL CONTACTO 1			
CÉDULA DE IDENTIDAD:	DE	100441592-1	
APELLIDOS Y NOMBRES:	Y	Carlosama Carlosama Edgar Diego	
DIRECCIÓN:		Avenida Carchi y María Angelica Idrobo	
EMAIL:		edcarlosamac@utn.edu.ec edcarlosama_uid23@outlook.com	
TELÉFONO FIJO:		TELÉFONO MÓVIL:	0999903204

DATOS DEL CONTACTO 2			
CÉDULA DE IDENTIDAD:	DE	172803901-5	
APELLIDOS Y NOMBRES:	Y	Quinatoa Ulcuango Paúl Alexander	
DIRECCIÓN:		Cayambe-Olmedo-Pesillo	
EMAIL:		paquinatoau@utn.edu.ec	
TELÉFONO FIJO:		TELÉFONO MÓVIL:	0963117079

DATOS DE LA OBRA			
TÍTULO:	Aplicación móvil para mejorar la calidad de servicio en la distribución de cilindros de gas domésticos en la zona urbana de la ciudad de Ibarra.		
AUTOR (ES):	Carlosama Carlosama Edgar Diego Quinatoa Ulcuango Paúl Alexander		
FECHA:	26/10/2023		
PROGRAMA:	<input checked="" type="checkbox"/>	PREGRADO	<input type="checkbox"/> POSTGRADO
TÍTULO POR EL QUE OPTA:	Ingeniería de software		
DIRECTOR:	Ing. Jorge Adrián Caraguay Procel MSc.		

2. CONSTANCIAS

El autor(es) manifiestan que la obra objeto de la presente autorización es original y se la desarrollo, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es(son) el (los) titular (es) de los derechos patrimoniales, que lo que asume(n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 26 días de octubre de 2023

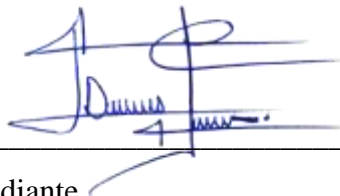
AUTORES



Estudiante

Paúl Alexander Quinatoa Ulcuango

C.I.: 1728039015



Estudiante

Diego Edgar Carlosama Carlosama

C.I.: 1004415921

CERTIFICADO DEL DIRECTOR

Ibarra, 19 de octubre del 2023

CERTIFICADO DIRECTOR DEL TRABAJO DE TITULACIÓN

Por medio del presente yo Ing. Jorge Caraguay, certifico que los Sres. Diego Carlosama portador de la cedula de ciudadanía número 100441592-1 y Alexander Quíntoa portador de la cedula de ciudadanía 172803901-5, han trabajado en el desarrollo del proyecto de grado "APLICACIÓN MÓVIL PARA MEJORAR LA CALIDAD DE SERVICIO EN LA DISTRIBUCIÓN DE CILINDROS DE GAS DOMÉSTICOS EN LA ZONA URBANA DE LA CIUDAD DE IBARRA", previo a la obtención del Título de Ingeniero en Software, este trabajo se ha realizado con interés profesional y responsabilidad

Atentamente,



Jorge Adrián Caraguay Procel
C.I.: 1102451687
CARRERA DE SOFTWARE
UNIVERSIDAD TÉCNICA DEL NORTE



UNIVERSIDAD TÉCNICA DEL NORTE

UNIVERSIDAD ACREDITADA

Resolución No. 173-SE-33 CACES – 2020

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

Ibarra, 23 de febrero del 2023

CARTA DE AVAL

Yo, Jorge Caraguay, en calidad de docente de la Carrera de Software de la Universidad Técnica del Norte, doy el aval técnico-científico al trabajo de titulación: **"APLICACIÓN MÓVIL PARA MEJORAR LA CALIDAD DE SERVICIO EN LA DISTRIBUCIÓN DE CILINDROS DE GAS DOMÉSTICOS EN LA ZONA URBANA DE LA CIUDAD DE IBARRA"**, de los señores estudiantes Edgar Diego Carlosama Carlosama, Paul Alexander Quinatoa Ulcuango, esta temática es parte de mi experticia que se encuentra como una sublínea de interés del área de investigación y formación académica.

Atentamente,

Ing. Jorge Adrián Caraguay Procel MSc.

C.I.: 1102451687

CARRERA DE SOFTWARE

UNIVERSIDAD TÉCNICA DEL NORTE

Ciudadela Universitaria UTN
Av. 17 de Julio 5-21

Telf: +593 62997800
e-mail: info@utn.edu.ec

UTN
IBARRA - ECUADOR
Vive
sueña
construye

Dedicatoria (Diego Carlosama)

“Este trabajo es dedicado a la memoria de Hugo Alexis Salazar Mar, nuestro amigo y compañero de la universidad que falleció trágicamente antes de culminar sus estudios. Hugo era una persona inteligente, perseverante, y sobre todo él me inspiró a continuar con el tema de integración curricular que no se pudo concretar por el desafortunado acontecimiento. Agradecido por haber compartido las aulas con nosotros en la universidad, siempre lo recordaremos. También dedico con especial cariño a mis padres y familiares que estuvieron presentes en el proceso de formación académica”

Diego Carlosama

Dedicatoria (Alexander Quinatoa)

Dedico este trabajo a toda mi familia, con un cariño especial para mi madre, Ernestina, y mi hermano, Mauricio. A ellos, les agradezco por su inquebrantable confianza en mí y por su apoyo incondicional a lo largo de mi carrera académica. Sus valores y dedicación son una fuente constante de inspiración para seguir avanzando y lograr mis metas.

A mis amigos, con quienes he compartido momentos inolvidables y cuya amistad fue pilar fundamental en mi vida. Su compañía ha enriquecido mi camino y mi desarrollo personal.

A mi compañero de aula, Brayan, por compartir su conocimiento y contribuir a mi aprendizaje. Agradezco sus enseñanzas.

A mi enamorada Cielo Jazmín, por su apoyo incondicional en esta etapa de mi vida. Tu apoyo ha sido una constante fuente de fortaleza y motivación. Aprecio profundamente tu presencia en este viaje.

Alexander Quinatoa

Agradecimientos

Queremos manifestar nuestro profundo agradecimiento por la colaboración y apoyo durante el desarrollo del trabajo de integración curricular. Sus consejos y recomendaciones han sido esenciales para alcanzar los objetivos propuestos.

En primer lugar, queremos agradecer al Ing. Jorge Caraguay, director de tesis, por su asesoría, paciencia, dedicación y por transmitir su conocimiento, por impulsarnos a seguir adelante y por apoyarnos en todo momento.

Así mismo, queremos expresar nuestros agradecimientos a los docentes de la universidad, especialmente al Ing. Alexander Guevara, Ing. Iván García, Ing. Diego Trejo y al Ing. Mauricio Rea, por su valiosa aportación en la formación académica. Gracias por compartir sus experiencias, por enseñarnos a pensar críticamente, a investigar por cuenta propia y a desarrollar soluciones innovadoras.

Finalmente, agradecemos a la familia y los amigos que estuvieron presentes en todo momento, por su amor, comprensión y apoyo incondicional. Gracias por ser nuestra inspiración y por animarnos a seguir adelante en la búsqueda de metas. Sin su ayuda y aliento, este logro no habría sido posible. ¡Gracias de todo corazón!

Atentamente,

Diego Carlosama, Alexander Quinatoa

Tabla de contenidos

Dedicatoria (Diego Carlosama)	VII
Dedicatoria (Alexander Quinatoa)	VIII
Agradecimientos	IX
Tabla de contenidos	X
Índice de tablas	XV
Índice de figuras	XVII
Resumen	1
Abstract	3
INTRODUCCIÓN	5
Planteamiento del problema	5
Objetivos	7
Objetivo General.....	7
Objetivos Específicos	7
Alcance	7
Frontend Aplicación móvil (Flutter).....	8
BackEnd como web service (API REST).....	8
Firebase como gestor de datos.....	8
Indicadores de calidad	8
Módulos.....	9
Validación de pruebas	9
Metodología	10
Justificación	11
Riesgos	12
CAPÍTULO I	15
1.1 Marco teórico	15
1.2 Revisión de literatura SLR	15
1.2.1 Determinar las Preguntas de Investigación.	16
1.2.2 Búsqueda de Documentos.	17

1.2.3	Selección de Artículos - Criterios de Inclusión y Exclusión.	18
1.2.4	Extracción de Datos Relevantes.	19
1.3	Situación actual	20
1.3.1	Situación Actual En La Zona Urbana De La Ciudad De Ibarra.	20
1.3.2	Cuestionario e Interpretación de Datos de la Encuesta.	20
1.3.3	Entrevista e interpretación de Datos.	24
1.4	Tecnologías de geolocalización	25
1.4.1	Geolocalización	25
1.4.2	Sistema de Posicionamiento Global (GPS)	27
1.4.3	Componentes del Sistema GPS y funcionamiento	28
1.4.4	Dispositivos en red mediante direcciones IP	30
1.4.5	Aplicaciones móviles con geolocalización.....	31
1.5	Técnicas, metodologías y licitación de requisitos	32
1.5.1	ISO-IEC-IEEE 29148-2011.....	32
1.5.2	Metodología Extreme Programming (XP)	33
1.6	Sistemas Operativos y aplicaciones	34
1.6.1	Sistema Operativo Android	36
1.6.2	Sistema Operativo IOS	36
1.6.3	Análisis de los sistemas operativos móviles más populares: iOS y Android ...	37
1.5.5	Aplicación móvil	37
1.5.6	Tipos de aplicaciones.....	38
1.5.6.1	Nativas	38
1.5.6.2	Web.....	39
1.5.6.3	Híbridas	39
1.7	Herramientas de desarrollo	39

1.7.1 Entorno de desarrollo Node.JS	40
1.7.2 Framework Fllutter	41
1.7.3 Gestor de base de datos	42
1.7.4 Servicios web.....	43
1.7.6 Android Estudio.....	45
1.7.7 Visual estudio Code.....	47
1.7.8 Fly.io.....	48
1.7.9 GitHub	49
1.7.10 Postman.....	49
1.7.11 Adode XD	50
1.7.12 Software Estadístico Epidat	51
1.7.13 Software para pruebas JMeter.....	52
1.7.14 Indicadores de herramientas de desarrollo.....	53
1.7.15 Importancia de la Usabilidad	54
1.7.16 Trabajos relacionados	57
1.6.17 Indicadores de trabajos relacionados	60
CAPÍTULO 2	61
2 Documento de Especificación de Requerimientos del Software (SRS – ISO/IEEE 29148)	61
2.1 Introducción	61
2.1.1 Propósito.....	61
2.1.2 Visión General del Producto.....	61
2.1.3 Perspectiva del producto.....	61
2.1.4 Funciones del Producto	61
2.1.5 Características del usuarios y limitaciones.....	62
2.2 Requerimientos específicos	63
2.2.1 Interfaz Externa	63

2.2.2	Funciones.....	63
CAPITULO 3		70
3	DESARROLLO DEL APLICATIVO MOVIL APLICANDO LA METODOLOGÍA XP	70
3.1	Fases de la metodología XP	70
3.1.1	Planificación.....	70
3.1.2	Diseño.....	82
3.1.3	Codificación.....	97
3.1.4	Pruebas.....	101
CAPITULO 4		121
4	Análisis de Usabilidad y Eficiencia en Aplicación Móvil con ISO 25010	121
4.1	Evaluación de Cumplimiento de Usabilidad	121
4.1.1	Evaluación de Pruebas Metodológicas de Usabilidad.....	121
4.1.2	Inteligibilidad.....	122
4.1.1	Aprendizaje.....	123
4.1.2	Operabilidad.....	124
4.1.3	Protección frente a errores.....	125
4.1.4	Estética.....	125
4.1.3	Accesibilidad (Colorimetría).....	126
4.2	Evaluación de eficiencia de desempeño	129
4.2.1	Escenario 1.....	133
4.2.2	Escenario 2.....	135
4.2.3	Escenario 3.....	137
CAPITULO 5		140
5	Validación de los Resultados de la Investigación	140
5.1	Resultados	140
5.2	Tamaño de Muestra	142
5.3	Análisis de la Encuesta CSUQ	143

5.4	Pruebas de experimentación	148
5.1	Discusión de resultados	151
	CONCLUSIONES	154
	RECOMENDACIONES	156
	BIBLIOGRAFÍA	157
	ANEXOS	166

Índice de tablas

Tabla 1	<i>Riesgos y estrategias de mitigación</i>	12
Tabla 2	<i>Preguntas de investigación de la revisión literaria</i>	16
Tabla 3	<i>Cadenas de búsqueda utilizado en librerías digitales</i>	18
Tabla 4	<i>Cuestionario sobre nivel de conocimiento de aplicaciones móviles enfocado a los usuarios</i>	21
Tabla 5	<i>Preguntas para identificar el proceso de distribución de gas domestico</i>	24
Tabla 6	<i>Ejemplo de usos de geolocalización en distintos escenarios</i>	25
Tabla 7	<i>Diferencias entre Geolocalización y Georreferenciación</i>	27
Tabla 8	<i>Funcionamiento del Sistema GPS, Escenario de ejemplo con tres satélites</i>	29
Tabla 9	<i>Prácticas de la metodología XP</i>	33
Tabla 10	<i>Características de un sistema operativo</i>	35
Tabla 11	<i>Sistemas operativos más populares más utilizados</i>	37
Tabla 12	<i>Diferencias entre SQL y NoSQL</i>	42
Tabla 13	<i>Características de REST, SOAP, GAPHQL</i>	44
Tabla 14	<i>Características del entorno de desarrollo Android estudio</i>	46
Tabla 15	<i>Características técnicas de Fly.io</i>	48
Tabla 16	<i>Guía de usabilidad móvil</i>	55
Tabla 17	<i>Indicadores de contenido en los trabajos relacionados</i>	60
Tabla 18	<i>Características de los usuarios y limitaciones</i>	62
Tabla 19	<i>Requerimientos no funcionales</i>	69
Tabla 20	<i>Módulos del proyecto</i>	70
Tabla 21	<i>Planificación de iteraciones</i>	71
Tabla 22	<i>Tarea 1.1 – Generación y envío de código de verificación</i>	72
Tabla 23	<i>Tarea 1.2 – Verificación y reenvío de códigos</i>	72
Tabla 24	<i>Tarea 2.1 – Reutilización de componentes para el inicio de sesión</i>	73
Tabla 25	<i>Tarea 2.2 Almacenamiento seguro de la información personal</i>	73
Tabla 26	<i>Tarea 2.3 – Identificación del rol y enrutamiento con la interfaz</i>	74
Tabla 27	<i>Tarea 3.1 – Realizar pedido a los distribuidores</i>	74
Tabla 28	<i>Tarea 3.2 – Cancelar Pedido</i>	74
Tabla 29	<i>Tarea 3.3 – Graficar ruta del cliente a distribuidor</i>	75
Tabla 30	<i>Tarea 3.4 – Notificación de aceptación de pedido del distribuidor</i>	75

Tabla 31	<i>Tarea 3.5 – Notificación de llegada del distribuidor</i>	76
Tabla 32	<i>Tarea 4.1 – Recopilar información del distribuidor</i>	76
Tabla 33	<i>Tarea 4.2 Recopilar información del vehículo</i>	77
Tabla 34	<i>Tarea 5.1 – Recepción de pedidos</i>	77
Tabla 35	<i>Tarea 5.2 – Aceptar pedido</i>	78
Tabla 36	<i>Tarea 5.3 – Graficar ruta de distribuidor a cliente</i>	78
Tabla 37	<i>Tarea 5.4 – Notificación de llegada al cliente</i>	79
Tabla 38	<i>Tarea 6.1 - Historial de pedidos</i>	79
Tabla 39	<i>Tarea 7.1 - Búsqueda de lugar en el mapa</i>	80
Tabla 40	<i>Tarea 7.2 - Integración de Ubicación Manual</i>	80
Tabla 41	<i>Tarea 8.1 – Edición de Perfil</i>	81
Tabla 42	<i>Cronograma de actividades</i>	81
Tabla 43	<i>Tamaño de botones recomendados por Google</i>	96
Tabla 44	<i>Resumen de pruebas realizadas backend</i>	117
Tabla 45	<i>Resumen de pruebas realizadas frontend</i>	119
Tabla 46	<i>Segundo principio heurístico de usabilidad</i>	122
Tabla 47	<i>Principio Heurístico de la Visibilidad del estado del sistema</i>	123
Tabla 48	<i>Control y libertad del usuario</i>	124
Tabla 49	<i>Quinto principio heurístico, protección de errores</i>	125
Tabla 50	<i>Principio heurístico ocho, Diseño estético minimalista</i>	125
Tabla 51	<i>Tabla de ítems para la colorimetría adecuada de la aplicación móvil</i>	126
Tabla 52	<i>Análisis de las tareas de usabilidad</i>	127
Tabla 53	<i>Características de las herramientas para evaluación de software</i>	129
Tabla 54	<i>Priorización de Funcionalidades</i>	131
Tabla 55	<i>Escala de Likert</i>	140
Tabla 56	<i>Encuesta CSUQ</i>	141
Tabla 57	<i>Tabla de frecuencias de respuestas CSUQ</i>	143
Tabla 58	<i>Análisis de encuesta CSUQ e interpretación de resultados</i>	145
Tabla 59	<i>Pruebas experimentales de los prototipos</i>	148

Índice de figuras

Figura 1 <i>Árbol de problemas</i>	6
Figura 2 <i>Arquitectura de la aplicación móvil</i>	9
Figura 3 <i>Metodología del proyecto</i>	11
Figura 4 <i>Matriz de riesgos</i>	14
Figura 5 <i>Matriz de conceptos de la revisión literaria</i>	19
Figura 6 <i>Segmento de Navstar</i>	28
Figura 7 <i>Ondas de radio bloqueadas</i>	29
Figura 8 <i>Georreferenciación – GPS: Triangulación satelital.</i>	30
Figura 9 <i>Georreferenciación – Telefonía móvil</i>	32
Figura 10 <i>Estructura de un sistema operativo</i>	35
Figura 11 <i>Tipo de aplicaciones móviles</i>	38
Figura 12 <i>Funcionamiento de Web Service</i>	43
Figura 13 <i>Visual Studio Code sigue siendo el IDE preferido entre todos los desarrolladores</i>	47
Figura 14 <i>Herramientas preferidas por los desarrolladores de APIS</i>	50
Figura 15 <i>Características de adobe XD</i>	50
Figura 16 <i>Software estadístico Epidat</i>	52
Figura 17 <i>Versión de software JMeter para pruebas de rendimiento</i>	53
Figura 18 <i>Indicadores de las herramientas seleccionadas para el desarrollo</i>	53
Figura 19 <i>Historia de Usuario 1 - Inicio de sesión</i>	64
Figura 20 <i>Historia de Usuario 2 - Registrarse</i>	64
Figura 21 <i>Historia de Usuario 3 - Actualización de información para distribuidores</i>	65
Figura 22 <i>Historia de Usuario 4 - Recepción de solicitud de pedido</i>	66
Figura 23 <i>Historia de Usuario 5 - Visualizar historial de pedidos</i>	66
Figura 24 <i>Historia de Usuario 6 - Realizar pedido</i>	67
Figura 25 <i>Historia de Usuario 7 – Ingreso Manual de Ubicación</i>	67
Figura 26 <i>Historia de Usuario 8 – Gestión de Información Personal</i>	68
Figura 27 <i>Diagrama de base de datos NoSql</i>	83
Figura 28 <i>Diagrama de proceso para la gestión de pedidos</i>	84

Figura 29 <i>Subproceso de Iniciar Sesión</i>	84
Figura 30 <i>Subproceso Registro</i>	85
Figura 31 <i>Subproceso Pedido</i>	85
Figura 32 <i>Subproceso Determinar Ubicación</i>	86
Figura 33 <i>Arquitectura interna</i>	86
Figura 34 <i>Arquitectura de la aplicación móvil</i>	87
Figura 35 <i>Diagrama de despliegue</i>	88
Figura 36 <i>Plataforma para diseñar prototipos</i>	89
Figura 37 <i>Methodology Behavioral Design Thinking Approach</i>	89
Figura 38 <i>Plantilla utilizada para la implementación del Mapa de Empatía</i>	90
Figura 39 <i>Lluvia de ideas para empezar a prototipar la aplicación móvil mi gas</i>	91
Figura 40 <i>Boceto de inicio de sesión para el usuario</i>	92
Figura 41 <i>Boceto de inicio de sesión para el distribuidor e ingreso de documentos del distribuidor</i>	92
Figura 42 <i>Iconos tomando de la página oficial de Flutter developers</i>	94
Figura 43 <i>Se muestra la fuente de texto recomendada por Google Fonts para Android</i>	95
Figura 44 <i>Prototipo completo del funcionamiento de la aplicación Usuario y Distribuidor</i>	97
Figura 45 <i>Estructura Backend</i>	99
Figura 46 <i>Estructura Frontend</i>	101
Figura 47 <i>Pruebas, pantalla Inicio de Sesión</i>	103
Figura 48 <i>Pruebas, resultado de inicio de sesión</i>	103
Figura 49 <i>Pruebas, registrarse</i>	104
Figura 50 <i>Pruebas, resultado de registrarse</i>	104
Figura 51 <i>Pruebas, registro adicional distribuidor</i>	105
Figura 52 <i>Pruebas, resultado de registro adicional distribuidor</i>	105
Figura 53 <i>Pruebas, realizar pedido</i>	106
Figura 54 <i>Pruebas, resultado realizar pedido</i>	107
Figura 55 <i>Pruebas, resultado de recepción de pedidos de pedidos</i>	107
Figura 56 <i>Pruebas, resultado de aceptar un pedido</i>	108
Figura 57 <i>Pruebas, editar información personal (cliente, distribuidor)</i>	109
Figura 58 <i>Pruebas, resultado de editar información personal (cliente, distribuidor)</i>	110
Figura 59 <i>Pruebas, editar número de teléfono</i>	111
Figura 60 <i>Pruebas, resultado</i>	111

Figura 61 Pruebas, API REST - verificación de código	112
Figura 62 Pruebas, visualizar la ruta en el mapa.....	113
Figura 63 Pruebas, visualizar listado de pedidos por cliente.....	113
Figura 64 Pruebas, visualizar lista de pedidos por distribuidor	114
Figura 65 Pruebas, visualizar nueva ubicación.....	115
Figura 66 Pruebas, visualizar ubicación del distribuidor en la vista del cliente	116
Figura 67 Pruebas, visualizar Información personal (cliente, distribuidor).....	116
Figura 68 Gráfico del resultado pruebas realizas backend.....	118
Figura 69 Gráfico del resultado pruebas realizas backend.....	120
Figura 70 Metodologías para evaluar la usabilidad	122
Figura 71 Escenario 1 – Iniciar sesión	133
Figura 72 Escenario 1 – Solicitar pedido	134
Figura 73 Escenario 1 – Aceptar pedido	134
Figura 74 Escenario 2 – Iniciar sesión	135
Figura 75 Escenario 2 – Solicitar pedido	136
Figura 76 Escenario 2 – Aceptar pedido	136
Figura 77 Escenario 3 – Iniciar sesión.....	137
Figura 78 Escenario 3 – Solicitar pedido	138
Figura 79 Escenario 3 - Aceptar pedido	138
Figura 80 Proyecto del FrontEnd subido en el repositorio de GitHub	166
Figura 8161 Proyecto del Backend subido en el repositorio de GitHub	166
Figura 82 Entrevista a distribuidores en la empresa ENI	167
Figura 83 Uso de la Aplicación Móvil por los Distribuidores.....	167
Figura 84 Uso de la Aplicación Móvil por los Clientes.....	168

Resumen

Este trabajo de grado se enfoca en desarrollar una aplicación móvil que optimice la gestión de la distribución de gas licuado de petróleo en la zona urbana de la ciudad de Ibarra. La aplicación busca abordar la insatisfacción de los usuarios, quienes a menudo deben esperar largos periodos para la entrega debido a la falta de flexibilidad en la distribución. Gracias al avance de las aplicaciones móviles se puede ofrecer una solución alternativa para los desafíos en el servicio de gas licuado de petróleo.

El trabajo consta de cinco capítulos, en la introducción se presenta los antecedentes, planteamiento del problema, objetivos generales, objetivos específicos, alcances y justificación del trabajo. En el primer capítulo, se establece una base teórica que proporciona soluciones y enfoques clave a ser aplicados en la construcción de la aplicación móvil, cimentando así los fundamentos necesarios para su desarrollo.

En el segundo capítulo, recurrimos a la norma ISO/IEC/IEEE 29148-2011 como marco de referencia fundamental para llevar a cabo el proceso de levantamiento de requisitos.

En el tercer capítulo, dentro del marco de trabajo XP, se procedió al desarrollo de la aplicación de manera estructurada. La etapa inicial fue la planificación, respaldada por historias de usuario, seguido del diseño, incluyendo procesos, arquitectura y la interfaz de usuario desarrollada con Adobe XD; junto con la metodología BDT (Behavioral Design Thinking). La fase de codificación se centró en la creación del Backend mediante una API REST basada en Node.js y Express, con la implementación de medidas de seguridad a través de JWT, y la elección de Firebase para el almacenamiento de datos. En el frontend, se utilizó Flutter y se aprovecharon las API de Google, en especial para la integración de mapas.

En el cuarto capítulo, se lleva a cabo la evaluación de la aplicación, haciendo hincapié en las características de usabilidad y eficiencia de desempeño según la norma ISO 25010. Se utilizan herramientas como J-Meter para pruebas de rendimiento y se aplican los principios de usabilidad de Jacob Nielsen.

En el quinto capítulo, se evalúa el impacto de la aplicación móvil mediante el uso de la encuesta CSUQ y pruebas experimentales. En resumen, esta tesis aborda la creación de una aplicación móvil con el objetivo de mejorar la distribución de gas licuado de petróleo en Ibarra, siguiendo un enfoque estructurado, respaldado por normas y metodologías reconocidas, y culminando en una evaluación detallada de su usabilidad y efectividad.

Palabras clave: Software, Geolocation, App Mobil, ISO/IEC/IEEE, usabilidad, Flutter, Android, Firebase, API de Google, GLP. API REST, Node JS. J-Meter, Backend, Frontend

Abstract

This degree work focuses on developing a mobile application that optimizes the management of liquefied petroleum gas distribution in the urban area of the city of Ibarra. The application seeks to address the dissatisfaction of users, who often must wait long periods for delivery due to the lack of flexibility in distribution. Thanks to the advancement of mobile applications, an alternative solution to the challenges in the service of liquefied petroleum gas can be offered.

The work consists of five chapters, the introduction presents the background, problem statement, general objectives, specific objectives, scope, and justification of the work. In the first chapter, we establish a theoretical basis that provides solutions and key approaches to be applied in the construction of the mobile application, thus cementing the necessary foundations for its development.

In the second chapter, we turn to the ISO/IEC/IEEE 29148-2011 standard as a fundamental reference framework to carry out the requirement elicitation process.

In the third chapter, within the XP framework, we proceeded to develop the application in a structured manner. The initial stage was planning, supported by user stories, followed by design, including processes, architecture, and user interface developed with Adobe XD; together with the BDT (Behavioral Design Thinking) methodology. The coding phase focused on the creation of the backend using a REST API based on Node.js and Express, with the implementation of security measures through JWT, and the choice of Firebase for data storage. On the frontend, Flutter was used, and Google APIs were leveraged, especially for map integration.

In the fourth chapter, the evaluation of the application is carried out, emphasizing usability features and performance efficiency according to ISO 25010. Tools such as J-Meter are used for performance testing and Jacob Nielsen's usability principles are applied.

In the fifth chapter, the impact of the mobile application is evaluated using the CSUQ survey and experimental tests. In summary, this thesis addresses the creation of a mobile application with the objective of improving the distribution of liquefied petroleum gas in Ibarra, following a structured approach, supported by recognized standards and methodologies, and culminating in a detailed evaluation of its usability and effectiveness.

Keywords: Software, Geolocation, App Mobil, ISO/IEC/IEEE, Usabilidad, Flutter, Android, Firebase, API de Google, GLP. API REST, Node JS. J-Meter, BackEnd, Frontend.

INTRODUCCIÓN

En la era digital actual, las aplicaciones móviles son protagonistas indiscutibles en la vida cotidiana, se han convertido en una herramienta imprescindible para el día a día de millones de personas. En este contexto, en esta tesis se propuso desarrollar una aplicación móvil abordando los aspectos técnicos clave, como las plataformas y lenguajes de programación utilizados, así como los aspectos de diseño y experiencia de usuario (UI/UX) que desempeñan un papel crucial en la aceptación y éxito de una aplicación. Además, consideraremos los estándares de calidad, como las normas ISO 25010 e ISO/IEC/IEEE 29148-2011, que son fundamentales para garantizar la eficiencia y la satisfacción de los usuarios al llevar a cabo una serie de tareas de manera eficiente y efectiva. Para ello, se partirá de un análisis exhaustivo de las necesidades y expectativas de los usuarios, así como de las tendencias y oportunidades de la situación actual.

A lo largo de este documento, se explorarán los desafíos y oportunidades asociados con el desarrollo de una aplicación móvil empresarial, desde la concepción de la idea hasta la implementación práctica. Además, se presentarán ejemplos de casos de estudio y resultados de pruebas piloto que demuestran la eficacia de la aplicación en entornos empresariales reales.

En definitiva, esta tesis representa una oportunidad para explorar en profundidad el mundo del desarrollo de aplicaciones móviles aplicando los conocimientos que se adquieren en las aulas de la Universidad y contribuir a mejorar situaciones de la vida diaria.

Planteamiento del problema

El gas licuado de petróleo (GLP) se ha comercializado en Ecuador durante más de 40 años a través de cilindros de uso doméstico. La distribución del GLP cumple con los estándares técnicos establecidos por la Agencia de Regulación y Control de Energía y Recursos Naturales no Renovables (ARCERNR, 2016), en colaboración con la compañía internacional independiente del sector petrolero (ENI). La sucursal de ENI se encuentra ubicada en la Panamericana Norte Km 5.5, La Dolorosa de Priorato, Ibarra, Imbabura, Ecuador.

El GLP en cilindros tiene una amplia presencia en Latinoamérica, con una participación que oscila entre el 8% y el 85% (García N, 2021). En Ecuador, que es uno de los países de América del Sur que más subsidia los combustibles, el GLP es la principal fuente de energía para las cocinas domésticas debido a los crecientes precios internacionales del petróleo y sus derivados (A.R.C.H, 2019). Sin embargo, los métodos de distribución actuales no satisfacen las necesidades de los

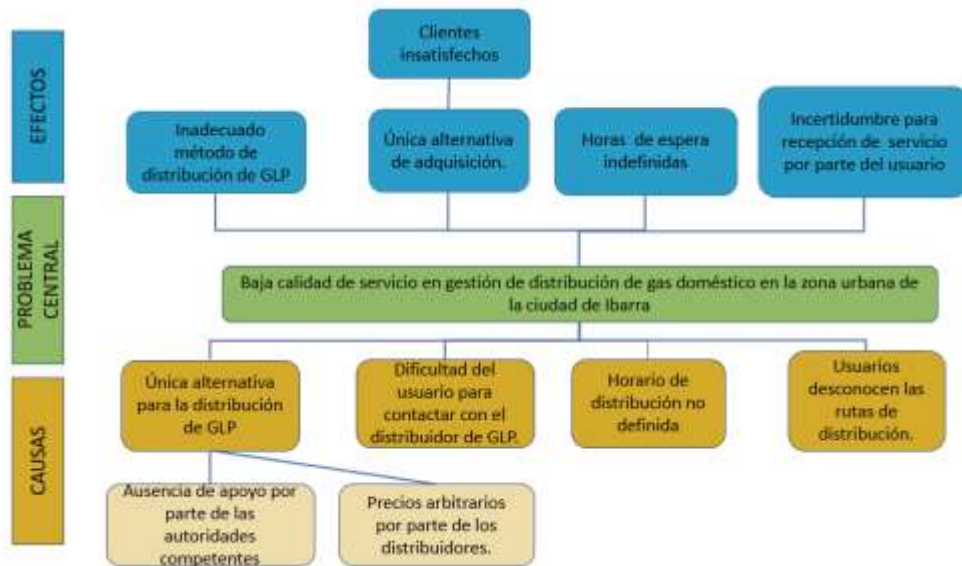
usuarios ni de los distribuidores, lo que aumenta los costos de operación. Por lo tanto, es necesario mejorar la distribución mediante una alternativa más eficaz y eficiente.

Según varios estudios (Acurio Javier, 2017; Eras Frank, 2020; Guerrón Álvaro, 2018), la distribución de gas en camiones autorizados produce diversos tipos de contaminación ambiental, incluyendo contaminación sonora, emisiones de dióxido de carbono y congestión vehicular. Esto se debe en gran parte a que los camiones no tienen horarios ni rutas fijas, lo que los lleva a recorrer las calles de forma ineficiente. Además, muchos usuarios no se encuentran en sus hogares durante las horas laborales, lo que dificulta la adquisición de cilindros de gas. Por lo tanto, es evidente la necesidad de explorar alternativas más eficientes y sostenibles para la distribución de gas.

De acuerdo con un auscultamiento informal con los usuarios, se resaltan las siguientes variables: inadecuada atención al cliente, inconformidades de precios arbitrarios por sectores, dificultad al contactar con el distribuidor y desconocimiento de horario de atención. Esto afecta al distribuidor económicamente en sus gastos operativos.

Figura 1

Árbol de problemas



Nota. Se detalla las causas y los efectos identificados en la zona urbana de la ciudad de Ibarra en el 2023.

Fuente: Elaboración propia.

Objetivos

Objetivo General

Mejorar la calidad de servicio en la distribución de gas GLP, integrando API REST y Flutter mediante una aplicación móvil, incidencia en la zona urbana de la ciudad de Ibarra.

Objetivos Específicos

- Realizar un marco teórico sobre tecnologías utilizadas para gestionar el proceso de distribución de GLP.
- Definir los requerimientos para la aplicación móvil, bajo la norma ISO/IEC/IEEE 29148-2011.
- Construir la aplicación móvil con XP como marco de trabajo, con base a los principios de usabilidad y Apis de Google para la geolocalización.
- Analizar la aplicación móvil utilizando indicadores de calidad para la usabilidad y eficiencia de desempeño con base a la norma ISO 25010.
- Validar los resultados de la investigación.

Alcance

El proyecto consiste en mejorar la calidad de servicio en distribución de GLP, conformada por cinco parroquias: Caranqui, San Francisco, Sagrario, Alpachaca y Priorato; mediante una aplicación móvil que permita al usuario realizar pedidos a domicilio a los camiones distribuidores. Se analizará los resultados y metodologías utilizadas en investigaciones realizadas a nivel local, nacional e internacional, que han abordado temáticas similares a las propuestas en este trabajo. Para justificar la necesidad de contar con una alternativa tecnológica en la distribución de GLP a nivel doméstico, se aplicará encuestas a: distribuidores, usuarios y autoridades competentes. Además de servir de apoyo para la identificación de necesidades, servirá para el levantamiento de requisitos que se aplicará bajo la norma IEEE 29148. Esta norma especifica los procesos requeridos implementados en las actividades de ingeniería que dan como resultado requisitos para sistemas y productos de software (incluidos los servicios) a lo largo del ciclo de vida (ISO/IEC/IEEE 29148:2011). Según Ávila (2022), la licitación de requisitos ayudará a conocer mejor las necesidades del desarrollador, de la misma manera servirá de guía para solventar necesidades de ámbito político del sector donde se va a desplegar la aplicación móvil (p, 3).

Frontend Aplicación móvil (Flutter)

En el sitio web de statista, en lo relacionado a Flutter se destaca la siguiente aseveración:

Flutter es el marco móvil multiplataforma más popular utilizado por desarrolladores globales, según una encuesta de desarrolladores de 2021, el 42 % de los desarrolladores de software utilizaron Flutter. Esto quiere decir que un tercio de los desarrolladores móviles utilizan tecnologías o marcos multiplataforma; el resto de los desarrolladores móviles utilizan herramientas nativas (Sujay, 2022).

Con base a la información obtenida, la aplicación será desarrollada con el framework SDK Flutter desarrollado por Google para crear aplicaciones móviles. La arquitectura será MVC (Modelo-Vista-Controlador) monolítica. Haciendo uso del lenguaje Dart, los widgets y las propiedades de lenguaje compilado que inciden en el desarrollo de la aplicación nativa (Android).

BackEnd como web service (API REST)

Se realizará desarrollando un Web Service de API REST y el API de Google para geolocalización con Express (Node.js). Las peticiones y respuestas dependerán del protocolo de HTTPS y JSON; para la parte lógica de monitorización de los distribuidores, se consumirá el Api de Google en donde el dispositivo debe estar conectado a internet y emitir coordenadas geográficas (Latitud y longitud), estos datos se almacenarán en la base de datos NoSQL Firebase.

Firebase como gestor de datos

La arquitectura de almacenamiento para el cliente, distribuidor y coordenadas de geolocalización, se usará la plataforma de Firebase. Firebase es una plataforma BaaS (Backend as a Service), desarrollada por Google que facilita el desarrollo de aplicaciones, proporcionando un servidor Backend para las aplicaciones (Chicaiza, 2020). Esto ayudara a tener una monitorización de geolocalización en tiempo real de clientes y distribuidores.

Indicadores de calidad

Para evaluar los indicadores de calidad se utilizará la Norma ISO 25010 que permite analizar: comportamiento temporal, utilización de recurso y capacidad que tiene la aplicación móvil cuando se ejecuta en un dispositivo Android. Se evaluará los principios de experiencia de usuario (UX/UI): Reconocibilidad de la adecuación, Aprendizabilidad, Operabilidad, Protección frente a errores del usuario, Estética y Accesibilidad (Colorimetría).

Módulos

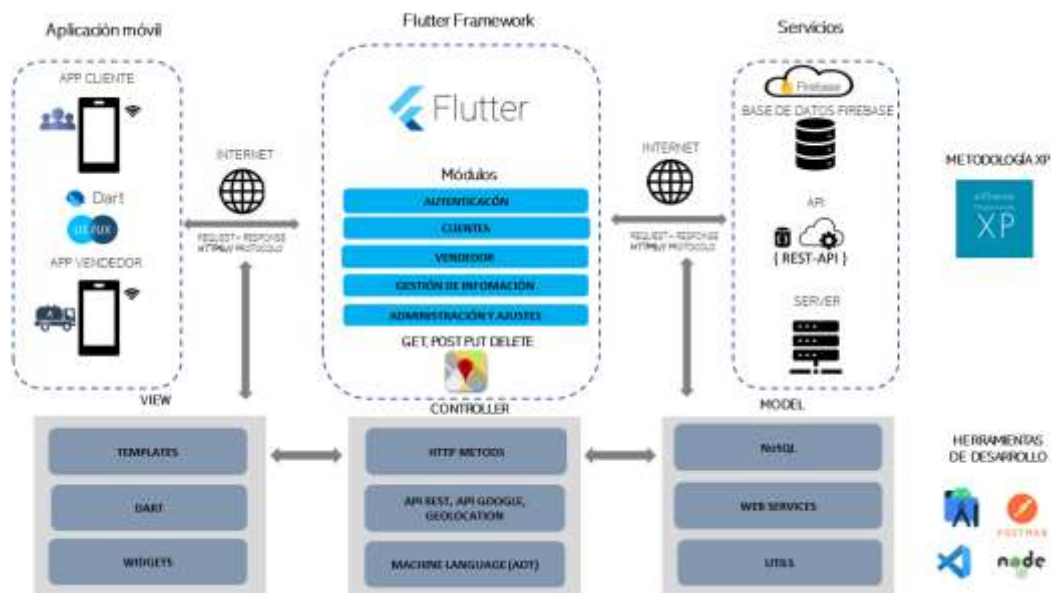
La aplicación contará con 5 módulos, en donde el módulo de autenticación se podrá gestionar el acceso a la plataforma, crear una cuenta única que posteriormente le permitirá elegir un rol de distribuidor o usuario. El módulo de vendedor solicitará datos básicos de conductor y una vez completados el registro recibirá notificaciones nuevas de ubicación del cliente que realiza la petición de servicio. El módulo de clientes permitirá que el usuario solicite el servicio. El módulo gestión de la información guardará los datos históricos de clientes y distribuidores. El módulo de administración y gestión permite configurar las preferencias de la aplicación y cambiar los roles de los usuarios asignados.

Validación de pruebas

Para la validación se implementará pruebas con usuarios y distribuidores, se comprobará el funcionamiento de la ampliación móvil, se aplicará una encuesta de aceptabilidad de la aplicación y comprobará la minimización de los efectos referidos en el problema central.

Figura 2

Arquitectura de la aplicación móvil



Nota. Se describe los módulos y el flujo de la aplicación entre el BackEnd y Frontend.

Fuente: Elaboración propia.

Metodología

Para cumplir el primer objetivo, se aplicará investigación de campo mediante recopilación de información utilizando el instrumento de encuestas personalizadas dirigidas hacia clientes y distribuidores, se aplicará la metodología SLR, para realizar la revisión sistemática de bibliografía referente al tema.

Para cumplir el segundo objetivo, se realizará levantamiento de requisitos utilizando la norma ISO/IEC/IEEE 29148-2011, estándar que guía el proceso del ciclo de vida del software, cuenta con definiciones del contenido y recomendaciones para la documentación, además ayuda dentro de la planificación del desarrollo de software mediante peticiones del usuario (Quiroz Carlolina, 2022).

Para cumplir el tercer objetivo, se investigará temas referentes a: framework, Api Rest y principios heurísticos de usabilidad que serán utilizados en el desarrollo del aplicativo móvil. Seguido se procederá al desarrollo de la aplicación aplicando el framework Flutter, Api de Google Maps y para diseño la usabilidad de la ISO/IEC 25010, todo el proceso de desarrollo se trabajará bajo el marco de trabajo Extreme Programming XP.

Para cumplir el cuarto objetivo, se realizará la validación del resultado del aplicativo móvil, mediante modelo de calidad del producto definido por la ISO/IEC 25010, considerando dos características: Usabilidad y Eficiencia de desempeño. La usabilidad se refiere a la aptitud del software para ser comprensible, adquirir habilidad de uso y generar interés en el usuario, en situaciones específicas. Incluye subcaracterísticas que se describen a continuación:

- **Inteligibilidad:** permite al usuario reconocer si un producto o sistema es apropiado para sus necesidades.
- **Aprendizaje:** el producto o sistema permite que el usuario aprenda a utilizarlo con efectividad, eficiencia y satisfacción en un contexto de uso específico
- **Operabilidad:** permite al usuario operar y controlar un producto/sistema con facilidad:
- **Protección frente a errores:** el sistema protege a los usuarios de cometer errores, los cuales pueden no ser rectificables.
- **Estética:** la interfaz del producto o sistema permite una interacción agradable y satisfactoria para el usuario.
- **Accesibilidad:** La colorimetría es un aspecto importante en el diseño accesible, ya que el contraste y la legibilidad del texto dependen en gran medida de los colores utilizados.

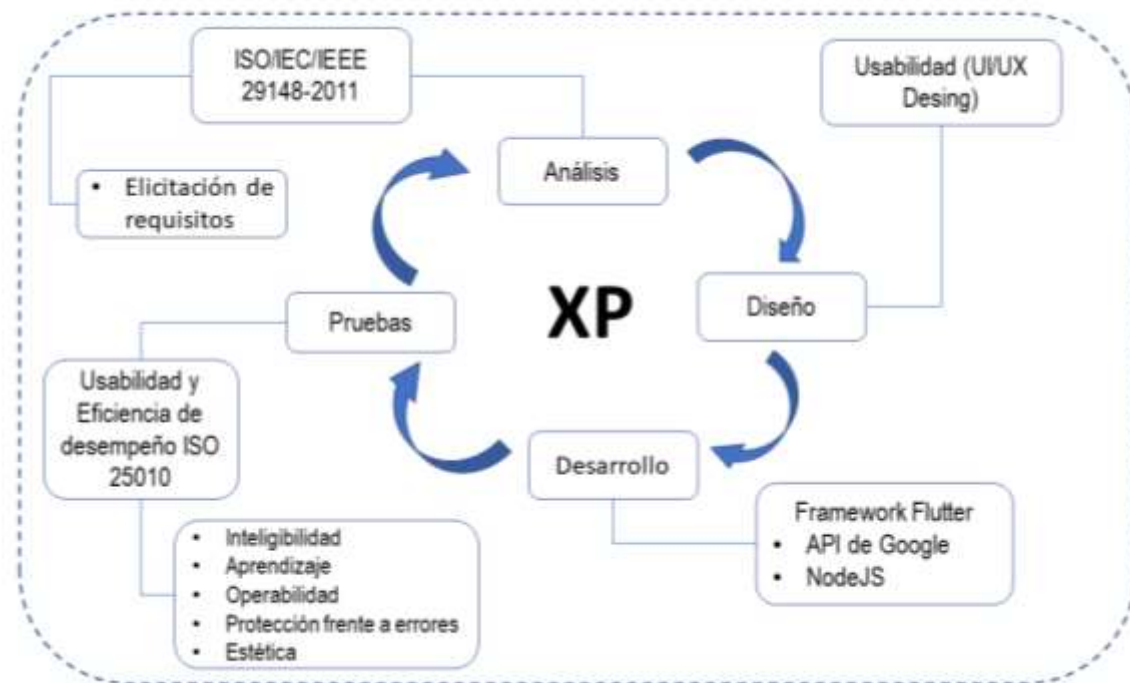
La eficiencia de desempeño representa la cantidad de recursos utilizados bajo determinadas condiciones, cuenta con subcaracterísticas que se mencionan a continuación:

- **Comportamiento temporal:** está constituida por los tiempos de respuesta y procesamiento de una aplicación en ejecución.
- **Capacidad:** se refiere a la capacidad de respuesta del software cuando funciona en línea.

Para el cumplimiento del quinto objetivo, se realizará simulaciones de uso de la aplicación móvil con usuarios y distribuidores definidos aleatoriamente en un lapso de prueba que permita obtener resultados que justifiquen o no el impacto en la comunidad objeto de estudio.

Figura 3

Metodología del proyecto



Nota. Se describe las metodologías, normas a utilizar en cada etapa de desarrollo.

Fuente: Elaboración propia.

Justificación

La forma tradicional de distribución de gas doméstico GLP se ha mantenido en el tiempo sin presentarse alternativas válidas que utilicen nuevas tecnologías, los usuarios se han allanado a

este método, sin expresar las inconformidades generadas a partir de este método de distribución. Por esa razón, se propone la investigación para mejorar la calidad del servicio (distribución de cilindros de gas), sustentado en dos objetivos de desarrollo sostenible (ODS):

En el primer caso tenemos al objetivo nueve: Industria, Innovación e Infraestructura, donde la innovación y el progreso tecnológico son claves para descubrir soluciones duraderas para los desafíos económicos y medioambientales; entre las nueve metas del objetivo, el proyecto se centrará en el apartado N°9.4, el cual menciona la modernización de infraestructuras para que sean sostenibles apoyándose en la adopción de nuevas tecnologías (Naciones Unidas, 2019).

En el segundo caso se refiere al objetivo de desarrollo sostenible número once: Ciudades y Comunidades Sostenibles, donde se pretende lograr que las ciudades sean más inclusivas, seguras, resilientes y sostenibles. De las once metas del objetivo, el proyecto se centrará en el apartado N°11.1, el cual indica el aseguramiento al acceso de todas las personas a servicios básicos adecuados, seguros, asequibles y de esa manera asegurar comunidades o ciudades sostenibles (Naciones Unidas, 2019).

Además, se pretende demostrar desde los puntos de vista social y tecnológico las razones que motivaron a la investigación, a continuación, se describen cada una de las justificaciones:

Justificación social: El gas doméstico al ser considerado un producto de primera necesidad, su demanda es alta; con el aplicativo móvil, los habitantes de la ciudad de Ibarra tendrán facilidad y eficiencia en la solicitud del servicio, la entrega del gas doméstico se realizará en menor tiempo por parte del distribuidor.

Justificación Tecnológica: La tecnología apoya en todas las operaciones de negocios y resulta muy importante para el crecimiento de cualquier empresa. Sin embargo, en muchos casos, la tecnología no es explotada ni valorada adecuadamente en el entorno empresarial, con lo que se pierden oportunidades de obtener los beneficios derivados de su uso (Ramón Hernández, 2016). Se considera la implementación de una aplicación móvil que mejorará los procesos de distribución de gas doméstico y aprovechar las facilidades que nos proporción la tecnología.

Riesgos

Tabla 1

Riesgos y estrategias de mitigación

Nro.	Riesgo	Estrategia de Mitigación
------	--------	--------------------------

R1	El dispositivo móvil no es compatible con la versión de aplicación móvil.	Establecer en Google Play desde que versión estará disponible la aplicación.
R2	Inaccesibilidad al Internet.	Contar con un plan de datos móviles que le permita al usuario poder conectarse a internet.
R3	La ubicación marcada no es precisa, y los repartidores tienden a realizar llamadas para saber la ubicación exacta del cliente.	Solicitar al cliente información detallada (fotos, descripción) de su ubicación.
R4	Requerimientos mal levantados para la aplicación móvil.	Indagar información más detallada sobre el proceso de distribución y definir alcances.
R5	Gastos de desarrollo e implementación que sobrepasen el presupuesto establecido.	Asignar un presupuesto extra para solventar gastos imprevistos.
R6	Resistencia al cambio por parte de los usuarios.	Socializar el tema con la ciudadanía de la ciudad de Ibarra para llegar a acuerdos y a futuro difundir el proyecto tecnológico como una alternativa de distribución.

Nota. Se detalla una tabla riesgos con la respectiva mitigación.

Fuente: Elaboración propia.

Figura 4

Matriz de riesgos

PROBABILIDAD	3	Alta	15	30	60
	2	Medio	10	20	40
	1	Baja	5	10	20
			Bajo	Medio	Alto
			5	10	15
			IMPACTO		
			<div style="display: flex; justify-content: space-around;"> <div style="background-color: yellow; padding: 5px;">R6 Resistencia al cambio</div> <div style="background-color: red; padding: 5px;">R4 Requerimientos deficientes</div> </div>		
			<div style="display: flex; justify-content: space-around;"> <div style="background-color: green; padding: 5px;">R5 Exceso de costos</div> <div style="background-color: yellow; padding: 5px;">R2 Inestabilidad al Internet</div> <div style="background-color: red; padding: 5px;">R1 Incompatibilidad dispositivo-aplicación</div> </div>		
			<div style="display: flex; justify-content: space-around;"> <div style="background-color: green; padding: 5px;">R5 Exceso de costos</div> <div style="background-color: yellow; padding: 5px;">R3 Ubicación imprecisa</div> </div>		

Nota. Se indica la matriz de riesgos clasificado por su valor de impacto.

Fuente: Elaboración propia adaptado.

CAPÍTULO I

1.1 Marco teórico

Este capítulo, centrado en la base teórica del proyecto, se dedica a exponer las metodologías empleadas para la recopilación de información y comprender las necesidades que influyen en el desarrollo del proyecto. En el contexto del desarrollo de la aplicación móvil, estas metodologías han sido fundamentales para la selección de las herramientas tecnológicas, estándares, marco de trabajo y pruebas que contribuirán a mejorar la calidad del software.

El marco teórico se desglosa en seis capítulos. El capítulo 1.1 se enfoca en la revisión sistemática de la literatura y en cómo buscar información de manera eficiente. En el capítulo 1.2, nos concentramos en la identificación de procesos y la determinación de la situación actual del lugar donde se desarrollará el proyecto. El capítulo 1.3 detalla conceptos y definiciones relacionados con las tecnologías que intervienen en el funcionamiento de la geolocalización en dispositivos móviles. El capítulo 1.4 se centra en las técnicas y metodologías de la recopilación de requisitos, en consonancia con la norma ISO-IEC-IEEE 29148-2011. En el capítulo 1.5, abordamos los sistemas operativos móviles para diversas plataformas, y finalmente, en el capítulo 1.6, se proporciona una descripción detallada de las herramientas utilizadas en el desarrollo de la aplicación.

1.2 Revisión de literatura SLR

Las revisiones sistemáticas de literatura han adquirido gran relevancia en la investigación de ingeniería de software. En la aplicación de esta metodología, aspectos críticos se relacionan con asegurar un diseño y ejecución apropiados, además de una estrategia de búsqueda efectiva. Para establecer los fundamentos teóricos de la investigación, se siguió un enfoque sistemático para examinar y evaluar la literatura existente relacionada con el tema de estudio. Este proceso condujo a la obtención de información relevante. Según (Miswar, Suhardi, & Kurniawan, 2018) propone seguir el siguiente modelo.

- a) Preguntas de investigación
- b) Búsqueda de documentos
- c) Selección de artículos
- d) Extracción de datos relevantes.

A continuación, se define la pregunta de investigación, el alcance de la revisión, los criterios de inclusión y exclusión, y finalmente la cadena de búsqueda.

1.2.1 Determinar las Preguntas de Investigación.

En la siguiente tabla se definen las preguntas de investigación y la motivación que tiene cada una para entender centrar la búsqueda bibliográfica.

Tabla 2

Preguntas de investigación de la revisión literaria.

Número	Preguntas de investigación	Motivación
RQ1	¿Cuál es el proceso actual de distribución de cilindros de gas doméstico en la zona urbana de la ciudad de Ibarra?	Conocer la situación actual del proceso desde la planta ENI y la distribución del cilindro de gas doméstico al usuario final.
RQ2	¿Cuáles son las necesidades y preferencias de los usuarios que deben abordarse en el diseño de la aplicación móvil?	Identificar las opiniones de los usuario y distribuidores para replantear los requerimientos.
RQ3	¿Cuál es el proceso óptimo para la recopilación de requisitos y la definición de funcionalidades en el desarrollo de una aplicación móvil?	Identificar las normas y procedimientos de los que se deban abordar antes de realizar el desarrollo.
RQ4	¿Cómo se pueden integrar características de geolocalización de manera efectiva en la aplicación móvil para mejorar su utilidad?	Conocer sobre el manejo de Geolocalización en la nube y bases de datos NoSQL en tiempo real
RQ5	¿Qué lenguajes de programación, herramientas y metodologías elegir?	Identificar los leguajes de programación más populares y analizar el desempeño en aplicaciones móviles.
RQ6	¿Cuál es el impacto de la elección de sistemas operativos móviles (iOS, Android, etc.) en la distribución y alcance de la aplicación?	Realizar una investigación para determinar el grupo específico

RQ7	¿Cuáles son los métodos más efectivos para la prueba y la validación de la aplicación móvil?	de usuarios a los que se dirigirá el proyecto. Conocer las metodologías normas, estándares y herramientas se usan actualmente para validar un software
------------	--	---

Nota. Se plantea las preguntas de investigación y la motivación que aportan al desarrollo de proyecto.

Fuente: Elaboración propia.

1.2.2 Búsqueda de Documentos.

Esta metodología se enfoca en realizar investigaciones a través de motores de búsqueda automatizadas proporcionados por las bibliotecas digitales. El proceso se inicia con la selección de las bases de datos bibliográficas como: IEEE Xplore, Google Scholar, Microsoft Academic, SpringerLink, repositorios de en países latinoamericanos y repositorios digitales universitarios del Ecuador, después se determinó las cadenas de búsqueda “ (Geolocation on mobile devices) OR (aplicación móvil OR Development app mobile) OR (Api Google Geolocation OR firebase en dispositivos móviles) OR (Usability principles in mobile applications o Usabilidad para aplicaciones móviles) OR (Web Services or Servicios web or Api REST) OR (Data base NoSQL OR Bases de datos no relacionales) AND (Aplicaciones móviles de distribución de gas doméstico o aplicaciones móviles con principios de usabilidad)” en el que se obtiene un listado de documentos recuperados.

En la Tabla 3 se presentan las cadenas de búsqueda empleadas en cada una de las bibliotecas digitales.

Tabla 3*Cadenas de búsqueda utilizado en librerías digitales*

IEEE Xplore	Google Scholar	SpringerLink	Repositorios digitales universitarios
Geolocation on mobile devices	Api Google Geolocation	Web Services or Servicios web or Api REST	Aplicaciones móviles de distribución de gas doméstico
Aplicación móvil o Development Mobile App	Firebase en dispositivos móviles	Agile development framework	Aplicaciones móviles con principios de usabilidad
ISO standards for software development	Development languages and associated methodologies	Gathering requirements or user stories	Aplicaciones móviles con Api Rest
ISO standards for software testing	Data base NoSQL OR Bases de datos no relacionales	Best practices for software security with international standards	Aplicaciones móviles con bases de datos no relacionales

Nota. Fuente: Elaboración propia.

1.2.3 Selección de Artículos - Criterios de Inclusión y Exclusión.

Para filtrar los estudios se aplicaron los siguientes criterios de inclusión/exclusión:

- Estudios que se centran en la distribución de gas doméstico y la tecnología móvil en el contexto de países en desarrollo.
- Se incluyeron estudios en 2 idiomas: inglés y español.
- Estudios que examinan la experiencia y la satisfacción del usuario con las aplicaciones móviles.
- Estudios que se centran en los procesos de levantamiento de requisitos.

- Estudios relacionados con trabajo con metodologías ágiles.
- Estudios relacionados con aplicación de normas y estándares ISO/IEEE
- Investigaciones que se enfocan en aplicaciones móviles con Apis y bases de datos no relacionales.
- Estudios centrados en aplicaciones móviles con Geolocalización.

1.2.4 Extracción de Datos Relevantes.

Se elaboró una matriz de conceptos para identificar conceptos relevantes de la investigación en el que se determinó herramientas importantes para el uso de la tecnología con geolocalización, así como también la importancia de realizar aplicaciones para dispositivos móviles como se puede visualizar en la figura 5.

Figura 5

Matriz de conceptos de la revisión literaria

Artículos		Conceptos							
Código	Geolocation	Mobile devices	Usability principles in mobile	Web Services or Servicios web or	Data base NoSQL ORelacionales	App Móvil para distribución de gas doméstico	firebase en dispositivos móviles	Software quality standard and standards	Agile methodologies
A1	•				•				
A2	•	•	•		•		•	•	•
A3	•	•	•	•	•				•
A4	•	•			•	•	•		
A5						•		•	
A6		•				•			
A7				•			•	•	•

Nota. Se detalla los indicadores de artículos y conceptos utilizados.

Fuente: Elaboración propia.

1.3 Situación actual

En Ecuador se presenta un aumento significativo en el uso de teléfonos inteligentes en los últimos años. Según un informe realizado por la página DataReportal en 2022, el 88.4% de la población ecuatoriana tienen al menos un teléfono móvil y el 75.6% de la población tiene conexión a internet a través de su dispositivo móvil. Además, según el Instituto Nacional de Estadística y Censos (INEC), las aplicaciones de redes sociales son las más utilizadas, seguidas de las aplicaciones de mensajería instantánea, banca en línea, compras en línea, entre otras de entretenimiento.

En cuanto a las preferencias de sistema operativo, Android es el más utilizado en Ecuador, con una participación de mercado del 86.75%, seguido por iOS con un 12.93%. En conclusión, el uso de dispositivos móviles está en constante crecimiento y se ha convertido en una herramienta importante para acceder a servicios y contenido en línea. Esto representa una oportunidad para el desarrollo de aplicaciones móviles y para el comercio electrónico en el país.

1.3.1 Situación Actual En La Zona Urbana De La Ciudad De Ibarra.

Existen aplicaciones móviles operando con servicios de geolocalización como inDriver (servicio de taxi), Uber Eats (pedidos de comida) y Pedidos Ya (multiservicio), pero no se conoce sobre una aplicación indicada para solicitar un cilindro de gas al hogar. En la ciudad de Ibarra cuando el usuario necesita gas doméstico, espera al camión distribuidor que circula con música por las calles ofreciendo sus servicios, consiste en canjear el cilindro de gas vacío por uno lleno y es la forma más común de distribución. Algunos distribuidores comparten el número de celular para recibir mensajes o llamadas de pedidos de los usuarios. Para obtener datos e identificar las necesidades; se aplica una encuesta para los usuarios y una entrevista para los distribuidores, la finalidad es obtener variables importantes e identificar los procesos que se llevan a cabo para el suministro de gas doméstico.

1.3.2 Cuestionario e Interpretación de Datos de la Encuesta.

Para realizar la encuesta se tomó una muestra poblacional de la zona urbana del cantón Ibarra con un número total de 170 549 habitantes. (Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra, 2022). La muestra poblacional aleatoria simple se calcula en el software de Epidat; es un software libre desarrollado por Consellería de Sanidade, Xunta de Galicia, España; Organización Panamericana de la Salud (OPS-OMS); Universidad CES, Colombia (Consellería

de Sanidade et al., 2016), dando como resultado 384 individuos a encuestar, con un nivel de confianza de 95% y el margen de error de 5%. Para mayor claridad de los datos Se solicita contestar la encuesta por un miembro familiar que está encargada de los quehaceres domésticos.

A continuación, en la Tabla 4, se muestra el cuestionario utilizado para recopilar las opiniones de los usuarios, junto con una tabulación de cada una de las preguntas.

Tabla 4

Cuestionario sobre nivel de conocimiento de aplicaciones móviles enfocado a los usuarios

Nro.	Pregunta												
1	<p>Genero</p> <p>El 67% de los encargados de realizar los que haceres domésticos son mujeres y el 32% realizan los hombres.</p> <div data-bbox="721 800 998 1031" data-label="Figure"> <table border="1"> <caption>Gender Distribution</caption> <thead> <tr> <th>Gender</th> <th>Percentage</th> </tr> </thead> <tbody> <tr> <td>Mujeres</td> <td>67%</td> </tr> <tr> <td>Hombres</td> <td>32%</td> </tr> </tbody> </table> </div>	Gender	Percentage	Mujeres	67%	Hombres	32%						
Gender	Percentage												
Mujeres	67%												
Hombres	32%												
2	<p>¿Cuál es el nivel de conocimiento que tiene sobre el manejo de una aplicación móvil?</p> <p>El 7.8 % no conocen nada, el 7.8% conocen, pero no usan, el 27% da una opinión neutra, el 36.9% conoce y usa y el 20,4% se desenvuelve excelente en el manejo de aplicaciones móviles, se puede notar que la mayoría tiene conocimiento a la pregunta planteada.</p> <div data-bbox="318 1346 1323 1493" data-label="Figure"> <table border="1"> <caption>Nivel de conocimiento</caption> <thead> <tr> <th>Nivel de conocimiento</th> <th>Porcentaje</th> </tr> </thead> <tbody> <tr> <td>Nada</td> <td>7.8%</td> </tr> <tr> <td>Conoce, pero no usa</td> <td>7.8%</td> </tr> <tr> <td>Neutro</td> <td>27%</td> </tr> <tr> <td>Conoce y usa</td> <td>36.9%</td> </tr> <tr> <td>Conoce mucho</td> <td>20.4%</td> </tr> </tbody> </table> </div>	Nivel de conocimiento	Porcentaje	Nada	7.8%	Conoce, pero no usa	7.8%	Neutro	27%	Conoce y usa	36.9%	Conoce mucho	20.4%
Nivel de conocimiento	Porcentaje												
Nada	7.8%												
Conoce, pero no usa	7.8%												
Neutro	27%												
Conoce y usa	36.9%												
Conoce mucho	20.4%												
3	<p>¿Conoce alguna aplicación móvil para pedir gas doméstico a domicilio?</p> <p>El 68% desconoce sobre alguna aplicación exclusiva para la distribución de cilindros de gas, el 13.6% conoce, pero no usa, el 15.5 % da una opinión neutra, el 1% conoce y usa, el 1% se desenvuelve de manera excelente con la aplicación; el 2% mencionado se asume que son ciudadanos de la provincia de Loja o Carchi, en donde existe una</p>												

aplicación para controlar la fuga del hidrocarburo a los países como Colombia y Perú (El Comercio, 2020).



4 **Si hubiera una aplicación móvil exclusiva para adquirir el cilindro de gas doméstico**

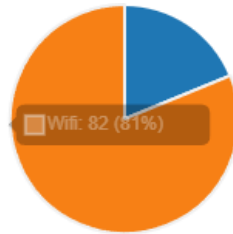
¿La utilizaría?

Los resultados indican que la mayoría de persona optarían por esta nueva alternativa de distribución con un porcentaje de aceptación de 67.3%, mientras que el 24% les es indiferente, el 1% está en desacuerdo y el 6.9% no utilizaría de ninguna manera.



5 **¿Como se conecta al internet en su celular cuando está en el hogar?**

El 19% se conecta mediante datos móviles y el 81% se conecta mediante una red WIFI en el hogar.



6 **¿Cuál es el sistema operativo de su dispositivo móvil?**

La mayoría de los ciudadanos de Ibarra usan el sistema operativo Android en un 68%, el 13% utiliza iOS, el 5% utiliza otros y el 14% desconoce del sistema operativo que tiene el dispositivo móvil.



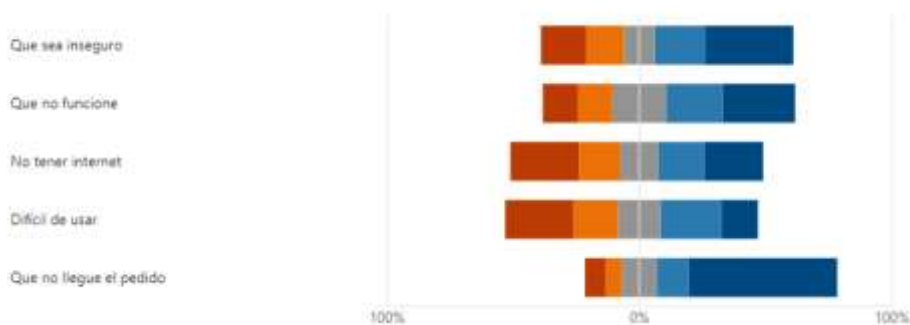
7 **En caso de solicitar gas doméstico a través de una aplicación móvil, ¿Cómo le gustaría que sea el medio de pago?**

La preferencia de los usuarios al momento de pagar por el cilindro de gas prefiere efectivo con el 64% de aceptabilidad y el 36% prefiere pagar con transferencia bancaria.



8 **¿Cuáles serían sus temores al usar una aplicación móvil para la adquirir cilindros de gas doméstico?**

Entre los temores de los ciudadanos se puede notar que les dan Mayor importancia a los ítems **“Que sea inseguro”**, **“Que no funcione”** y **“Que no llegue el pedido”**, mientras que los ítems **“No tener internet”** y **“Difícil de usar”** le consideran de menor importancia”



Nota. Interpretación de resultados realizado a los usuarios.

Fuente: Elaboración propia.

1.3.3 Entrevista e interpretación de Datos.

En la Tabla 5 se detallan las preguntas específicas que se dirigieron exclusivamente a los distribuidores. Esta entrevista resultó fundamental para obtener datos valiosos y comprender en detalle los procesos de distribución de gas doméstico en la localidad. El número de entrevistados fue siete distribuidores.

Tabla 5

Preguntas para identificar el proceso de distribución de gas domestico

Nro.	Preguntas para los distribuidores
1	<p>¿Conoce alguna aplicación móvil para vender gas?</p> <p>Con respecto a la pregunta uno, tres de cuatro entrevistados menciona no conocer o usar alguna aplicación móvil para vender gas en la zona urbana de la ciudad de Ibarra.</p>
2	<p>¿Cuántos cilindros de gas vende en promedio al día?</p> <p>Todos los entrevistados en promedio mencionan que se vende de 40-60 en temporada baja, en temporada alta se menciona que se vende de 80-100. Esto hace referencia al inicio de clases en colegios, escuelas y universidades.</p>
3	<p>¿Qué requisitos se necesita para ser parte de la empresa ENI?</p> <p>Todos los entrevistados mencionan lo siguiente: Se debe tener bodega de almacenamiento aprobada por ENI, camión, cilindros, permisos del municipio, permiso de bomberos, certificado otorgado por la ley de hidrocarburo, facturación electrónica, impuestos de hidrocarburos; 400 por bodega y 200 por vehículo.</p>
4	<p>¿Existe límite de adquisición de gas en la planta envasadora?</p> <p>Todos los entrevistados mencionan que el límite de compra depende de la capacidad del camión distribuidor y los cupos que tenga otorgados 180-200 cilindros, depende de la capacidad del camión. Si no vende, el cupo es disminuido.</p>
5	<p>¿Tiene rutas asignadas para cada conductor o puede vender el gas en cualquier ubicación?</p> <p>Todos los entrevistados menciona que no existen rutas asignadas, los distribuidores circulan por toda la zona urbana de la ciudad de Ibarra.</p>

¿Existe diferencia de precios entre los colores de los cilindros de gas?

6 Todos los entrevistados mencionan que el precio no varía en base al color del cilindro, los precios establecidos son a un dólar \$1,60 centavos en la bodega; fuera de la bodega se considera como servicio a domicilio y el precio establecido es de \$3 dólares por cilindro. Solo se autoriza vender dos cilindros por familia.

¿Cuál es el horario de distribución permitido?

7 Todos los entrevistados mencionan que el horario de distribución es de seis de la mañana hasta la seis de la tarde, pasado las seis de la tarde no es permitido circular el camión haciendo sonar el alto parlante por la ley de hidrocarburo y el municipio de Ibarra; algunos distribuidores venden sin sonar el alto parlante, en la bodega de ENI los camiones solo pueden abastecerse hasta las tres de la tarde.

¿Alguna vez sufrió algún incidente que haya puesto en peligro su vida en horas laborables?

8 Por testimonios de los distribuidores mencionan que están asegurados pero el peligro de robo y de sufrir lesiones también está presente.

Nota. Detalle de opiniones de los distribuidores. Elaboración propia.

1.4 Tecnologías de geolocalización

1.4.1 Geolocalización

La geolocalización es una tecnología utilizada en diversas áreas, desde la navegación, publicación de mapas en línea, seguimiento de vehículos y personas, entre otras. La geolocalización se refiere a la capacidad de determinar la ubicación física de un objeto o persona utilizando tecnologías como el Sistema de Posicionamiento Global (GPS), Wi-Fi, torres de telefonía móvil y Bluetooth. (Evaluando Software.com, 2021)

Por ejemplo:

Tabla 6

Ejemplo de usos de geolocalización en distintos escenarios

Navegación	Los sistemas de navegación GPS son ampliamente utilizados para proporcionar información de dirección y ruta en tiempo real a los conductores.
-------------------	---

Redes sociales	Las aplicaciones de redes sociales utilizan la geolocalización para permitir a los usuarios compartir su ubicación y encontrar amigos cercanos.
Comercio electrónico	Los servicios de comercio electrónico utilizan la geolocalización para ofrecer a los clientes promociones y descuentos en función de su ubicación geográfica.
Seguridad	Las aplicaciones de seguridad utilizan la geolocalización para proporcionar información de emergencia a los servicios de rescate y permitir a los usuarios enviar alertas de seguridad en tiempo real.
Publicidad	La publicidad geolocalizada permite a las empresas enviar anuncios relevantes a los usuarios en función de su ubicación geográfica.

Nota. Elaboración propia.

La mayoría de las aplicaciones de software tienen detección de ubicación integrada. Esto aumenta el alcance de las tareas que puede realizar y mejorar la calidad de los servicios prestados. (Madleňák, 2021)

La geolocalización determina la ubicación territorial en cualquier lugar que este situado un objeto o persona, desde un dispositivo tecnológico como computadora, smartphones o directamente por el receptor interno GPS del dispositivo. (Manzaba, 2020)

Por otra parte, Gerson (Beltrán, 2016) define que: la geolocalización es una tecnología que combina la ubicación de objetos del mundo real con la información obtenida a través de internet. Esto incluye aplicaciones de geolocalización en línea que se pueden utilizar desde cualquier dispositivo conectado a internet, como un ordenador de escritorio, un portátil, un dispositivo móvil o un navegador GPS. Para llevar a cabo el proceso de geolocalización, se utiliza un dispositivo hardware que actúa como plataforma. (pág. 60).

Es importante diferenciar los términos: Geolocalización y Georreferenciación, sus significados no son muy similares y se aplican en diferentes contextos (Avila, 2022), por lo tanto:

La geolocalización es un término asociado al ámbito social, por medio del cual se puede inferir la localidad de un objeto o persona en base a diversas fuentes, estas fuentes pueden ser variadas, como por ejemplo recursos multimedia como videos y/o fotografías, en este campo también se asocian a las redes sociales como Facebook, Twitter entre otros como fuente de

información por el cual se puede acceder a información que permita determinar la posición de objetos y/o personas (Schlosser et al., 2021)

En tanto la georreferenciación, de acuerdo con la Deyde DataCentric, define que: es la técnica de posicionamiento espacial de una entidad en una localización geográfica única y bien definida, mediante un sistema de coordenadas y un datum específicos. La georreferenciación está relacionada a un punto en un contexto geográfico que se obtiene a partir de datos que no se están tomando “en tiempo real”. (Deyde DataCentric, 2018)

por lo que se puede aseverar que nos iguales, los dos sirven para marcar un punto sobre el planeta, pero en distintas formas.

Tabla 7

Diferencias entre Geolocalización y Georreferenciación

Geolocalización	Georreferenciación
Termino asociado al ámbito social	Termino asociado a la geociencia
Proceso por el cual se puede determinar donde se encuentra localizado un objeto y/o persona.	Proceso por el cual un objeto y/o persona es atribuido un valor geográfico que determina la posición de este en la tierra.
La fuente de información pueden ser fotos, videos, redes sociales, entre otros.	La fuente de información puede ser las constelaciones de satélites que permiten determinar las coordenadas de un objeto, sistemas GIS, entre otros.

Nota. Se establece una comparativa entre Geolocalización y Georreferenciación.

Fuente: (Ávila, 2022)(UNR, 2019)(Schlosser et al., 2021).

1.4.2 Sistema de Posicionamiento Global (GPS)

Los dispositivos GPS completan un sistema de radionavegación vía satélite desarrollado y gestionado por el departamento de defensa de los EE. UU, creado originalmente con fines militares y entro en total operación desde julio de 1995. (Julen Castellano & David Casamichana, 2014) Este sistema, que hoy se utiliza con regularidad, es posible gracias al descubrimiento de la resonancia magnética que permitió a su vez la creación de relojes atómicos de elevada precisión y que son la base de esta tecnología. (Rigden, 2000)

Actualmente, se encuentran en funcionamiento el sistema norteamericano GPS, el ruso GLONASS; y aunque prestan ya algunos servicios, están en fase de desarrollo el Europeo GALILEO (primeros 2 satélites lanzados en octubre de 2011 y primeros servicios en diciembre de 2016), el chino BEIDOU (primera versión en el año 2000), el japonés QZSS (primer satélite lanzado en 2010) y el indio IRNSS. (García, 2019)

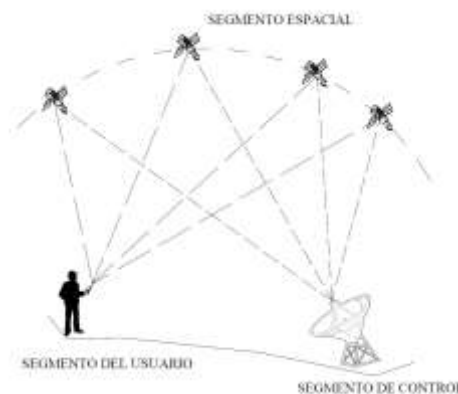
Se utiliza ampliamente en una variedad de aplicaciones, desde navegación en vehículos, en aviones y hasta seguimiento de la ubicación en tiempo real en aplicaciones móviles. Los dispositivos GPS constan de un receptor GPS que recibe señales de satélite y un software que interpreta estas señales para calcular la posición. (Yujra, 2017)

1.4.3 Componentes del Sistema GPS y funcionamiento

El sistema GPS se divide en tres segmentos: espacial, control y usuario. Los segmentos espaciales y control son gestionados por Navstar (Navigation Satellite Timing and Ranging) EE. UU que está formado por 27 satélites emisores de las señales. El segmento control está constituido por un conjunto de estaciones de elevada precisión situadas estratégicamente en la tierra, incluyendo una estación principal o maestra y varias antenas terrestres. Finalmente, el segmento usuario corresponde a los receptores GPS que han sido diseñados para recibir y decodificar las señales transmitidas por los satélites. (Castellano & Casamichana, 2014)

Figura 6

Segmento de Navstar



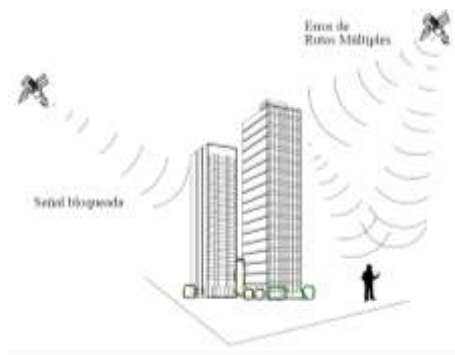
Nota. Tomado de (Casanova, 2018).

La precisión de esta técnica es exacta con errores de centímetros, sin embargo, al ser ondas de radio es susceptible a que existan múltiples inconvenientes, como el ruido, el movimiento de

los objetos tanto del emisor como del receptor, la ubicación es un impedimento para que las ondas de radio no sean receptadas y enviadas de manera correcta. (Kumar et al., 2021)

Figura 7

Ondas de radio bloqueadas



Nota. Tomado de (Casanova, 2018).

Los dispositivos portátiles tienen una gran acogida por parte de los usuarios, existen dispositivos que permiten realizar un uso específico y otros que permiten múltiples tareas, un ejemplo de esto es los celulares. Estos dispositivos pueden ser pasivos o activos, pasivos son aquellos dispositivos portátiles que no emiten sus coordenadas de regreso si no que son almacenados dentro de estos para luego ser analizados, al contrario de los dispositivos portátiles activos que, si emiten en tiempo real sus coordenadas. (Ubicalo, 2021) Los pasos que debe seguir para determinar una ubicación se detallan en la siguiente tabla:

Tabla 8

Funcionamiento del Sistema GPS, Escenario de ejemplo con tres satélites

Paso 1: Cada satélite transmite de forma continua su código de sincronización. Un receptor tras haber recibido el código de sincronización, lo compara con su propio código producido de manera local, para determinar el tiempo de propagación. Esa diferencia de tiempo multiplicada por la velocidad de la señal de radio determina la distancia al satélite.

Paso2: Cada satélite indica que el receptor se encuentra en un punto de la superficie de una esfera como centro en sí mismo y el radio la distancia total del receptor. Cuando se recopila la información de dos satélites, se señala que el receptor se encuentra sobre el anillo resultante de cortarse las dos esferas individuales de cada satélite.

Paso 3: Para afinar la posición, se adquiere la misma información de un tercer satélite y un nuevo anillo corta el anterior en dos puntos. Uno de ellos se puede descartar porque no ofrece

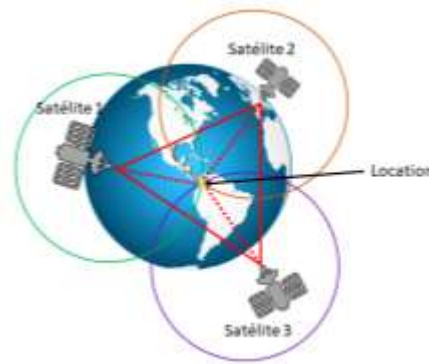
una posición coherente. Así se tendría la posición 3D; el inconveniente es que el reloj incorporado en los receptores GPS, no está sincronizado con los relojes atómicos de los satélites, por lo tanto, los dos puntos determinados no son precisos.

Paso 4: Para corregir esa falta de sincronización con un cuarto satélite, que generará una posición 3D más exacta. La intersección resultante de las cuatro esferas con centro en los satélites es un pequeño volumen en lugar de ser un punto. Para esta corrección, a medida que se utilicen los satélites y se obtenga más intersecciones, la hora del receptor se ajustará de manera que el volumen se transformara en un punto más exacto.

Nota. Tomado de (García, 2019).

Figura 8

Georreferenciación – GPS: Triangulación satelital.



Nota. Elaboración propia.

1.4.4 Dispositivos en red mediante direcciones IP

La geolocalización mediante direcciones IP es una técnica que se utiliza para determinar la ubicación geográfica de un dispositivo conectado a internet. Cada dispositivo que se conecta a internet se le asigna una dirección IP única, que es una serie de números que identifica al dispositivo en la red. A través de esta dirección IP, es posible determinar la ubicación del dispositivo en el mundo real.

Existen bases de datos que contienen información sobre las direcciones IP y sus ubicaciones geográficas asociadas. Estas bases de datos se actualizan regularmente y se utilizan en muchas aplicaciones de geolocalización basadas en IP, como la publicidad en línea, la seguridad de redes y la identificación de fraudes en línea. Sin embargo, la precisión de la geolocalización

basada en IP puede variar y no siempre es exacta debido a factores como el uso de redes privadas virtuales (VPN) o la asignación incorrecta de una dirección IP a un dispositivo.

1.4.5 Aplicaciones móviles con geolocalización

Al principio, los servicios de localización en redes móviles no fueron vistos con fines comerciales, sino como potentes herramientas para localizar llamadas de emergencia y servir de apoyo en tareas de salvamento (Bellavista et al., 2008). Pero, en la actualidad existen un número limitado de formas para localizar un objeto en la tierra, según (Evaluando Software.com, 2021) las principales formas y las más usadas son por medio de GPS, dispositivos en red y la telefonía móvil.

1.4.5.1 Telefonía móvil

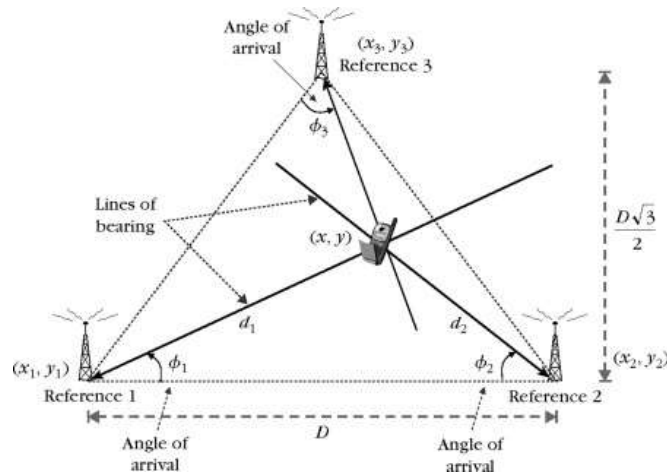
La telecomunicación ha revolucionado la forma en que las personas se comunican y ha llevado a las compañías telefónicas a competir para mejorar sus servicios y la calidad que brindan a sus clientes. Según (Kos et al., 2006), "hoy en día es de conocimiento público que las telefonías compiten entre ellas para mejorar los servicios y la calidad de estos a sus clientes, de estas constantes mejoras nace la geolocalización como parte de servicios que brindan estas". La geolocalización se realiza a través de las torres telefónicas ubicadas en un área determinada, que transmiten señales a los dispositivos móviles para georreferenciarlos mediante la aplicación de geometría simple.

La geolocalización a través de dispositivos móviles se realiza utilizando varios sensores de ubicación como GPS, Wi-Fi, Bluetooth y redes celulares, que permiten determinar la ubicación geográfica del usuario. Estos sensores utilizan señales de satélite y puntos de acceso cercanos para calcular la ubicación del dispositivo. La técnica de geolocalización se utiliza en varias aplicaciones móviles, incluyendo mapas y navegación, servicios de emergencia, seguimiento de entregas y servicios de entrega de alimentos.

En la figura 9 se muestra una representación gráfica de la georreferenciación móvil mediante la metodología Angle of Arrival (AoA).

Figura 9

Georreferenciación – Telefonía móvil



Nota. Tomado de (Munoz et al., 2009).

1.5 Técnicas, metodologías y licitación de requisitos

1.5.1 ISO-IEC-IEEE 29148-2011

Los requerimientos de software en su definición más simple es lo que expresa las necesidades y funcionalidades requeridas en un software para resolver problemas del mundo real, estas necesidades de cierta manera son críticas cuando han sido realizadas de manera errónea porque pueden conducir a un desarrollo de una aplicación que no se buscaba y no resuelve una problemática planteada, lo que inevitablemente lleva a pérdidas de dinero, esfuerzo y tiempo. (SWEBOK, 2014)

Un requisito de software es una unidad que expresa lo que debería y lo que necesita hacer el sistema, un requisito representa funcionalidad por lo tanto diseños de la aplicación usualmente no son considerados como requisitos, sin embargo, a medida como estos son analizados por ejemplo un requisito en función de la arquitectura de un sistema ya conlleva a que obtenga un valor superior. (ISO/IEC/IEEE 29148, 2011)

La licitación de requisitos es el punto de partida del desarrollo de software y netamente es una actividad humana en la cual se extraen los requerimientos del software, es la fase en la cual se entiende la problemática a resolver con una propuesta tecnológica, en esta fase del proceso de requisitos del software es muy importante la interacción entre la persona que es designada como la representante del equipo de desarrollo encargada de comunicarse con uno o varios Stakeholders

los cuales presentaran los requerimientos, es común que esta actividad se realice en más de una vez. (SWEBOK, 2014)

(ISO/IEC/IEEE 29148, 2011) Menciona que la licitación de requisitos es una actividad iterativa, se puede repetir las veces que se consideren necesarias para lograr entender las verdaderas necesidades de los clientes, para la realización de esta actividad existen múltiples metodologías que pueden ser aplicadas, de las cuales se destacan:

- Brainstorming
- Entrevistas y cuestionarios
- Revisión documental
- Análisis del mercado
- Menchmarking
- Análisis técnico organizacional

1.5.2 Metodología Extreme Programming (XP)

XP fue creado por Kent Beck en marzo de 1996, durante su trabajo en el proyecto de nómina de sistema de compensación integral, desde entonces ha venido perfeccionándose y adaptándose hasta la actualidad. Este proceso agile de desarrollo de software, se diferencia de las metodologías tradicionales principalmente pone más énfasis en la adaptabilidad que en la previsibilidad (Ulloa, 2014). XP también es un modelo de desarrollo de software que se centra en la calidad de software, mejoras y capacidad de respuesta a los cambiantes requisitos del cliente (Yasvi & Yadav, 2019a). Esta metodología se basa en ciclos rápidos de lanzamiento y comunicación continua entre los desarrolladores y las parte interesadas, en gran medida la comunicación es oral, las pruebas son frecuentes, la revisión y diseño de código (Yasvi & Yadav, 2019b). XP ayuda a realizar entregas rápidas con una exposición mínima de riesgo y es adecuado para proyectos de software pequeños (Yasvi & Yadav, 2019c).

XP cuenta con 12 prácticas que se llevan a lo largo del ciclo de desarrollo, estas prácticas garantizan una mejor legibilidad, comprensión de código y también ayudan a mejorar la comunicación entre el desarrollador y el cliente para que se cree un mejor producto.

Tabla 9

Prácticas de la metodología XP

Planificación	Establecer objetivos claros y metas para el proyecto.
----------------------	---

Lanzamiento pequeño	Dividir el trabajo en pequeños entregables y frecuentes.
Metáfora	Utilizar una metáfora compartida para comprender el sistema.
Diseño simple	Mantener el diseño del sistema lo más simple posible.
Pruebas	Realizar pruebas de manera continua a lo largo del desarrollo.
Refactorización	Mejorar constantemente la estructura del código sin cambiar su funcionalidad.
Programación en parejas	Trabajar en colaboración, con dos programadores en una computadora.
Propiedad colectiva	Todos los miembros del equipo son responsables del código.
Integración continua	Integrar el código en el repositorio compartido con frecuencia.
Semana de 40 horas	Evitar el exceso de trabajo para mantener la sostenibilidad del equipo.
Cliente en el sitio	Tener al cliente disponible para responder preguntas y tomar decisiones.
Estándares de codificación	Establecer estándares de codificación para mantener la consistencia.

Fuente: (Yasvi & Yadav, 2019).

XP también tiene otras aplicaciones, por ejemplo: la combinación de Extreme Programming (XP) y Usability Engineering (UE) que lleva a un nuevo método llamado Extreme Usability (XU). XU es muy prometedor, especialmente para la educación en ingeniería de software (Yasvi & Yadav, 2019e). permite la participación de diferentes clientes en diferentes iteraciones, lo que puede mejorar la cobertura general del tiempo del cliente en el equipo y permitir pruebas de usabilidad en diferentes usuarios finales reales en cualquier momento. Además, XU mantiene todas las mejores prácticas de Usability Engineering (UE) en XP durante la planificación, con una restricción de los aspectos de usabilidad en la próxima iteración y el tratamiento igualitario de Usabilidad y Funcionalidad (Holzinger et al., 2005).

1.6 Sistemas Operativos y aplicaciones

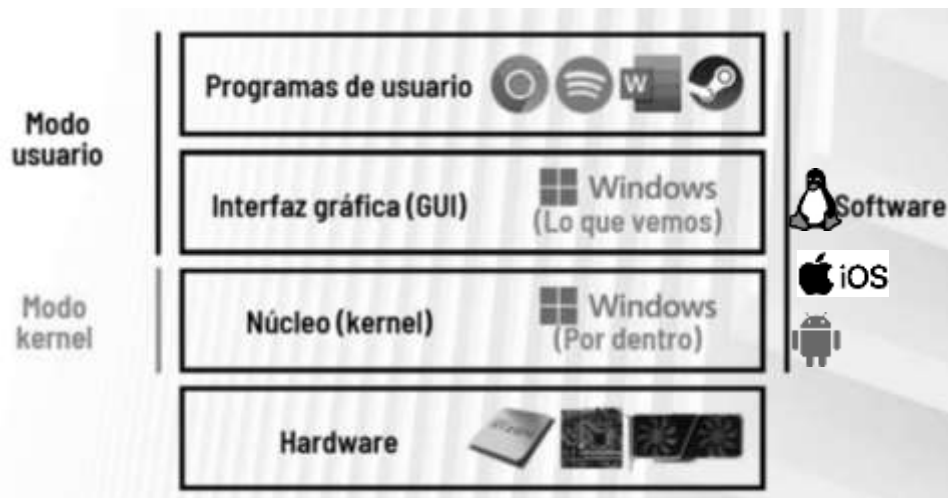
Según (Maza Maria & Bohorquez Yesica, 2012; Salazar Steven, 2017), un sistema operativo es el software fundamental que conecta el hardware de un dispositivo con el usuario y las capacidades y forma de interactuar de un dispositivo móvil dependen del sistema operativo que

utilice. Los sistemas operativos móviles administran los recursos de la máquina, como el procesador y la memoria, y ofrecen servicios a las aplicaciones, como el acceso a la red y la gestión de archivos. Cada sistema operativo móvil tiene sus propias características y aplicaciones específicas, lo que los hace adecuados para diferentes tipos de dispositivos móviles y usuarios.

Los dispositivos móviles necesitan un sistema operativo para funcionar y realizar todas las tareas que el usuario espera de ellos. Existen muchas opciones de sistemas operativos móviles en el mercado, incluyendo Symbian OS, Windows Mobile, iPhone OS, Android y Linux, cada uno con sus propias fortalezas y debilidades. Por lo tanto, la elección de un sistema operativo móvil es crucial para determinar las capacidades y la funcionalidad de un dispositivo móvil.

Figura 10

Estructura de un sistema operativo



Nota. Fuente: <https://www.profesionalreview.com/2022/10/05/sistema-operativo/>

Las características principales de un sistema operativo móvil son:

Tabla 10

Características de un sistema operativo

Interfaz de usuario: proporciona una interfaz de usuario que permite al usuario interactuar con el dispositivo móvil.

Gestión de recursos: gestiona los recursos de hardware del dispositivo móvil, como el procesador, la memoria y el almacenamiento.

Multitarea: permite a los usuarios ejecutar varias aplicaciones al mismo tiempo.

Seguridad: ofrece características de seguridad para proteger el dispositivo móvil y sus datos.

Conectividad: ofrece características de conectividad, como Wi-Fi, Bluetooth, y redes móviles.

Personalización: permite a los usuarios personalizar el dispositivo móvil según sus preferencias.

Soporte de aplicaciones: proporciona un entorno para que las aplicaciones se ejecuten y se comuniquen con el hardware del dispositivo.

Actualizaciones: recibe actualizaciones periódicas que mejoran la funcionalidad y la seguridad del dispositivo móvil.

Nota. Fuente: (StudySmarter, 2023).

1.6.1 Sistema Operativo Android

Android es una plataforma de código abierto para el desarrollo de aplicaciones móviles que se basa en el núcleo operativo Linux. Originalmente fue creada por la compañía Android, Inc. y respaldada financieramente por Google, quien finalmente la adquirió en 2005. En 2007, Google lideró la creación de la Open Handset Alliance (OHA), conformada por 34 miembros que incluyen fabricantes de dispositivos móviles, desarrolladores de aplicaciones, operadores de telecomunicaciones y fabricantes de chips (Velazco Raúl, 2017). Actualmente, Android se ha consolidado como la plataforma líder para dispositivos móviles frente a otros sistemas operativos como iOS, Windows Phone, BlackBerry, Palm, Java Mobile Edition o Linux Mobile, y su última versión disponible es Android 12.

1.6.2 Sistema Operativo IOS

iOS es un sistema operativo móvil exclusivo de Apple Inc. que fue presentado en 2007 bajo el nombre de iPhone OS en la MacWorld Conference. Fue desarrollado originalmente para el iPhone, pero desde entonces ha sido utilizado en otros dispositivos como el iPod touch, iPad y Apple TV (Velazco Raúl, 2017). El sistema operativo se basa en el sistema operativo macOS de Apple y comparten el mismo núcleo Mach/FreeBSD y lenguajes de programación principales como C y Objective-C.

La interfaz de usuario de iOS se basa en el concepto de manipulación directa y utiliza gestos multitáctiles para interactuar con el sistema operativo. La respuesta a las órdenes del usuario es inmediata y la interacción incluye gestos como deslizamientos, toques y pellizcos que tienen diferentes definiciones según el contexto. Apple no permite la instalación de iOS en hardware de terceros.

1.6.3 Análisis de los sistemas operativos móviles más populares: iOS y Android

En la tabla se menciona los análisis estadísticos de los sistemas operativos que se usan y es preferido por los usuarios.

Tabla 11

Sistemas operativos más populares más utilizados

	ANDROID	iOS
Cifra mundial	De acuerdo con las estadísticas proporcionadas por StatCounter en septiembre de 2023, Android es el sistema operativo móvil más utilizado en todo el mundo, con una cuota de mercado global del 69.74%.	iOS ocupa el segundo lugar con una cuota del 29.58%.
Popularidad de Mercado	En cuanto a su popularidad regional, Android es el sistema operativo móvil más utilizado en la mayoría de las regiones del mundo, incluyendo Asia, Europa, América Latina y África.	Por otro lado, iOS es más popular en América del Norte, Europa occidental y Australia.
Descarga de Aplicaciones	En cuanto al uso de aplicaciones móviles, según los datos de Data.ai en diciembre de 2022, tanto la App Store de iOS como la Play Store de Android registraron un aumento en las descargas de aplicaciones.	Aunque es importante destacar que la App Store de iOS registró un mayor gasto en aplicaciones que la Play Store de Android.

Nota. Fuente: (Jonathan Briskman, 2022; StatCounter, 2023).

1.5.5 Aplicación móvil

Las aplicaciones móviles permiten realizar una amplia variedad de tareas de manera sencilla y eficiente, mejorando la comunicación y facilitando la difusión de información a través del uso de recursos tecnológicos en dispositivos móviles.

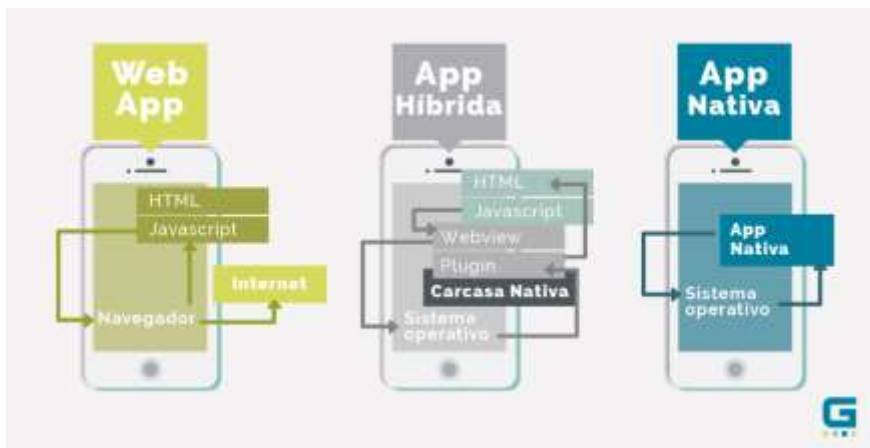
Una aplicación móvil, también conocida como "app", es un software diseñado para cumplir una función específica o genérica, que permite al usuario interactuar con ella y realizar diversas tareas. Estas aplicaciones pueden venir preinstaladas en el dispositivo móvil de fábrica, pero muchas también se pueden descargar a través de una conexión a internet para que el usuario las instale en su dispositivo (Pinos Kevin, 2017). Las aplicaciones pueden ser de pago o gratuitas, y pueden ser utilizadas para una variedad de propósitos, desde juegos y entretenimiento hasta productividad y educación. Es importante destacar que las aplicaciones móviles se han convertido en una parte esencial de la vida de muchas personas, y han transformado la forma en que interactuamos con nuestros dispositivos móviles y entre nosotros mismos.

1.5.6 Tipos de aplicaciones

Existen tres variedades de aplicaciones: las aplicaciones nativas, las aplicaciones web y las aplicaciones híbridas. Estas últimas son el resultado de combinar las funcionalidades de las aplicaciones nativas con las aplicaciones web, que se elaboran con tecnologías web como HTML, CSS y JavaScript. Estas tecnologías se compilan y se empaquetan de manera que se genera una aplicación para dispositivos móviles.

Figura 11

Tipo de aplicaciones móviles.



Nota. Adaptado de Innovaciones Informáticas GSOFT, Europa 2018, <https://www.gsoft.es/articulos/que-necesito-web-apps-app-nativa-o-app-hibrida/>

1.5.6.1 Nativas

Las aplicaciones nativas son aquellas desarrolladas con un lenguaje y entorno de desarrollo específicos, lo que garantiza funcionamiento estable y fluido en el sistema operativo para el que

fueron creadas. Aunque estas aplicaciones tienen ventajas, también tienen desventajas (García María, 2015). Por ejemplo, no requieren conexión a internet para funcionar y se descargan e instalan a través de tiendas de aplicaciones específicas, lo que facilita el proceso de marketing y promoción. Además, pueden utilizar las notificaciones del sistema operativo para mostrar avisos importantes al usuario, incluso si la aplicación no está abierta. Por otro lado, también pueden aprovechar todas las características de hardware del dispositivo móvil, como la cámara y los sensores.

1.5.6.2 Web

Las aplicaciones web son aquellas que se ejecutan dentro del navegador web de un dispositivo a través de una URL, lo que permite el acceso a la información desde cualquier sistema operativo y dispositivo (García María, 2015). A diferencia de las aplicaciones nativas, las aplicaciones web no necesitan ser instaladas y siempre ofrecen la última versión. El desarrollo de aplicaciones web es más económico que el de aplicaciones nativas, y la interfaz suele ser más genérica e independiente de la apariencia del sistema operativo, lo que puede reducir la experiencia de identificación del usuario con los elementos de navegación e interacción.

1.5.6.3 Híbridas

Las aplicaciones híbridas combinan elementos de las aplicaciones nativas y de las aplicaciones web, utilizando lenguajes propios de la web y permitiendo su uso en diferentes plataformas (García María, 2015). Se pueden desarrollar con herramientas gratuitas y se pueden integrar con herramientas de aplicaciones nativas. Estas aplicaciones se mantienen en la tienda de aplicaciones y pueden aprovechar las capacidades del hardware del dispositivo, además de acceder a las características del teléfono a través de librerías. Aunque su diseño visual no se identifica en gran medida con el del sistema operativo, permiten un desarrollo más rápido y económico que las aplicaciones nativas.

1.7 Herramientas de desarrollo

En este capítulo se detalla las herramientas que se utilizaron en este proyecto, abarcan un conjunto de software y recursos que permitieron diseñar, codificar, depurar, optimizar y agilizar el proceso de desarrollo.

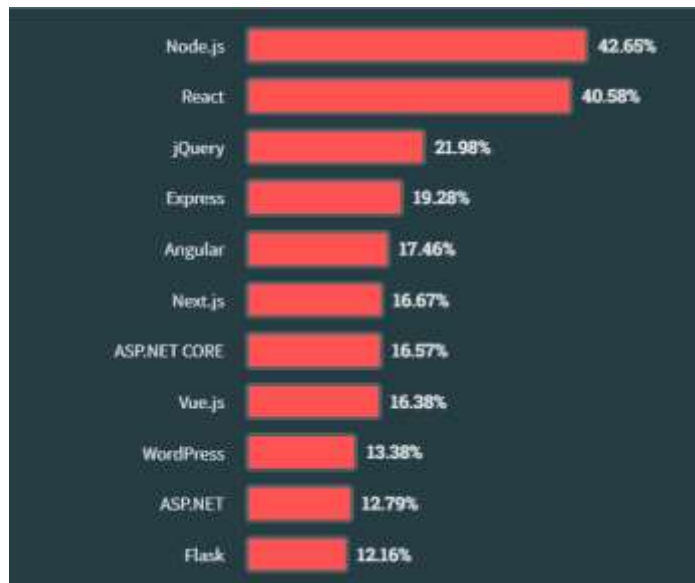
1.7.1 Entorno de desarrollo Node.JS

Node.js es una plataforma de desarrollo de software basada en eventos y diseñada para la construcción de aplicaciones web y servidores escalables y de alto rendimiento. Se destaca por su capacidad para manejar múltiples conexiones de forma asincrónica, lo que la hace ideal para aplicaciones de tiempo real (Morales Gustavo, 2020).

Entonces, las estadísticas de Node.js muestran que, desde su creación en 2009 por Ryan Dahl, NodeJS ha sido popular entre los desarrolladores y propietarios de empresas. Sin embargo, hay varias razones por las que utilizar Node.js para el desarrollo BackEnd en 2023 (Ashvin Kumbhani, 2023). Las estadísticas de uso de NodeJS entre desarrolladores es una de las opciones preferidas entre propietarios de empresas y desarrolladores en la actualidad. Para comprender mejor esto, consideremos los datos disponibles con (Stack Overflow, 2023). Node.js es la tecnología web más común utilizada por todos los desarrolladores. Ha obtenido el 42,65% de los votos y se ha convertido nuevamente en la tecnología más popular y utilizada por los desarrolladores.

Figura 11

Porcentaje de uso a nivel mundial de NodeJS



Nota. Adaptado de: (Stack Overflow, 2023).

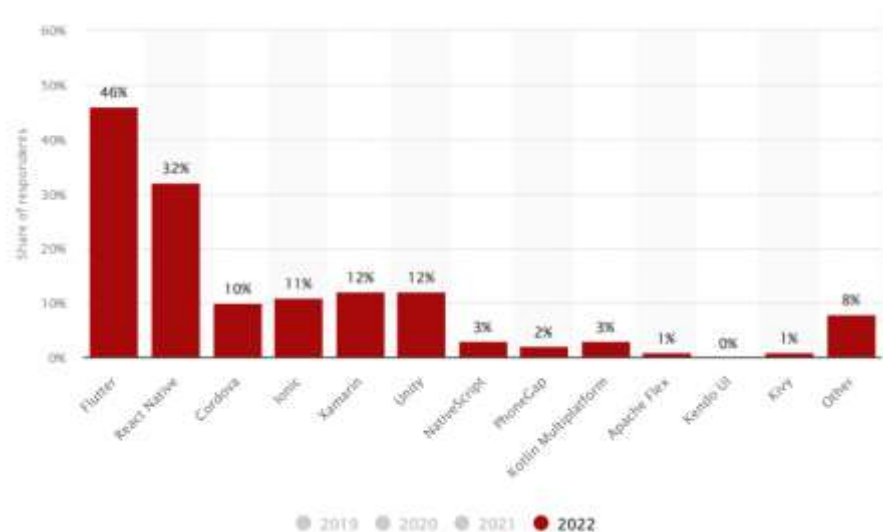
Además, Node.js es altamente flexible y adaptable gracias a su ecosistema de módulos y librerías disponibles, que simplifican el proceso de desarrollo y permiten a los desarrolladores enfocarse en la lógica del negocio.

1.7.2 Framework Flutter

Flutter es un framework de desarrollo de aplicaciones móviles de código abierto. La peculiaridad de Flutter radica en que construye sus propios widgets a nivel de aplicación, lo que permite una mayor libertad en el diseño de las aplicaciones. Flutter proporciona un marco de desarrollo móvil, reducen el tiempo de desarrollo, ya que proporcionan las bibliotecas, los compiladores y los componentes móviles necesarios para las diferentes plataformas móviles. Los equipos de software también pueden utilizarlos para crear aplicaciones multiplataforma, crear aplicaciones móviles a partir de una única base de código que funciona en Android, iOS, Windows y macOS (AWS, 2023). Flutter es una tecnología muy nueva pero madura, ya que es utilizada por Google en sus herramientas internas. Su curva de aprendizaje es muy baja gracias al uso de un único lenguaje sencillo. En resumen, Flutter es el marco móvil multiplataforma más popular utilizado por los desarrolladores globales con el 46%, según una encuesta de desarrolladores hasta 2022 (Statista & Sujay, 2023); además de contar con herramientas de desarrollo de alta calidad y una curva de aprendizaje baja (Vázquez, 2018).

Figura 12

Marcos móviles multiplataforma



Nota. Fuente: Adaptado de: (Statista & Sujay, 2023).

1.7.3 Gestor de base de datos

1.7.3.1 Sql

SQL (Structured Query Language) es un lenguaje de programación estandarizado utilizado para trabajar con bases de datos relacionales. Es un lenguaje declarativo que se utiliza para definir, consultar y modificar datos en una base de datos relacional. A través de su capacidad para combinar operaciones algebraicas y de cálculo relacional con operadores adicionales, permite la creación de consultas y la realización de cambios en una base de datos de manera eficiente. Además, SQL no sólo incluye comandos de consulta, sino que también cuenta con una amplia variedad de comandos, cláusulas, operadores y funciones de agregado que permiten definir, mantener y consultar bases de datos relacionales de manera sencilla y efectiva. (Valencia Luis, 2021).

1.7.3.2 noSql

El término NoSQL surgió con la llegada de la web 2.0, cuando cualquier usuario podía subir contenido a la red, lo que provocó un gran aumento de datos y problemas en la gestión de la información almacenada en bases de datos relacionales. Para solucionar estos problemas de escalabilidad y rendimiento, surgieron sistemas de almacenamiento de información que no cumplen con el esquema entidad-relación y utilizan otros formatos de almacenamiento, como clave-valor, mapeo de columnas o grafos. Estas bases de datos NoSQL son ideales para situaciones donde las bases de datos relacionales no son suficientes, como en casos de miles de usuarios concurrentes y millones de consultas diarias. Además, las bases de datos NoSQL permiten una mayor flexibilidad y escalabilidad en el almacenamiento de datos que las bases de datos relacionales tradicionales (Acens, 2013).

Tabla 12

Diferencias entre SQL y NoSQL

	SQL	NoSQL
Estructura de datos	Utiliza una estructura de datos tabular	Utiliza diferentes estructuras de datos como clave-valor, documentos, grafos, entre otros.
Escalabilidad	Se limita en escalabilidad vertical	Es altamente escalable horizontalmente.
Tipo de consulta	Utiliza el lenguaje declarativo para consultas	Utiliza el lenguaje de programación para consultas.

Transacciones	Garantiza la integridad de los datos mediante transacciones ACID	Puede ser eventualmente consistente
Flexibilidad	Requiere cambios en el esquema	Flexible en términos de cambios en la estructura de los datos
Velocidad	Es más rápido en el procesamiento de grandes cantidades de datos no estructurados	Es más rápido en el procesamiento de datos estructurados y complejos.
Escenario de uso	Adecuado para aplicaciones con datos estructurados	Adecuado para aplicaciones con datos no estructurados.

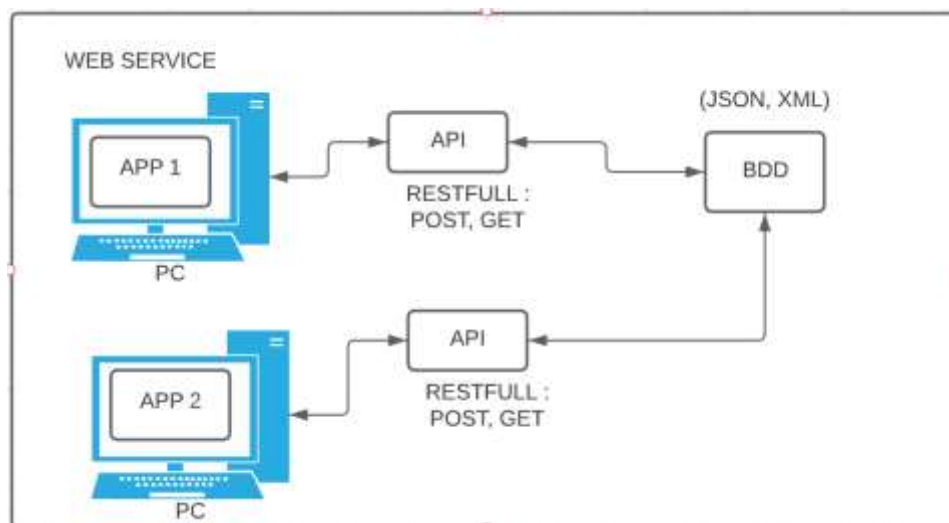
Nota. Se detallan las características de las bases de datos. Elaboración propia.

1.7.4 Servicios web

Los servicios web (web services, en inglés) son un conjunto de tecnologías y protocolos que permiten a diferentes aplicaciones, sistemas y dispositivos comunicarse e intercambiar datos a través de Internet. Estos servicios se basan en estándares abiertos, como HTTP, XML y SOAP, y utilizan lenguajes de programación como Java, Python o .NET para implementar la lógica empresarial.

Figura 12

Funcionamiento de Web Service



Nota. Se detalla el flujo de información mediante los métodos de las Apis. Recuperado de DeveloSt. Accueil. (2023). Retrieved 24 May 2023, from <https://acortar.link/oTTWnw>.

1.7.4.1 SOAP

SOAP (Simple Object Access Protocol) es un protocolo estándar de comunicación basado en XML (Extensible Markup Language) utilizado para intercambiar información entre aplicaciones a través de redes de computadoras. SOAP define un formato de mensaje que incluye encabezados y cuerpos de mensajes, lo que permite la transferencia de datos estructurados y complejos entre sistemas heterogéneos (IBM, 2021a).

SOAP utiliza HTTP (Hypertext Transfer Protocol) como protocolo de transporte para enviar y recibir mensajes entre aplicaciones, y puede ser utilizado en una variedad de plataformas y lenguajes de programación. La estructura de un mensaje SOAP incluye una sección de encabezado y una sección de cuerpo, y puede contener información sobre la autenticación, el control de acceso y la seguridad. Además, SOAP también admite la definición de servicios web mediante la especificación de WSDL (Web Services Description Language), lo que facilita la integración entre sistemas de diferentes organizaciones y proveedores (IBM, 2021b).

1.7.4.2 API REST

API REST (Representational State Transfer) es una arquitectura de software que permite la comunicación y transferencia de datos entre sistemas a través del protocolo HTTP. Esta arquitectura se basa en el principio de que cada recurso (por ejemplo, una entidad de una base de datos) puede ser identificado mediante una URI (Uniform Resource Identifier) y accedido mediante los métodos estándar de HTTP, como GET, POST, PUT y DELETE. Además, API REST utiliza un formato de intercambio de datos como JSON o XML para enviar y recibir información entre los sistemas. Esta arquitectura es ampliamente utilizada en el desarrollo de aplicaciones web y móviles, y permite una integración más sencilla y flexible entre diferentes sistemas (Amodeo Enrique, 2013).

1.7.4.3 GraphQL

Tabla 13

Características de REST, SOAP, GAPHQL

Característica	REST	SOAP	GraphQL
Protocolo	HTTP	HTTP, SMTP, FTP, etc.	No específico
Formato de mensaje	JSON, XML, etc.	XML	JSON

Lenguaje de descripción	Swagger, RAML, etc.	WSDL	SDL (Schema Definition Language)
Método de acceso a datos	CRUD (Create, Read, Update, Delete)	CRUD	Solo lectura
Escalabilidad	Escalable	Menos escalable	Escalable
Uso de ancho de banda	Uso eficiente del ancho de banda	Uso ineficiencia del ancho de banda	Uso eficiente del ancho de banda
Caché	Compatible con la caché	Compatible con la caché	No hay soporte para caché
Estructura del mensaje	Sin estructura definida	Estructura definida	Estructura definida por el esquema
Curva de aprendizaje	Fácil de aprender	Difícil de aprender	Moderadamente difícil de aprender
Soporte de plataforma	Amplio soporte de plataforma	Menor soporte de plataforma	Amplio soporte de plataforma

Nota. Elaboración propia.

En comparación con otros servicios web como SOAP y GraphQL, una API REST tiene varias ventajas. En primer lugar, su estructura simple y bien definida la hace fácil de entender y utilizar para los desarrolladores de aplicaciones. Además, una API REST es altamente escalable, lo que significa que puede manejar grandes volúmenes de tráfico y solicitudes sin afectar el rendimiento de la aplicación. En segundo lugar, una API REST es más flexible en términos de formato de datos, ya que puede enviar y recibir datos en diferentes formatos, como JSON y XML, lo que permite que se integre fácilmente con diferentes aplicaciones y sistemas. En tercer lugar, una API REST es ampliamente adoptada y utilizada en la industria, lo que significa que hay una gran cantidad de recursos y herramientas disponibles para ayudar a los desarrolladores a implementar APIs RESTful en sus aplicaciones.

1.7.6 Android Estudio

Android Studio es el entorno de desarrollo integrado oficial de la plataforma Android. Se anunció en la conferencia Google I/O del 16 de mayo de 2013 que sustituiría a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android. En diciembre de 2014, se lanzó la

versión inicial estable. Está basado en IntelliJ IDEA de JetBrains y está disponible de forma gratuita bajo la Licencia Apache 2.0. Se puede ejecutar en GNU/Linux, macOS, Microsoft Windows y Chrome OS (Developers, 2023a).

Las herramientas para desarrolladores de Flutter y Dart, ofrece aún más funciones que mejoran tu productividad cuando compila aplicaciones para Android (Flutter, 2023), realiza cambios en caliente (hot reload), permite compilar y depura la aplicación en un dispositivo móvil externo, existe la facilidad de crear emuladores que ayuden a realizar la compilación de las aplicaciones móviles y además ayuda a perfeccionar el diseño con la ayuda de Lint, enfocado en usabilidad que proporciona el editor de código. Android estudio es el sexto entorno de desarrollo integrado preferido por los desarrolladores con el 16.82% de aceptabilidad (Stack Overflow, 2023).

Tabla 14

Características del entorno de desarrollo Android estudio

Nro.	Características de Android estudio
1	Un sistema de compilación flexible basado en Gradle
2	Un emulador rápido y cargado de funciones
3	Un entorno unificado donde puedes desarrollar para todos los dispositivos Android
4	Ediciones en vivo para actualizar elementos componibles en emuladores y dispositivos físicos, en tiempo real
5	Integración con GitHub y plantillas de código para ayudarte a compilar funciones de apps comunes y también importar código de muestra
6	Variedad de marcos de trabajo y herramientas de prueba
7	Herramientas de Lint para identificar problemas de rendimiento, usabilidad y compatibilidad de versiones, entre otros
8	Compatibilidad con C++ y NDK
9	Compatibilidad integrada con Google Cloud Platform, que facilita la integración con Google Cloud Messaging y App Engine

Nota. Fuente: (Developers, 2023b).

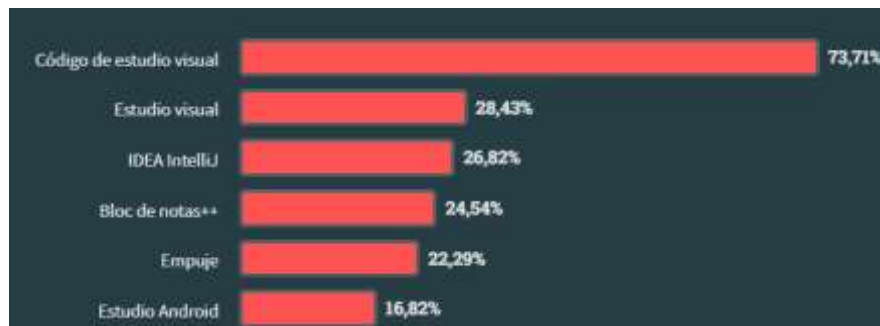
1.7.7 Visual estudio Code

Visual Studio Code es un editor de código fuente gratuito, liviano pero potente que se ejecuta en el escritorio, la web y está disponible para los sistemas operativos Windows, macOS, Linux y Raspberry Pi. Viene con soporte integrado para JavaScript, TypeScript y Node.js y tiene un rico ecosistema de extensiones para otros lenguajes de programación (como C++, C#, Java, Python, PHP y Go), tiempos de ejecución (como .NET y Unity), entornos (como Docker y Kubernetes) y nubes (como Amazon Web Services, Microsoft Azure y Google Cloud Platform).

Visual Studio Code tiene control de código fuente integrado que incluye compatibilidad con Git. Gran parte de esto fue adaptado de la tecnología de Visual Studio (Martin Heller, 2022). Visual Studio Code propiamente dicho se crea utilizando Electron Shell, Node.js, TypeScript y Language Server Protocol, y se actualiza mensualmente. El soporte online varía según los diferentes lenguajes de programación y sus extensiones, desde el simple resaltado de sintaxis y la coincidencia de corchetes hasta la depuración y refactorización. Puede agregar soporte básico para su idioma favorito a través de los colorizadores TextMate si no hay ningún servidor de idioma disponible (Visual Studio Code, 2023). Visual estudio code contiene miles de adaptaciones, a lenguajes de programación, ayuda a compilar y depura código en caliente (hot reload), depura y copia aplicaciones móvil y web y es el más utilizado y preferido por los programadores con una aceptabilidad de 74% entre 2018 a 2022 (Sujay, 2023). En cuanto a la preferencia del entorno de desarrollo integrado tiene una aceptación del 73.71% a nivel mundial (Stack Overflow, 2023).

Figura 13

Visual Studio Code sigue siendo el IDE preferido entre todos los desarrolladores



Nota. Fuente: (Stack Overflow, 2023).

1.7.8 Fly.io

Fly.io es una plataforma de computación en la nube que te permite desplegar y ejecutar aplicaciones web y de BackEnd en múltiples regiones del mundo (Fly.io, 2023b). Con fly.io, puedes crear y administrar máquinas virtuales (VM) que se ejecutan en contenedores Docker, y que se conectan entre sí mediante una red privada. Fly.io es una alternativa a Heroku que te ofrece más flexibilidad, control y opciones para desplegar tus aplicaciones en la nube. Fly.io te ofrece una serie de características para optimizar el rendimiento y la escalabilidad de tus aplicaciones, tales como: un potente CLI que te permite construir, desplegar y controlar tus VM desde la línea de comandos; un servicio de compilación remota que te permite crear imágenes Docker sin necesidad de tener Docker instalado en tu máquina local; una red privada que te permite comunicar tus VM entre sí y con servicios externos de forma segura y eficiente (Fly.io, 2023b).

Tabla 15

Características técnicas de Fly.io

Características	Fly.io
Categoría	PaaS
Empresa matriz	Independiente (Serie B)
Nivel gratuito	Sí
Plataforma de contenedores	Sí
Bases de datos	PostgreSQL Redis MySQL (usando Dockerfile) Firebase
HTTP2	Sí
TLS de borde	Sí
Micro máquina virtual dedicada	Sí
Regiones	América del Norte, Europa, América del Sur, Europa, Asia, Australia
Clientes destacados	Falco de Tailscale Cars.com

Nota. Fuente. Adaptado de: <https://blog.back4app.com/heroku-vs-fly-io/>

1.7.9 GitHub

GitHub es un servicio basado en la nube que aloja un sistema de control de versiones (VCS) llamado Git (GitHub Docs, 2023). Éste permite a los desarrolladores colaborar y realizar cambios en proyectos compartidos, a la vez que mantienen un seguimiento detallado de su progreso. Utilizado principalmente por ingenieros de software para hacer un seguimiento de las modificaciones realizadas en el código fuente, el sistema de control de versiones les permite analizar todos los cambios y revertirlos sin repercusiones si se comete un error (Hostinger, 2023). En la última década según Statista, el número de desarrolladores que utilizan la plataforma GitHub ha crecido de solo tres millones de usuarios en 2013 a 100 millones de usuarios en 2023 (Sujay, 2023). Además, en la página de Stack Overflow se menciona que es preferido por los desarrolladores con una aceptación de 56.04% y 70% por desarrolladores de Inteligencia Artificial (Stack Overflow, 2023).

1.7.10 Postman

Postman es una plataforma que facilita la realización de pruebas en APIs REST, ya sean propias o de terceros. Cuenta con una comunidad de 25 millones de desarrolladores (Johnston-Gilbert, 2023) y proporciona herramientas nativas para sistemas Windows, Mac y Linux. Postman permite la prueba y organización de APIs tanto para el frontend como para el backend, además de gestionar el ciclo de vida de las APIs, generar documentación y colaborar en entornos de desarrollo (Muradas, 2019). La aceptación del software de que tienen los desarrolladores es de 75% a nivel mundial (Postman, 2023).

Postman utiliza métodos como:

GET: Obtener información

POST: Agregar información

PUT: Reemplazar la información

PATCH: Actualizar alguna información

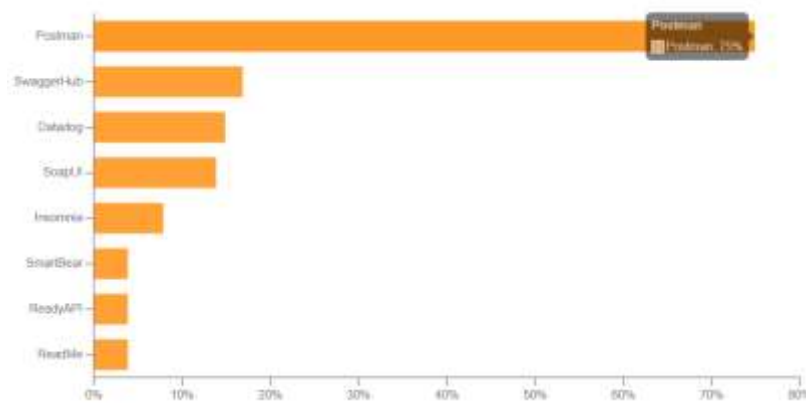
DELETE: Borrar información

De los métodos mencionados, si la respuesta dada se encuentra en el rango de “200” quiere decir que toda la petición ha salido sin inconvenientes; mientras que el rango de los códigos de

error “400” hacen referencia a errores con el cliente y aquellos errores en la línea de los “500” tienen que ver con fallos en el servidor (Muradas, 2019).

Figura 14

Herramientas preferidas por los desarrolladores de APIS



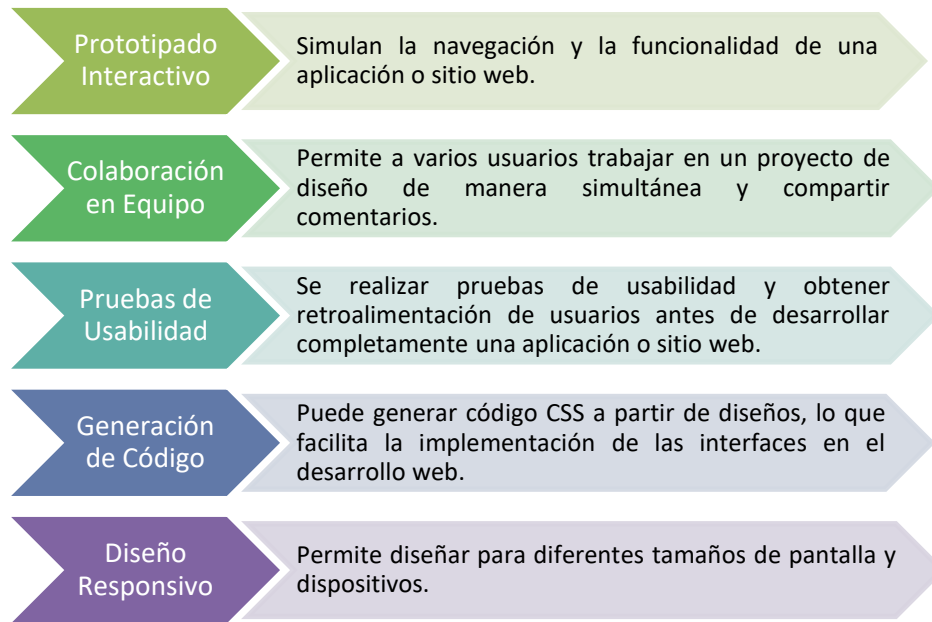
Nota. Fuente: (Postman, 2023).

1.7.11 Adobe XD

Es una herramienta de diseño vectorial de interfaz de usuario (UI) y experiencia de usuario (UX) que se utiliza para diseñar aplicaciones móviles y sitios web desde cero o importar desde otros programas. Tiene una gran cantidad de herramientas de interacción que permiten a los diseñadores crear prototipos dinámicos e incluso con la licencia premium se puede exportar a código fuente de HTML, animación y sonido (Rae, 2020). Adobe XD permite generar enlace para visualizar desde un celular o un computador los diseños dinámicos.

Figura 15

Características de adobe XD



Nota. Fuente: (Rae, 2020).

1.7.12 Software Estadístico Epidat

Epidat fue inicialmente concebido como una calculadora sencilla para satisfacer las necesidades de realizar consultas estadísticas y epidemiológicas básicas en un momento en el que no existían programas específicos a principios de los años noventa. A lo largo del tiempo, ha evolucionado desde versiones simples hasta la actual versión 4.2, migrando del lenguaje de programación Delphi a Java y siendo compatible con varios sistemas operativos como Windows, Linux y Macintosh. En la actualidad, Epidat cuenta con algoritmos, módulos y técnicas estadísticas que la convierten en una herramienta flexible y versátil (OPS-OMS & CES, 2016).

Figura 16

Software estadístico Epidat



Nota. Fuente: (OPS-OMS & CES, 2016).

1.7.13 Software para pruebas JMeter

Apache JMeter es una herramienta de software de código abierto en Java que permite realizar pruebas de carga, evaluar el comportamiento funcional y medir el rendimiento de aplicaciones. Es compatible con múltiples plataformas para analizar y mejorar el desempeño de sistemas y aplicaciones (Apache Software Foundation, 2023a). El software prueba todas las funciones que permite la grabación rápida del plan de prueba (desde navegadores o aplicaciones nativas), compilación y depuración.

Tiene la capacidad para cargar y probar el rendimiento de muchas aplicaciones/servidores/tipos de protocolos diferentes (Apache Software Foundation, 2023b):

- Web: HTTP, HTTPS (Java, NodeJS, PHP, ASP.NET, ...)
- Servicios web SOAP/REST
- FTP
- Base de datos a través de JDBC
- LDAP
- Middleware orientado a mensajes (MOM) a través de JMS
- Correo - SMTP(S), POP3(S) e IMAP(S)
- Comandos nativos o scripts de shell
- TCP
- Objetos Java

Figura 17

Versión de software JMeter para pruebas de rendimiento



Nota. Fuente: Adaptado de: <https://jmeter.apache.org/index.html>

1.7.14 Indicadores de herramientas de desarrollo

En la siguiente tabla se muestra los indicadores recopilados de las diferentes herramientas que se utilizarán en el proyecto. Las celdas que están marcadas se han seleccionado en base a la experiencia de los programadores en este proyecto. La ausencia de marcas en algunas celdas no implica que no cumplan con los requisitos técnicos del software; simplemente indica que, en este contexto, esas variables no se utilizaron o no se consideraron relevantes debido a los diferentes usos que se les dio.

Figura 18

Indicadores de las herramientas seleccionadas para el desarrollo

Indicadores	Aceptación	Eficiencia	Escalabilidad	Compatibilidad	Flexibilidad	Soporte y comunidad	Seguridad	Accesible
Herramientas								
Firestore	●	●	●	●		●	●	●
Android Studio	●	●		●		●		●
Node JS	●	●	●	●	●	●	●	●
Flutter	●	●	●	●	●	●		●
Postman	●	●		●			●	●
Visual Studio Code	●	●		●	●	●		●
Git Hub	●	●	●	●	●	●	●	●
Fly.io	●		●	●	●	●		●
Adobe XD	●	●	●	●	●	●		●
Epidat		●					●	●
JMeter	●	●	●		●	●		●

Nota. Elaboración propia.

1.7.15 Importancia de la Usabilidad

El diseño y la interfaz de usuario son elementos clave para el éxito de las aplicaciones móviles. Según Flores y Ceballos (2020), "la interfaz de usuario es el medio por el cual el usuario interactúa con la aplicación móvil y es fundamental para proporcionar una experiencia de usuario satisfactoria". La usabilidad, la accesibilidad y la estética son factores importantes que considerar en el diseño de la interfaz de usuario de una aplicación móvil, ya que influyen en la percepción del usuario sobre la calidad y la utilidad de la aplicación. Además, es importante tener en cuenta las limitaciones de los dispositivos móviles, como el tamaño de la pantalla y la capacidad de procesamiento, para asegurar que la aplicación sea eficiente y fácil de usar en cualquier dispositivo.

La norma ISO 25010 establece los criterios de calidad para la evaluación de sistemas y software, incluyendo el diseño y la interfaz de usuario. Según esta norma, la calidad de la interfaz de usuario se evalúa a través de los siguientes atributos:

- Usabilidad: La capacidad del sistema para ser fácilmente utilizado, aprendido y recordado por el usuario. Se evalúa la eficiencia, efectividad y satisfacción del usuario al interactuar con la interfaz.
- Estética visual: La percepción subjetiva del usuario sobre la apariencia visual del sistema. Se evalúa la consistencia y estética visual, la claridad de la información presentada, y la capacidad de personalización.
- Accesibilidad: La capacidad del sistema para ser utilizado por usuarios con discapacidades. En el caso de este proyecto se dará importancia la colorimetría que se enfocaría a personas con problemas para distinguir los colores.

En resumen, la norma ISO 25010 establece una serie de atributos que se deben tener en cuenta al diseñar y evaluar la interfaz de usuario de un sistema o software, con el objetivo de garantizar la calidad y satisfacción del usuario al interactuar con la interfaz.

En esta sección se detalla las recomendaciones para ubicar botones, textos, contenedores y más. En vista de que existen estudios y publicaciones que no todos usan las aplicaciones, se debe tomar la posición de los dedos y las recomendaciones que se dan para ello.

Tabla 16

Guía de usabilidad móvil

**Como las
utilizan su
dispositivo móvil**

Un estudio menciona que los usuarios tocan las pantallas o los botones de sus teléfonos sujetaron sus teléfonos de 6 formas básicas, pero el 75% usando el pulgar y menos del 50% sosteniendo el teléfono con una mano. Por lo tanto, la aplicación debe estar enfocada ser manejable con una sola mano, en especial con el dedo pulgar.



**Movimientos
oculares**

Las aplicaciones móviles con las aplicaciones web no comparten el mismo patrón, Para aplicaciones web se usa el efecto “F”. Pero para los usuarios de dispositivos móviles no; Ya que tienden a mirar primero al centro de la pantalla.

- El centro también representa el lugar más fácil para que puedan alcanzar y tocar con los pulgares en la mayoría de los teléfonos inteligentes.
- La precisión del tacto disminuye a medida que los usuarios alcanzan las esquinas más externas de la pantalla.



**Comprender el
contexto de uso**

Tomar en cuenta que las diferentes aplicaciones móviles tendrán diferentes contextos y lugares donde se usarán. Si es para entretenimiento, trabajar, hacer ejercicio o conducir.

La aplicación de preferencia debe contar con un diseño de una sola columna en el móvil que múltiples columnas.

Sigue la regla de oro (s)



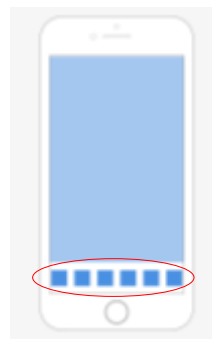
La aplicación debe contar con uno de los principales principios del diseño industrial, que el contenido siempre debe aparecer por encima de los controles para que las manos de los usuarios no bloqueen su línea de visión.

Pon el contenido primero



El objetivo táctil óptimo es de 7x7mm, lo que se traduce en píxeles equivale a aproximadamente 40. Pero para tener en cuenta varios tamaños de visualización, 44 píxeles es el espacio ideal para cubrir las zonas táctiles y evitar el error del usuario.

Manténgase en el objetivo (táctil)



Nota. Se detalla algunas recomendaciones que la aplicación móvil debe cumplir para cubrir las necesidades.
Fuente: adaptado de: (Esoldo Christine, 2023a; Hooper, 2017).

1.7.16 Trabajos relacionados

En este subtema se abordará los trabajos relacionados que sirven de apoyo para el desarrollo, haciendo tres preguntas de análisis; “¿Qué se hizo?, ¿Cómo se hizo? y ¿Qué resultados se obtuvieron?”, Cada estudio contiene la referencia enumerada, con tres párrafos respectivamente a las preguntas planteadas y en la parte final se presenta una tabla con el análisis de los indicadores.

REFERENCIA 1: Una tesis que sirve de ejemplo es la de Christian Javier Acurio Acurio de la Universidad Politécnica Salesiana de Quito en el 2017, con el título “**Desarrollo de una aplicación móvil que envíe una solicitud de requerimiento de gas doméstico a los distribuidores o a los vehículos de distribución más cercanos a una ubicación en la ciudad de Quito mediante georreferenciación**”.

El proyecto se enfocó en los ciudadanos de la ciudad de Quito que tengan dispositivos móviles Android. En términos técnicos, la aplicación móvil funciona a través de un web services, el cual se encuentra dentro de una máquina virtual en la plataforma de Windows Azure. Esta aloja el servidor XAMPP que contiene la base de datos Mysql y los archivos PHP, los cuales permiten a la aplicación móvil interactuar con la base de datos de forma remota. (Acurio, 2017) Todo esto se realizó bajo el marco de trabajo XP.

Para obtener los resultados, se realizaron pruebas de funcionamiento en dos dispositivos móviles Android y el software JMeter. Se realizó pruebas de rendimiento y pruebas de estrés; en el inicio de sesión, envío de coordenadas y consulta de coordenadas. Esto les permitió conocer el límite de peticiones de los módulos correspondientes. Para corregir que el distribuidor se pueda visualizar en tiempo real, se aplicó el método `requestLocationUpdates()`, obteniendo mejores resultados.

REFERENCIA 2: Este trabajo de titulación realizado por Frank Eras Camacho de la Escuela Politécnica Nacional de Quito en el 2020, con el título “**Aplicación móvil en Android para la adquisición de gas licuado de petróleo (GLP) doméstico**”. Utiliza componentes y características de geolocalización en un radio más cercano para que los distribuidores reciban las solicitudes mediante notificaciones en la ciudad de Quito.

Se desarrolla utilizando la plataforma Android estudio, el lenguaje Java, los servicios de bases de datos Firebase, las APIs de Google para la geolocalización y es exclusivamente para el SO Android. Para ubicar al distribuidor más cercano y realizar el pedido, las rutas se grafican con

la función DATA_PARSER incluida en los servicios de Google. La programación realiza los métodos privados, clases, herencias y encapsulamiento distintivo de Java.

Los resultados de la funcionalidad se detallan en una tabla describiendo el escenario, la prueba y el criterio de aceptación. Para las pruebas de rendimiento se utilizó la herramienta Android Profiler, en donde se analizó en tiempo real el consumo de memoria, red y batería. Las pruebas se realizaron mediante una emulación de usuarios y distribuidores. (Eras, 2020)

REFERENCIA 3: Este artículo realizado por Faouzi Kamouna y otros. Presentado en “**The eleventh International Conference on Ambient Systems, Networks and Technologies (ANT)**”, Varsovia, Polonia, el 9 de Abril de 2020, con el título “**A geolocation-aware mobile crowdsourcing solution for the emergency supply of oxygen cylinders**”. Esta solución móvil ayuda a que los pacientes que sufren de dificultades respiratorias repentinas puedan solicitar el suministro de emergencia de cilindros de oxígeno; mediante una aplicación móvil, que permite conectar con proveedores y entre pacientes que tengan dificultades para respirar.

Para el análisis y diseño, se utilizan la técnica Design Thinking que permite crear prototipos, probar y mejorar. Para obtener los requisitos funcionales del sistema se realiza el modelo de casos de uso UML. Se desarrolló en el SDK Android estudio, incluido la biblioteca Volley que facilita y agiliza el uso de redes en aplicaciones Android. Los datos, solicitudes de emergencia y notificaciones se manejan con Firebase Cloud Messaging (FCM). La arquitectura consta de cuatro capas: Data Acces Layer, Busines Layer, Service Layer e Interfaz; todo este conjunto lógico interactúa mediante RESTful. Finalmente, la aplicación interpreta los datos obtenidos del cilindro mediante el sistema de monitoreo bluetooth y visión artificial para analizar el nivel de oxígeno en el cilindro mediante imágenes.

Los resultados reflejan en una interfaz donde el usuario ubica a los proveedores o los usuarios cercanos que disponen de oxígeno y se realiza una llamada para acordar los detalles de obtención del cilindro. Para garantizar el grado de usabilidad se aplica la ISO 9241-210:2010, y la ISO/IEC 25022:2016 para medir la eficiencia, eficacia y satisfacción del usuario. Las pruebas obtenidas fueron realizadas con un equipo de investigación, voluntarios y participantes exitosamente.

REFERENCIA 4: Otro proyecto similar es de la **Agencia de Regulación y Control Hidrocarburo (ARCH)** que operan desde enero del 2020 en las fronteras norte y sur del Ecuador, para evitar la salida del hidrocarburo del país.

Se trata de una aplicación que permite a los distribuidores de gas verificar los datos del comprador desde su teléfono celular. Entre otras cosas, el nuevo dispositivo que se activa escaneando la cédula de los ciudadanos de las localidades fronterizas ayuda a verificar en el teléfono móvil cuántos cilindros pueden adquirir en el mes, a partir del número de miembros de una familia. También está enlazado a la base de datos del Registro Civil y se actualiza, por ejemplo, cuando fallece una persona (El Comercio, 2020).

De esa manera se controla un porcentaje para que el hidrocarburo no salga de manera ilegal del país, la aplicación no debe estar conectada al internet permanentemente, pero al final del día esos datos se envían a la agencia de hidrocarburos (El Comercio, 2020).

REFERENCIA 5: Trabajo de titulación (Complexivo) realizado por Guerrón Paspuel Álvaro Marcelo de la Universidad Regional Autónoma De Los Andes (UNIANDES), con su sucursal en Ibarra, en el 2018, con el título “**Alternativa tecnológica con geoposicionamiento y rastreo satelital para reducir la contaminación acústica producida por parte de los distribuidores de gas doméstico de la ciudad de Ibarra**”. El estudio fundamenta bibliográficamente los procesos de geoposicionamiento y rastreo satelital y desarrolla el prototipo móvil que permitirá reducir la contaminación acústica producida por los distribuidores de gas doméstico en la ciudad de Ibarra.

La metodología empleada para el marco de trabajo es XP; para obtener los datos reales se realizó charlas, pruebas de campo, encuestas de satisfacción a los usuarios con la finalidad de identificar las necesidades y el nivel de contaminación sonora; para el diseño y modelado se utiliza diagramas de casos de uso en PowerDesigner; el BackEnd se desarrolla en Python; el Frontend en Django para móvil e Ionic para web; para la geolocalización y la base de datos (Firebase) se desarrolla con los servicios de Google. Para las pruebas de rendimiento se utiliza el software JMeter y para el funcionamiento se prueba en dispositivos con Android 5.1.1 y 6.0.1 conectados a wifi o datos móviles.

Los resultados de la investigación se validaron con pruebas de campo, encuestas y conversatorios de los ciudadanos del sector Alpachaca, San Francisco, Sagrario, Priorato, Caranqui; obteniendo un nivel mayor a 80% de aceptación (Guerrón Alvaro, 2018).

REFERENCIA 6: Este artículo científico realizado por Paula Voorheis en la Universidad de Toronto, marzo de 2022, con el título de “**Integrating Behavioral Science and Design Thinking to Develop Mobile Health Interventions: Systematic Scoping Review**”. Este estudio

describe el proceso de diseño en aplicaciones móviles de la salud (mHealth) y la importancia de aplicar metodologías, teorías, modelos, marcos o técnicas. Los trabajos combinan a Behavioral Design (BD) y Design Thinking (DT) que son reconocidos como las mejores prácticas en sus áreas correspondientes.

Par definir el alcance del estudio, se realiza una recopilación de información mediante un manual del instituto Joanna Briggs que incluye teorías, modelos y trabajos citados. Además, se involucra al personal interesado (Ingenieros de software, Arquitectos de software), diseñadores de sanidad móvil, entre otros para realizar la selección de artículos útiles de un total de 1012, pasando a reducir a 75 que cumplieron los criterios de elegibilidad. Luego se integran entre BD y DT con los principios congruentes que son: empatizar, idear, prototipar y probar. Este enfoque se presenta en un diagrama con tres aspectos principales, Identificación de la efectividad del diseño en aplicaciones móviles, los métodos que se usaron.

En un total de 75 artículos entre el 2012 - 2021 cumplieron con el criterio de inclusión. Los estudios demuestran que integraron tanto BD como DT y siguieron los pasos mencionados que es empatizar con el cliente, definir los requisitos de usuario, idear características, crear un prototipo y probar la solución. El reto a los que se enfrentaron principalmente fue a las implicaciones de los socios y pacientes, pero esto dio resultados positivos en las mejoras y diseños de interfaces amigables para los usuarios.

1.6.17 Indicadores de trabajos relacionados

Tabla 17

Indicadores de contenido en los trabajos relacionados

Indicadores	ISO/IEC C Calidad	Marco de trabajo Ágil	ISO/IEC / IEEE Requi- sitos	NoS QL	Geolo- Caliza- ción	SO Android	Api REST	SLR	UI/U X	Produ- cción
Trabajos relacionados										
REFERENCIA 1		●		●	●	●	●			
REFERENCIA 2		●		●	●	●				
REFERENCIA 3	●	●		●	●	●	●		●	
REFERENCIA 4	●				●	●				●
REFERENCIA 5		●		●	●	●	●			
REFERENCIA 6									●	

Nota. Elaboración propia.

CAPÍTULO 2

2 Documento de Especificación de Requerimientos del Software (SRS – ISO/IEEE 29148)

2.1 Introducción

2.1.1 *Propósito*

El propósito del desarrollo está enfocado en una alternativa tecnológica para solicitar el servicio de cilindros de gas doméstico mediante una aplicación móvil, en donde los autores principales son el usuario y el distribuidor. La aplicación cuenta con localización GPS para obtener una ubicación exacta del usuario y que el distribuidor pueda cumplir con la entrega, será desarrollado exclusivamente para los usuarios de la ciudad de Ibarra.

2.1.2 *Visión General del Producto*

El aplicativo móvil, tiene como objetivo facilitar la compra y venta de gas a domicilio, tanto para los ciudadanos como para los distribuidores. Los ciudadanos podrán solicitar un cilindro de gas cuando lo necesiten, sin tener que esperar en las calles o depender de un horario fijo. Los distribuidores podrán optimizar sus rutas de entrega, conociendo dónde y cuándo se requiere el servicio de abastecimiento de gas. De esta manera, se pretende reducir los costos de operación, el tiempo de espera y el ruido ambiental causado por los altoparlantes de los camiones.

2.1.3 *Perspectiva del producto*

La aplicación está diseñada para trabajar en dispositivos móviles, esto permite a los usuarios tener acceso fácil desde cualquier lugar en donde se encuentre conectado al internet. Además, esta aplicación está diseñada con el sistema operativo Android que es el más usado en la localidad. Se estima que los usuarios acepten como una mejor alternativa.

2.1.4 *Funciones del Producto*

La funcionalidad en general es permitir al usuario solicitar gas doméstico desde su hogar en el horario establecido por los camiones distribuidores. Para los distribuidores es poder recibir notificaciones de pedidos más cercanos y acudir a la entrega guiados por rutas marcadas en el GPS.

2.1.4.1 **Usuario:**

- Permitir el acceso a través de la aplicación móvil mediante autenticación en línea, identificando si el usuario es Distribuidor o Cliente.

2.1.4.2 Cliente:

- Permitir a los clientes realizar pedidos de gas a través de la aplicación, especificando la cantidad de gas requerida y la dirección de entrega.
- Permitir a los clientes hacer seguimiento del estado de su pedido, desde la confirmación hasta la entrega.

2.1.4.3 Distribuidor:

- Permitir a los distribuidores visualizar historial de sus ventas, incluyendo información sobre la cantidad de pedidos recibidos, entregados.
- Permitir a los distribuidores recibir y aceptar los pedidos de gas que los clientes realizan a través de la aplicación. La aplicación debe asignar automáticamente los pedidos a los distribuidores disponibles.
- Permitir a los distribuidores recibir alertas en tiempo real cuando se les asigna un nuevo pedido, para que puedan responder y aceptar el pedido de inmediato.

2.1.5 Características del usuarios y limitaciones.

Tabla 18

Características de los usuarios y limitaciones

Usuario	Descripción	Limitaciones
Cliente	Los clientes son usuarios que necesitan comprar y recibir cilindros de gas para uso doméstico. Podrían ser personas que viven en una casa o apartamento, dueños de pequeños negocios, restaurantes, entre otros.	Los clientes solo podrán programar pedidos dentro de las horas de servicio establecidas por los distribuidores. Los clientes solo podrán realizar pagos en efectivo.
Distribuidor	Los distribuidores, individuos encargados de la entrega de gas doméstico a los clientes.	Los distribuidores solo podrán aceptar pedidos cuando se encuentren conectados a internet.

2.2 Requerimientos específicos

2.2.1 Interfaz Externa

Se desarrolla una interfaz de usuario utilizando Flutter para crear una aplicación móvil que sea compatible con dispositivos de diferentes tamaños de pantalla y sistemas operativos. El objetivo perseguido por la interfaz es lograr un entorno amigable, claro e intuitivo para el usuario, adaptado específicamente a la experiencia móvil.

Al aprovechar las capacidades de diseño adaptable y responsive design de Flutter, se garantiza que la aplicación se vea y funcione de manera óptima en una variedad de dispositivos móviles, como teléfonos inteligentes y tabletas. Se pone especial énfasis en la experiencia del usuario en pantallas táctiles y en aprovechar las interacciones y gestos nativos de los dispositivos móviles.

Asimismo, se utilizan los widgets y componentes predefinidos de Flutter para construir una interfaz de usuario coherente y estéticamente agradable. Esto implica el uso de botones, formularios, menús desplegados y otros elementos interactivos comunes en las aplicaciones móviles. El objetivo es proporcionar una experiencia fluida y consistente a los usuarios, logrando que la aplicación se sienta nativa en cada plataforma móvil, ya sea iOS o Android. También se trabaja en optimizar el rendimiento de la aplicación para que sea rápida y receptiva, incluso en dispositivos móviles con recursos limitados.

2.2.2 Funciones

Para el desarrollo de la aplicación, se llevaron a cabo encuestas dirigidas tanto a los distribuidores de la ciudad de Ibarra como a los residentes de dicha localidad, teniendo en cuenta la muestra obtenida. Estas encuestas permitieron identificar las necesidades que deberá abordar la aplicación. A continuación, se presentan dichas necesidades en las tablas de historias de usuarios.

2.2.2.1 Requisitos funcionales

2.2.2.1.1 Historias de usuarios

Figura 19

Historia de Usuario 1 - Inicio de sesión

Historia de Usuario	
Número: 1	Usuario: Cliente/Distribuidor
Nombre historia: Inicio de sesión	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 2	Interacción asignada: 1
Programador responsable: Alexander Quinatoa, Diego Carlosama	
Descripción: Los usuarios, ya sean distribuidores o clientes, deberán tener la capacidad de iniciar sesión en la aplicación móvil introduciendo su número de teléfono. A continuación, recibirán un código de verificación para acceder a la interfaz principal, que se ajustará según su rol registrado.	
Observaciones: <ul style="list-style-type: none">Garantizar la aplicación de medidas de seguridad en la gestión de autenticación, como el almacenamiento seguro de contraseñas.	

Nota. Elaboración propia.

Figura 20

Historia de Usuario 2 - Registrarse

Historia de Usuario	
Número: 2	Usuario: Cliente/Distribuidor
Nombre historia: Registrarse	
Prioridad en negocio: Medio	Riesgo en desarrollo: Media
Puntos estimados: 2	Interacción asignada: 1
Programador responsable: Alexander Quinatoa, Diego Carlosama	
Descripción:	

Los usuarios deberán ser capaces de registrarse de forma sencilla en la aplicación móvil. El proceso de registro iniciará con la verificación del número de teléfono, seguido de la introducción de información personal, como nombre completo, dirección de correo electrónico (opcional), foto (opcional) y elección de su rol en la aplicación.
<p>Observaciones</p> <ul style="list-style-type: none"> • Considerar la inclusión de validaciones en los campos de entrada para asegurar la calidad de los datos ingresados. • Implementar una política de privacidad y términos de uso que los usuarios deban aceptar durante el proceso de registro.

Nota. Elaboración propia.

Figura 21

Historia de Usuario 3 - Actualización de información para distribuidores

Historia de Usuario	
Número: 3	Usuario: Distribuidor
Nombre historia: Actualización de información para distribuidores	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 3	Interacción asignada: 2
Descripción: La aplicación solicitará a los distribuidores completar información básica para verificar su elegibilidad. Esto incluirá la fecha de nacimiento, número de licencia de conducir, número de cédula de identidad y detalles relacionados con su vehículo. Una vez completada esta etapa, podrán ofrecer sus servicios en línea y recibir notificaciones de pedidos.	
<p>Observaciones:</p> <ul style="list-style-type: none"> • Asegurarse de que los datos personales y documentos se almacenen y transmitan de forma segura. 	

Nota. Elaboración propia.

Figura 22*Historia de Usuario 4 - Recepción de solicitud de pedido*

Historia de Usuario	
Número: 4	Usuario: Distribuidor
Nombre historia: Recepción de solicitud de pedido	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 3	Interacción asignada: 2
Programador responsable: Alexander Quinatoa, Diego Carlosama	
Descripción: La aplicación permitirá a los distribuidores recibir solicitudes de los clientes en tiempo real, proporcionando información relevante para atenderlas eficientemente. Cuando un proveedor de servicios acepte una solicitud, se proporcionará una ruta en el mapa con tiempo estimado y distancia en kilómetros. Los clientes serán notificados automáticamente cuando su solicitud sea aceptada. La solicitud se cerrará cuando el distribuidor llegue al destino y notifique al cliente.	
Observaciones: <ul style="list-style-type: none"> • Asegurar una comunicación en tiempo real y confiable entre clientes y distribuidores. • Implementar una funcionalidad de seguimiento de ubicación para garantizar la seguridad y eficiencia de las entregas. 	

Nota: Elaboración propia**Figura 23***Historia de Usuario 5 - Visualizar historial de pedidos*

Historia de Usuario	
Número: 5	Usuario: Distribuidor/Cliente
Nombre historia: Visualizar historial de pedidos	
Prioridad en negocio: Medio	Riesgo en desarrollo: Baja
Puntos estimados: 2	Interacción asignada: 1
Programador responsable: Alexander Quinatoa, Diego Carlosama	

Descripción: La aplicación móvil ofrecerá a los distribuidores y/o clientes la funcionalidad de visualizar un historial completo de los pedidos. Esto proporcionará a los usuarios un registro detallado de todas las solicitudes realizadas a lo largo del tiempo.
Observaciones:

Nota. Elaboración propia.

Figura 24

Historia de Usuario 6 - Realizar pedido

Historia de Usuario	
Número: 6	Usuario: Cliente
Nombre historia: Realizar pedido	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 3	Interacción asignada: 1
Programador responsable: Alexander Quinatoa, Diego Carlosama	
Descripción: La aplicación permitirá a los clientes solicitar servicios de distribución de gas doméstico en tiempo real. Proporcionarán información necesaria para que los distribuidores acepten las solicitudes, incluyendo la visualización en tiempo real del tiempo estimado de llegada del distribuidor.	
Observaciones:	

Nota. Elaboración propia.

Figura 25

Historia de Usuario 7 – Ingreso Manual de Ubicación

Historia de Usuario	
Número: 7	Usuario: Cliente
Nombre historia: Ingreso manual de ubicación	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 3	Interacción asignada: 4
Programador responsable: Alexander Quinatoa, Diego Carlosama	

<p>Descripción: La aplicación brindará a los clientes la capacidad de definir manualmente una ubicación específica para su pedido mediante una búsqueda interactiva en el mapa. Esto garantizará que, al realizar un pedido, se considere la nueva posición elegida por el cliente en lugar de la ubicación predeterminada.</p>
<p>Observaciones:</p> <ul style="list-style-type: none"> • Implementar una función de búsqueda en el mapa que sea intuitiva y fácil de usar para los clientes. • Asegurarse de que la nueva ubicación seleccionada por el cliente se integre de manera efectiva en el proceso de pedidos.

Figura 26

Historia de Usuario 8 – Gestión de Información Personal

Historia de Usuario	
Número: 7	Usuario: Cliente/distribuidor
Nombre historia: Gestión de Información Personal	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 3	Interacción asignada: 4
Programador responsable: Alexander Quinatoa, Diego Carlosama	
<p>Descripción: Se centra en la administración de la información personal por parte de los usuarios, ya sean clientes o distribuidores, en la aplicación móvil. Los usuarios podrán actualizar, modificar y gestionar su información personal, lo que incluye datos como nombre, dirección de correo electrónico (opcional), foto (opcional), número de teléfono. La gestión de información personal es esencial para mantener los perfiles de usuario actualizados y garantizar una experiencia fluida en la aplicación.</p>	
Observaciones:	

2.2.2.2 Requerimientos no funcionales

Los requerimientos no funciones de aplicativo móvil para la distribución de gas doméstico se muestra en la siguiente tabla.

Tabla 19*Requerimientos no funcionales*

Requerimientos No Funcionales	
Requerimiento No Funcional	Descripción Requerimiento No Funcional
Desempeño	Garantizar un tiempo de respuesta rápido para las solicitudes de los usuarios al servidor.
Escalabilidad	Debe ser capaz de agregar fácilmente nuevos módulos en caso de necesidad futura.
Disponibilidad	La aplicación móvil debe ser accesible a través de una conexión a internet en cualquier momento, garantizando su disponibilidad constante.
Confiabilidad	Asegurar que la información sea consistente y coherente en todo el sistema, promoviendo la confiabilidad de los datos.
Tolerancia a Fallos	Información resguardada en el caso de fallas del servidor o del sistema.
Mantenibilidad	Las actualizaciones del software, librerías, ambiente, etc., deben realizarse de manera rápida y transparente para el usuario, garantizando la facilidad de mantenimiento y mejora continua del sistema.

Nota. Elaboración propia.

CAPITULO 3

3 DESARROLLO DEL APLICATIVO MOVIL APLICANDO LA METODOLOGÍA XP

Bajo el marco de trabajo XP, se procederá con el desarrollo de la aplicación móvil siguiendo un proceso estructurado. Comenzando con la etapa de planificación, se revisarán detenidamente los requisitos previamente establecidos en el capítulo 2. Luego, en la fase de diseño, se definirá la arquitectura y se elaborarán las especificaciones técnicas.

La etapa de codificación seguirá, donde se implementarán las funcionalidades siguiendo las mejores prácticas de desarrollo. Por último, se llevarán a cabo pruebas exhaustivas para garantizar la funcionalidad y calidad del aplicativo. Este enfoque paso a paso asegura un desarrollo metódico y eficiente de la aplicación.

3.1 Fases de la metodología XP

3.1.1 Planificación

3.1.1.1 Módulos del proyecto

Tabla 20

Módulos del proyecto

Modulo	Descripción
Autenticación	La actividad principal de este módulo es permitir que los usuarios inicien sesión en la aplicación móvil de distribución de gas doméstico y verifiquen su identidad. Los usuarios pueden crear una cuenta nueva o iniciar sesión con una cuenta existente.
Clientes	Este módulo permite a los usuarios solicitar el servicio de distribución de gas doméstico a través de la aplicación móvil. Los clientes pueden seleccionar la cantidad de cilindros de gas que desean y proporcionar información de ubicación para la entrega.
Vendedores	Este módulo permite que los conductores se registren en la aplicación móvil de distribución de gas doméstico proporcionando información básica de conductor. Una vez completado el registro, los conductores pueden recibir notificaciones de nuevas solicitudes de servicio y la ubicación de los clientes que realizan la petición.

Gestión de Información

Este módulo proporciona una manera fácil de almacenar y acceder a datos históricos de clientes y distribuidores. Los usuarios pueden acceder a información detallada sobre pedidos anteriores, como el historial de los pedidos.

Administración y ajustes

Este módulo permite a los usuarios revisar su historial, configurar sus preferencias (como cambiar su número), consultar información acerca de la aplicación, cambiar su rol, acceder a información de seguridad y protección con números de emergencia para policía y bomberos competentes en la ciudad de Ibarra, encontrar guías de ayuda para interactuar con la aplicación móvil y cerrar sesión cuando sea necesario.

3.1.1.2 Planificación de iteraciones

A continuación, se presenta la planificación de las iteraciones del proyecto de la aplicación móvil para gas doméstico.

Tabla 21

Planificación de iteraciones

Nro.	Nombre	Puntos Estimados	Prioridad	Riesgo	Número de iteración
1	Inicio de sesión	2	Alta	Medio	1
2	Registrarse	2	Medio	Medio	1
3	Realizar pedido	3	Alta	Medio	3
4	Actualización de información para distribuidores	3	Alta	Medio	2
5	Recepción de solicitud de pedido	3	Alta	Medio	3
6	Visualizar historial de ventas y desempeño	2	Medio	Baja	3
7	Ingreso manual de ubicación	3	Alta	Medio	4
8	Gestión de información personal	3	Medio	Medio	4

3.1.1.3 Tareas por iteración

HU 1: Inicio de sesión

En las siguientes tablas se describen todas las tareas correspondientes a la HU 1 – Inicio de sesión.

Tabla 22

Tarea 1.1 – Generación y envío de código de verificación

Tarea	
Nro. de tarea: 1.1	Nro. Historia de usuario: 1
Nombre de la tarea: Generación y envío de código de verificación	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 3
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción:	
La actividad implica enviar un código de verificación al número de teléfono registrado en la base de datos. El código será válido por 5 minutos y se utiliza para confirmar la identidad del usuario.	

Tabla 23

Tarea 1.2 – Verificación y reenvío de códigos

Tarea	
Nro. de tarea: 1.2	Nro. Historia de usuario: 1
Nombre de la tarea: Verificación y reenvío de códigos	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 2
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: La tarea implica verificar la precisión y validez del código proporcionado por el usuario y, en caso de ser necesario, generar y enviar un nuevo código para garantizar una verificación exitosa. Esto asegura la autenticidad de la información y permite al usuario acceder a las funcionalidades o servicios correspondientes.	

HU 2: Registrarse

En las siguientes tablas se describen todas las tareas correspondientes a la HU 2 – Registrarse.

Tabla 24

Tarea 2.1 – Reutilización de componentes para el inicio de sesión

Tarea	
Nro. de tarea: 2.1	Nro. Historia de usuario: 2
Nombre de la tarea: Reutilización de componentes para el inicio de sesión en el proceso de registro.	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 2
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: Esta tarea implica reutilizar los componentes utilizados en el inicio de sesión para guiar al usuario, en caso de que no esté registrado, a través de todo el proceso necesario para proporcionar la información personal requerida y así poder utilizar la aplicación móvil.	

Tabla 25

Tarea 2.2 Almacenamiento seguro de la información personal

Tarea	
Nro. de tarea: 2.2	Nro. Historia de usuario: 2
Nombre de la tarea: Almacenamiento seguro de toda la información personal.	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 3
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: La tarea implica recopilar la información personal necesaria, para que los usuarios se integren en la aplicación móvil y su posterior almacenamiento en la base de datos correspondiente. Esta información es vital para permitir el acceso y uso adecuado de todas las funcionalidades ofrecidas por la aplicación, garantizando así una experiencia óptima y segura para los usuarios.	

Tabla 26*Tarea 2.3 – Identificación del rol y enrutamiento con la interfaz*

Tarea	
Nro. de tarea: 2.3	Nro. Historia de usuario: 2
Nombre de la tarea: Identificación del rol del usuario y enrutamiento a la interfaz correspondiente	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 3
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: La tarea implica la identificación del rol específico, extraído de la base de datos, y en función de dicho rol, dirigirse a la interfaz correspondiente para utilizar las funcionalidades pertinentes. Este proceso permite una experiencia personalizada y eficiente para cada usuario, asegurando que puedan acceder y aprovechar las características adecuadas según su perfil y necesidades.	

HU 3: Realizar pedido (cliente)

En las siguientes tablas se describen todas las tareas correspondientes a la HU 3 – Realizar pedido.

Tabla 27*Tarea 3.1 – Realizar pedido a los distribuidores*

Tarea	
Nro. de tarea: 3.1	Nro. Historia de usuario: 3
Nombre de la tarea: Implementar la funcionalidad de envío de solicitudes a los distribuidores	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 9
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: En esta tarea, se llevará a cabo la configuración de la lógica necesaria para enviar en tiempo real las solicitudes de los clientes a los distribuidores disponibles.	

Tabla 28*Tarea 3.2 – Cancelar Pedido*

Tarea	
Nro. de tarea: 3.2	Nro. Historia de usuario: 3

Nombre de la tarea: Cancelación de pedido

Tipo de tarea: Desarrollo

Tiempo estimado en horas: 7

Responsable: Diego Carlosama y Alexander Quinatoa

Esta tarea implica la implementación de la funcionalidad de cancelación de pedidos que han sido asignados a los distribuidores. Además, se asegurará de que la lista de pedidos de los distribuidores se actualice en tiempo real para reflejar la cancelación. Esta característica es esencial para brindar flexibilidad a los clientes y garantizar que los distribuidores tengan información precisa y actualizada sobre sus pedidos en curso.

Tabla 29

Tarea 3.3 – Graficar ruta del cliente a distribuidor

Tarea	
Nro. de tarea: 3.3	Nro. Historia de usuario: 3
Nombre de la tarea: Graficar ruta de cliente a distribuidor	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 9
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: Esta tarea engloba la captura de las ubicaciones tanto del distribuidor como del cliente, y esto se lleva a cabo tan pronto como el distribuidor acepta el pedido. Luego, se procede a representar gráficamente la ruta en un mapa. Además, se proporciona una estimación del tiempo necesario para que el distribuidor complete el recorrido y de la distancia en kilómetros que deberá cubrir. Esta funcionalidad brinda a los clientes una perspectiva visual de la ruta del distribuidor, mejorando la transparencia en el proceso y permitiendo una mejor planificación de la entrega.	

Tabla 30

Tarea 3.4 – Notificación de aceptación de pedido del distribuidor

Tarea	
Nro. de tarea: 3.4	Nro. Historia de usuario: 3
Nombre de la tarea: Configurar notificaciones en tiempo real para informar al cliente sobre la aceptación del pedido del distribuidor	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 8

Responsable: Diego Carlosama y Alexander Quinatoa

Descripción: En esta tarea se llevará a cabo la implementación de un sistema de notificaciones en tiempo real que enviará mensajes automáticos al cliente una vez que el distribuidor haya aceptado la solicitud de servicio y esté en camino hacia la ubicación del cliente. Esta notificación proporcionará al cliente información relevante sobre la aceptación del pedido, brindando claridad y mejorando la experiencia del usuario.

Tabla 31

Tarea 3.5 – Notificación de llegada del distribuidor

Tarea	
Nro. de tarea: 3.5	Nro. Historia de usuario: 3
Nombre de la tarea: Configurar notificaciones en tiempo real para informar al cliente sobre la llegada del distribuidor	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 8
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: Esta tarea implica la implementación de un sistema de notificaciones en tiempo real que enviará mensajes automáticos al cliente tan pronto como el distribuidor llegue a su destino. La notificación proporcionará al cliente la información relevante sobre la llegada, mejorando la experiencia y brindando transparencia en el proceso de entrega.	

HU 4: Actualización de información para distribuidores

En las siguientes tablas se describen todas las tareas correspondientes a la HU 4 – Actualización de información para distribuidores.

Tabla 32

Tarea 4.1 – Recopilar información del distribuidor

Tarea	
Nro. de tarea: 4.1	Nro. Historia de usuario: 4
Nombre de la tarea: Capturar información personal del distribuidor	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 5
Responsable: Diego Carlosama y Alexander Quinatoa	

Descripción: Esta tarea consiste en obtener la información básica del distribuidor, incluyendo su fecha de nacimiento, número de licencia de conducir y número de cédula de identidad. Además, se debe implementar la funcionalidad necesaria para validar y verificar los datos ingresados por el distribuidor, garantizando la precisión y consistencia de la información registrada.

Tabla 33

Tarea 4.2 Recopilar información del vehículo

Tarea

Nro. de tarea: 4.2

Nro. Historia de usuario: 4

Nombre de la tarea: Capturar información del vehículo

Tipo de tarea: Desarrollo

Tiempo estimado en horas: 4

Responsable: Diego Carlosama y Alexander Quinatoa

Descripción: Esta tarea consiste en obtener la información de los distribuidores relacionados con su vehículo, como el número de placa, una foto del camión y la tarjeta de identificación emitida por ENI. Asimismo, se debe implementar la funcionalidad necesaria para validar y verificar los datos ingresados del vehículo, asegurando la exactitud y coherencia de la información registrada.

HU 5: Recepción de solicitud de pedido

En las siguientes tablas se describen todas las tareas correspondientes a la HU 5 – Recepción de solicitud de pedido.

Tabla 34

Tarea 5.1 – Recepción de pedidos

Tarea

Nro. de tarea: 5.1

Nro. Historia de usuario: 5

Nombre de la tarea: Implementar la recepción de solicitudes en tiempo real

Tipo de tarea: Desarrollo

Tiempo estimado en horas: 5

Responsable: Diego Carlosama y Alexander Quinatoa

Descripción: Esta tarea consiste en realizar la funcionalidad que permita recibir y mostrar en tiempo real las solicitudes de los clientes en la aplicación de proveedores de servicios. Además, se debe establecer una comunicación fluida entre el cliente y el proveedor para recibir y transmitir la información necesaria con el fin de atender la solicitud de manera eficiente. Esto implica desarrollar mecanismos que permitan la recepción y visualización de las solicitudes, así como la implementación de canales de comunicación efectivos para intercambiar información relevante entre ambas partes.

Tabla 35

Tarea 5.2 – Aceptar pedido

Tarea	
Nro. de tarea: 5.2	Nro. Historia de usuario: 5
Nombre de la tarea: Notificar al cliente sobre la aceptación de la solicitud	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 6
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: Esta tarea implica la aceptación del pedido por parte del distribuidor y la notificación en tiempo real tanto al cliente, informándole que su pedido ha sido asignado al distribuidor que lo aceptó. La comunicación en tiempo real es esencial para brindar información actualizada y mantener al cliente al tanto de la aceptación del pedido por parte del distribuidor, lo que mejora la experiencia del usuario.	

Tabla 36

Tarea 5.3 – Graficar ruta de distribuidor a cliente

Tarea	
Nro. de tarea: 5.3	Nro. Historia de usuario: 5
Nombre de la tarea: Implementar la visualización de la ruta	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 5
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: Esta tarea consiste en integrar un servicio de mapas que permita trazar la ruta desde la ubicación actual del distribuidor hasta el destino solicitado por el cliente. Se debe mostrar en el mapa la ruta trazada, proporcionando al proveedor información adicional como el tiempo	

estimado de llegada y la distancia en kilómetros. De esta manera, el proveedor podrá contar con una guía visual que le permita seguir la ruta de manera eficiente y llegar al destino deseado dentro del tiempo estimado, mejorando así la calidad del servicio ofrecido.

Tabla 37

Tarea 5.4 – Notificación de llegada al cliente

Tarea	
Nro. de tarea: 5.4	Nro. Historia de usuario: 5
Nombre de la tarea: Implementar la finalización de la solicitud al llegar al destino	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 6
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: La tarea implica habilitar una función que notifica la llegada al destino del cliente y concluye el proceso eficientemente, incluyendo la generación del recibo o confirmación de entrega, si es necesario. Automatizar este cierre asegura una finalización adecuada de la solicitud y mejora la experiencia tanto para el distribuidor como para el cliente.	

HU 6: Visualizar historial de pedidos

En las siguientes tablas se describen todas las tareas correspondientes a la HU 6 – Visualizar historial de pedidos.

Tabla 38

Tarea 6.1 - Historial de pedidos

Tarea	
Nro. de tarea: 6.1	Nro. Historia de usuario: 6
Nombre de la tarea: Implementar la funcionalidad de filtrado del historial de pedidos.	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 5
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: Esta tarea consiste en implementar la lógica necesaria para filtrar los registros del historial. Esto implica realizar consultas adecuadas a la base de datos para obtener los resultados filtrados y presentarlos de manera clara al distribuidor, brindándole una visión específica de su historial.	

HU 7: Ingreso Manual de Ubicación

En las siguientes tablas se describen todas las tareas correspondientes a la HU 7 – Ingreso manual de ubicación.

Tabla 39

Tarea 7.1 - Búsqueda de lugar en el mapa

Tarea	
Nro. de tarea: 7.1	Nro. Historia de usuario: 7
Nombre de la tarea: Búsqueda de lugar en el mapa	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 5
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: Esta tarea implica diseñar y desarrollar una función de búsqueda en el mapa que sea intuitiva y fácil de usar para los clientes. La función debe permitir a los clientes ingresar manualmente una ubicación específica para posterior a ellos selecciona manualmente su ubicación para ser preciso en la ubicación de su pedido.	

Tabla 40

Tarea 7.2 - Integración de Ubicación Manual

Tarea	
Nro. de tarea: 7.2	Nro. Historia de usuario: 7
Nombre de la tarea: Integración de ubicación manual	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 6
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: En esta tarea, se realizará la integración efectiva de la nueva ubicación seleccionada por el cliente en el proceso de pedidos. Esto garantizará que, al realizar un pedido, se considere la ubicación manual en lugar de la ubicación predeterminada.	

HU 8: Gestión de Información Personal

En las siguientes tablas se describen todas las tareas correspondientes a la HU 6 – Gestión de información personal.

Tabla 41*Tarea 8.1 – Edición de Perfil*

Tarea	
Nro. de tarea: 8.1	Nro. Historia de usuario: 8
Nombre de la tarea: Integración de edición de información personal	
Tipo de tarea: Desarrollo	Tiempo estimado en horas: 6
Responsable: Diego Carlosama y Alexander Quinatoa	
Descripción: Esta tarea implica desarrollar la funcionalidad que permitirá a los usuarios, tanto clientes como distribuidores, editar y actualizar su información personal. Esto incluye campos como nombre, dirección de correo electrónico, foto de perfil (opcional) y número de teléfono. Los usuarios deben poder realizar cambios en sus perfiles de manera efectiva.	

3.1.1.4 Velocidad del proyecto

El desarrollo del proyecto se rige por un cronograma de actividades que se llevarán a cabo a lo largo del proceso. En la **Tabla 52** se especifican cada una de las tareas a realizar, así como su fecha estimada de entrega, lo que proporciona una visión general del proyecto y la cantidad de tiempo necesaria para su realización. Este enfoque permite una mejor gestión y seguimiento del progreso del proyecto, brindando un marco temporal claro y preciso.

Tabla 42*Cronograma de actividades*

Nro. HU	Nro. Tarea	Tarea	Fecha estimada	Horas de desarrollo
1	1.1	Generación y envío de código de verificación	15/05/2023	3
1	1.2	Verificación y reenvío de códigos	19/05/2023 – 27/05/2023	3
2	2.1	Reutilización de componentes para el inicio de sesión en el proceso de registro.	29/05/2023 – 03/06/2023	2
2	2.2	Almacenamiento seguro de toda la información personal	05/06/2023 – 09/06/2023	3
2	2.3	Identificación del rol del usuario y enrutamiento a la interfaz correspondiente	12/06/2023 – 17/06/2023	3
3	3.1	Implementar la funcionalidad de envío de solicitudes a los distribuidores	19/06/2023	9

3	3.2	Cancelación de pedido	20/06/2023 – 23/06/2023	7
3	3.3	Graficar ruta del cliente a distribuidor	26/06/2023 - 30/06/2023	9
3	3.4	Configurar notificaciones en tiempo real para informar al cliente sobre la aceptación del pedido del distribuidor	3/07/2023	8
3	3.5	Configurar notificaciones en tiempo real para informar al cliente sobre la llegada del distribuidor	04/07/2023 - 7/03/2023	8
4	4.1	Capturar información personal del distribuidor	10/07/2023	5
4	4.2	Capturar información del vehículo	11/07/2023 – 14/03/2023	4
5	5.1	Implementar la recepción de solicitudes en tiempo real	17/07/2023 – 19/07/2023	5
5	5.2	Notificar al cliente sobre la aceptación de la solicitud	20/07/2023 – 21/07/2023	6
5	5.3	Graficar ruta de distribuidor a cliente	24/07/2023 – 28/03/2023	5
5	5.4	Graficar ruta de distribuidor a cliente	31/07/2023 – 4/08/2023	7
6	6.1	Implementar la funcionalidad de filtrado del historial de ventas y calificaciones	07/08/2023 – 09/08/2023	5
7	7.1	Búsqueda de lugar en el mapa	10/08/2023 – 11/08/2023	5
7	7.2	Integración de ubicación manual	14/08/2023 – 18/08/2023	6
8	8.1	Integración de edición de información personal	21/08/2023 – 23/08/2023	6
			Total horas	109

3.1.2 Diseño

Se describe el proceso inicial de diseño general, explicando los pasos, las herramientas y las metodologías utilizadas. Se seleccionan las herramientas adecuadas apoyándose en la revisión de la literatura realizada el marco teórico. Además, se tienen en cuenta las ideas y sugerencias que se obtuvieron de la socialización con los usuarios y los distribuidores para el levantamiento de requisitos.

3.1.2.1 Modelado de la base de datos

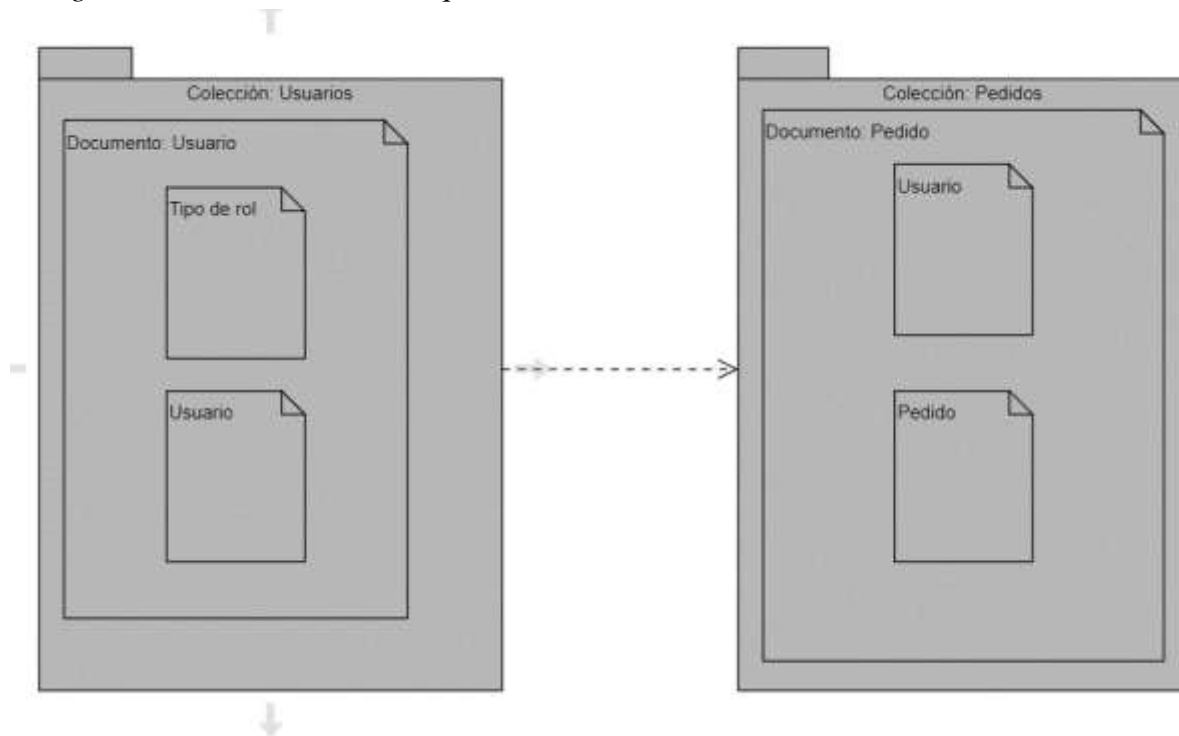
El modelo de base de datos NoSQL, representado en la figura 13, se centra en dos colecciones principales: "Usuarios" y "Pedidos".

En la colección de "Usuarios", se almacena toda la información personal de los usuarios, incluyendo datos relativos a su rol de usuario. Esta estructura permite flexibilidad en el almacenamiento de información personalizada para cada usuario.

En la colección de "Pedidos", se estructura un documento que recopila la información necesaria para llevar a cabo un pedido. Esta información se relaciona con la colección de "Usuarios" para obtener detalles tanto de los usuarios como de los distribuidores correspondientes. La flexibilidad del modelo de base de datos permite adaptarse a diversas necesidades y requerimientos del sistema.

Figura 27

Diagrama de base de datos NoSql



Nota. Fuente: Elaboración propia.

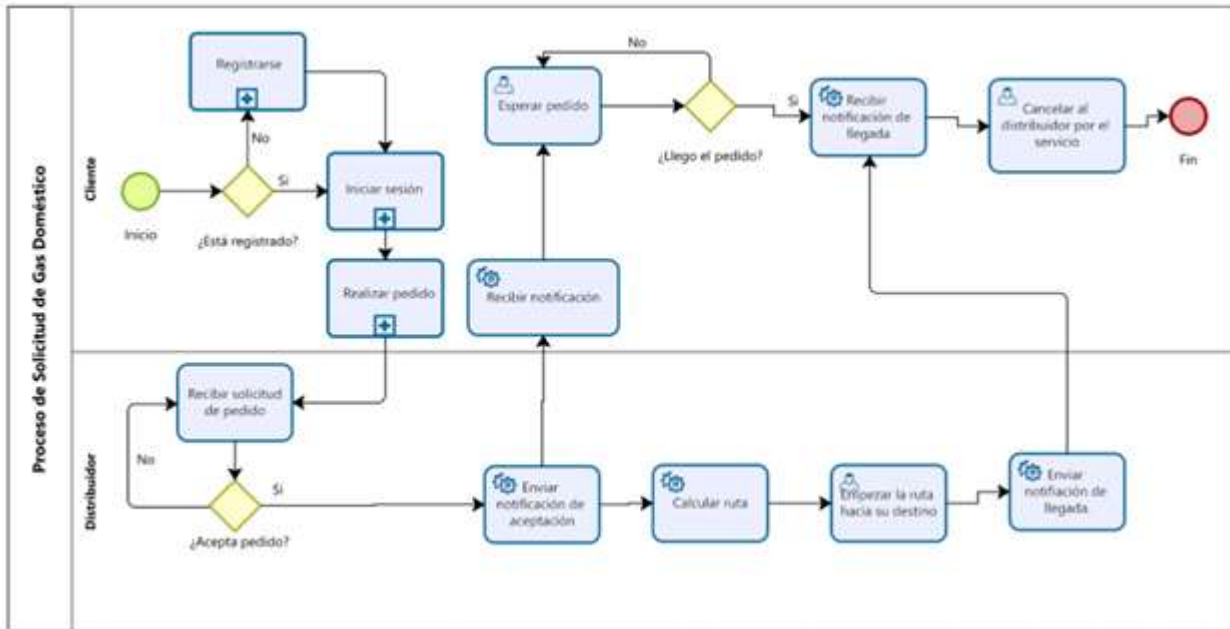
3.1.2.2 Diagramas de procesos

A continuación, se presentan los diagramas que describen los procesos identificados tras el análisis de los requisitos funcionales y no funcionales, los cuales contribuirán a cumplir el objetivo del proyecto. Estos diagramas proporcionan una visualización clara de las interacciones y pasos involucrados en el flujo de trabajo, brindando una guía precisa para el desarrollo y la implementación del sistema.

En la figura 18, se detalla de manera específica el proceso principal el cual es el de solicitud de gas doméstico por parte del cliente hacia el distribuidor, el cual incluye varios subprocesos relacionados con dicha solicitud.

Figura 28

Diagrama de proceso para la gestión de pedidos



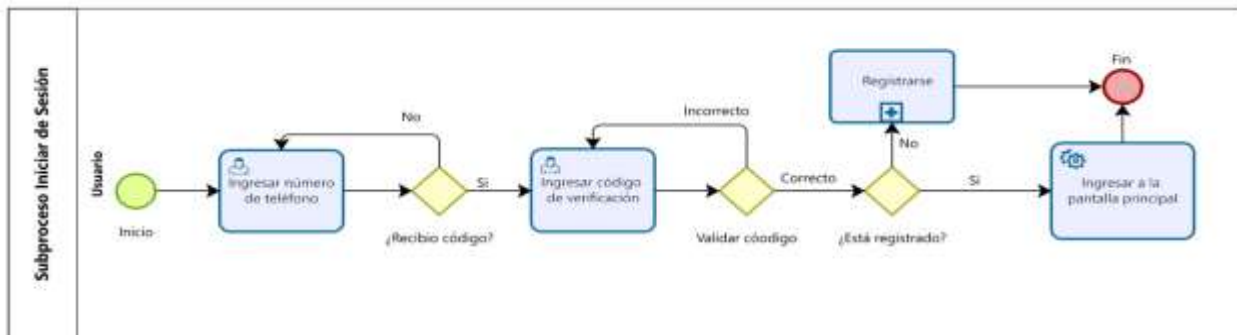
Nota. Fuente: Elaboración propia.

A continuación, se presentan en las figuras subsiguientes los subprocesos que complementan el proceso principal detallado en la figura 28:

- **Subproceso Iniciar Sesión:** Este subproceso, representado en su figura respectiva, permite a los usuarios acceder a las funcionalidades relacionadas con su rol seleccionado.

Figura 29

Subproceso de Iniciar Sesión

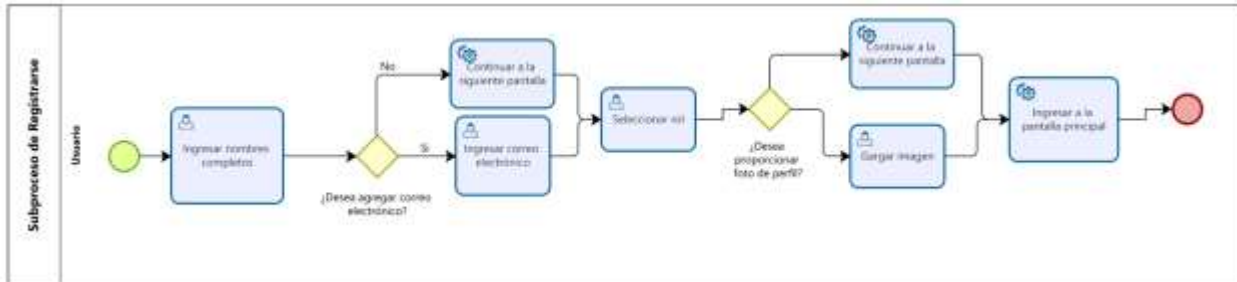


Nota. Fuente: Elaboración propia.

- **Subproceso Registro:** También presente en su figura correspondiente, este subproceso desempeña un papel esencial al recolectar la información necesaria de los usuarios y otorgar la autorización para el uso de la aplicación móvil.

Figura 30

Subproceso Registro

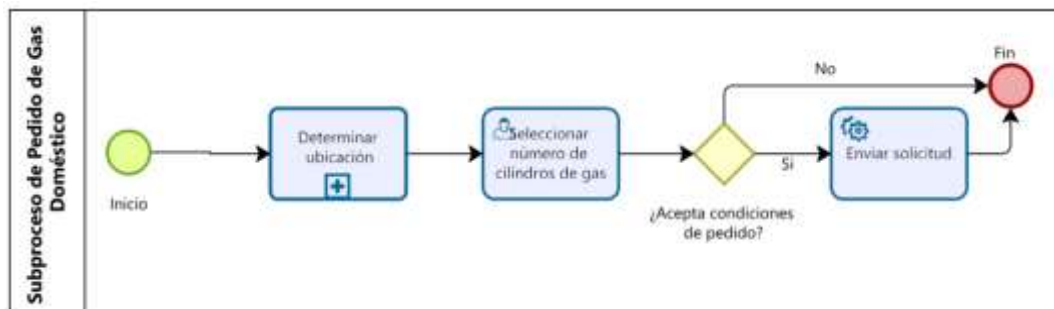


Nota. Fuente: Elaboración propia

- **Subproceso Pedido:** El subproceso descrito en la figura correspondiente resulta fundamental al determinar los requisitos necesarios previos a la realización de un pedido.

Figura 31

Subproceso Pedido

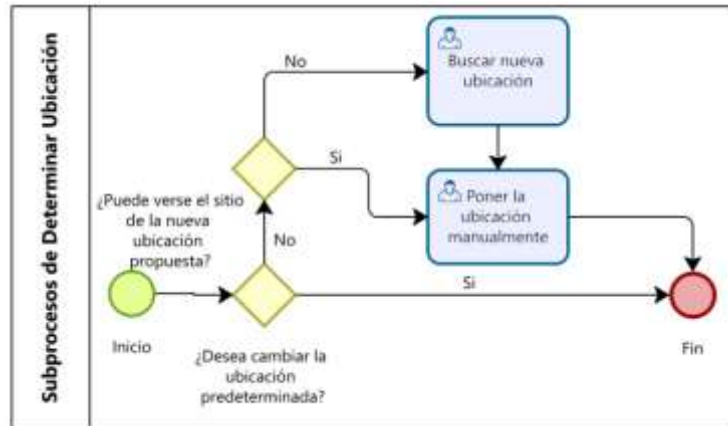


Nota. Fuente: Elaboración propia

- **Subproceso Determinar Ubicación:** El subproceso que se muestra en su figura respectiva tiene como propósito ofrecer a los clientes la opción de cambiar su ubicación predeterminada a un destino específico donde deseen que se entregue su pedido.

Figura 32

Subproceso Determinar Ubicación



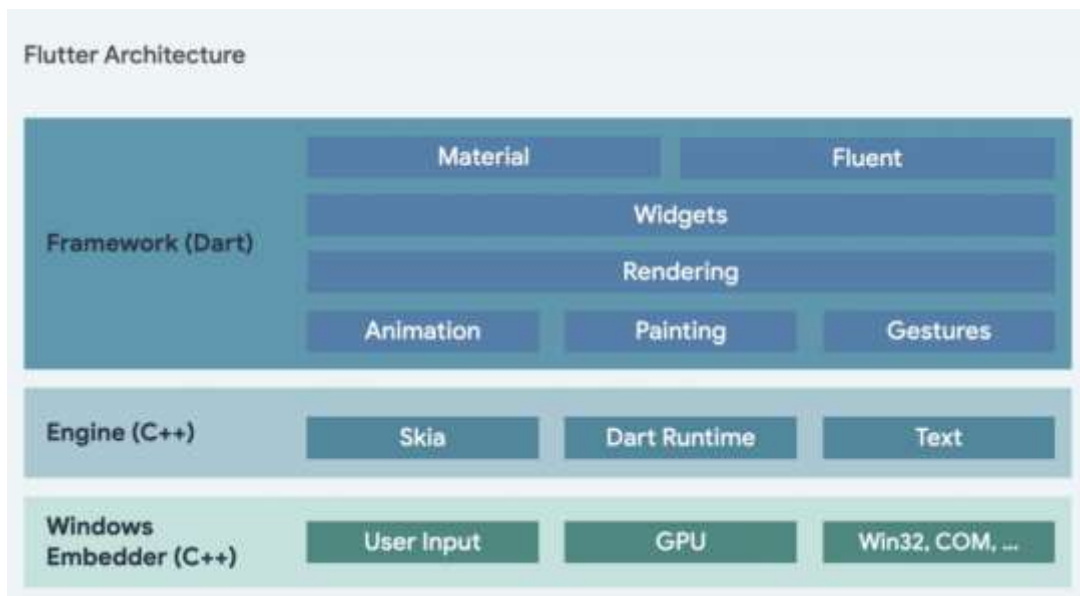
Nota. Fuente: Elaboración propio.

2.1.1.1 Arquitectura interna del proyecto

Para la arquitectura del proyecto; todos los elementos del marco Flutter pueden ser utilizados por aplicaciones. En Windows, también puede comunicarse con Win32, COM y las API de Windows Runtime, ya sea directamente a través de la capa de interoperabilidad C de Dart o mediante complementos de plataforma escritos en C++ (Google for Developers, 2022.).

Figura 33

Arquitectura interna



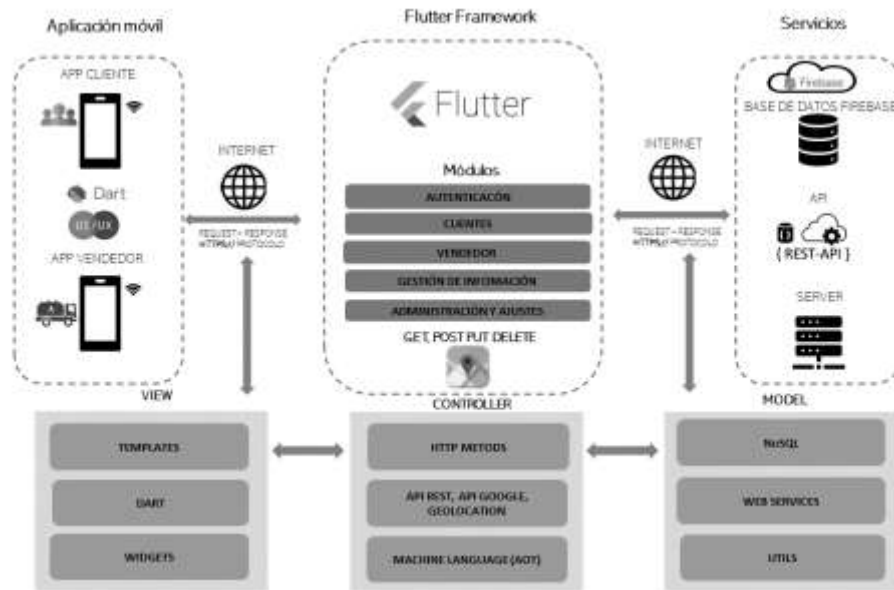
Nota. Se describe como está estructurado el proyecto internamente cuando se crea una aplicación de Flutter. Adaptado de: <https://developers-kr.googleblog.com/2022/03/announcing-flutter-for-windows.html>

3.1.2.3 Arquitectura de la aplicación

La arquitectura es realizada en modelo vista controlador de tipo monolítica, esta aplicación al ser un prototipo pequeño no necesita consumir microservicios de alta calidad.

Figura 34

Arquitectura de la aplicación móvil



Nota. Fuente: Elaboración propia.

3.1.2.4 Despliegue de la aplicación

3.1.2.4.1 Despliegue de del Api Rest

El proyecto consta de un servicio de Api Rest desarrollado en Node JS para la parte del BackEnd, este puede ser consumido desde cualquier lugar y para ello se desplegó en la plataforma online Fly.io. Esta plataforma permite desplegar servidores de aplicaciones cerca de los usuarios, utilizando Docker y una red privada global (Fly.io, 2023a).

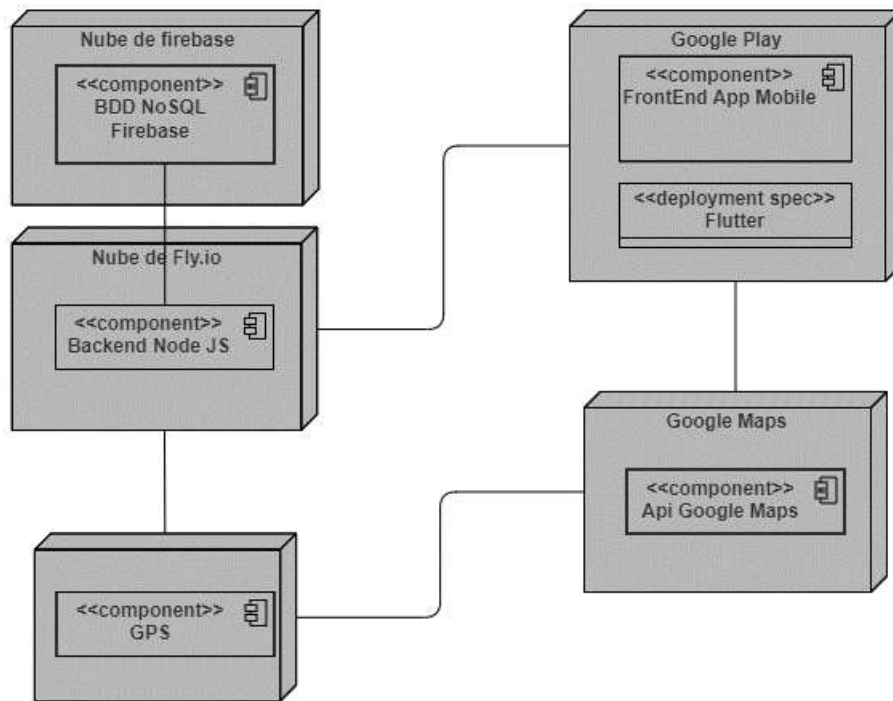
3.1.2.4.2 Despliegue de Aplicativo Móvil

Se llevó a cabo un despliegue en la plataforma de Google Play para hacer que la aplicación móvil esté disponible tanto para los clientes como para los distribuidores. Antes de realizar el despliegue en Google Play Store, se realizaron pruebas en diferentes dispositivos y se revisaron los términos, políticas y requisitos. Además, se creó una cuenta de desarrollador en Google Play

Console y se configuraron los detalles de la aplicación, como el título, la descripción y las capturas de pantalla. Posteriormente, se procedió a subir la APK o el paquete de la aplicación, especificando información relacionada con la versión y la distribución. Por último, la aplicación fue publicada a través de Google Play Console para su revisión y aprobación, asegurándose de cumplir con los requisitos y directrices establecidos por Google Play Store (Google Help, 2023a).

Figura 35

Diagrama de despliegue



Nota. Se describe como están empaquetados y cómo funciona el flujo de información. Fuente. Elaboración propia.

3.1.2.5 Prototipo de la aplicación móvil

Para el diseño de la aplicación móvil se utiliza la herramienta Adobe XD. Esta herramienta permite crear prototipos funcionales aplicando colorimetría, diseño y se hace mediante la interpretación del archivo SVG que exporta Adobe XD (MacHado & Campos, 2021); además, simular el funcionamiento del prototipo y desplegar mediante un enlace generado en la web, esto ayuda a probar e interactuar el BackEnd con el Frontend remotamente, se realiza un control de versiones con el programador y se ajusta los requerimientos nuevos.

Figura 36

Plataforma para diseñar prototipos

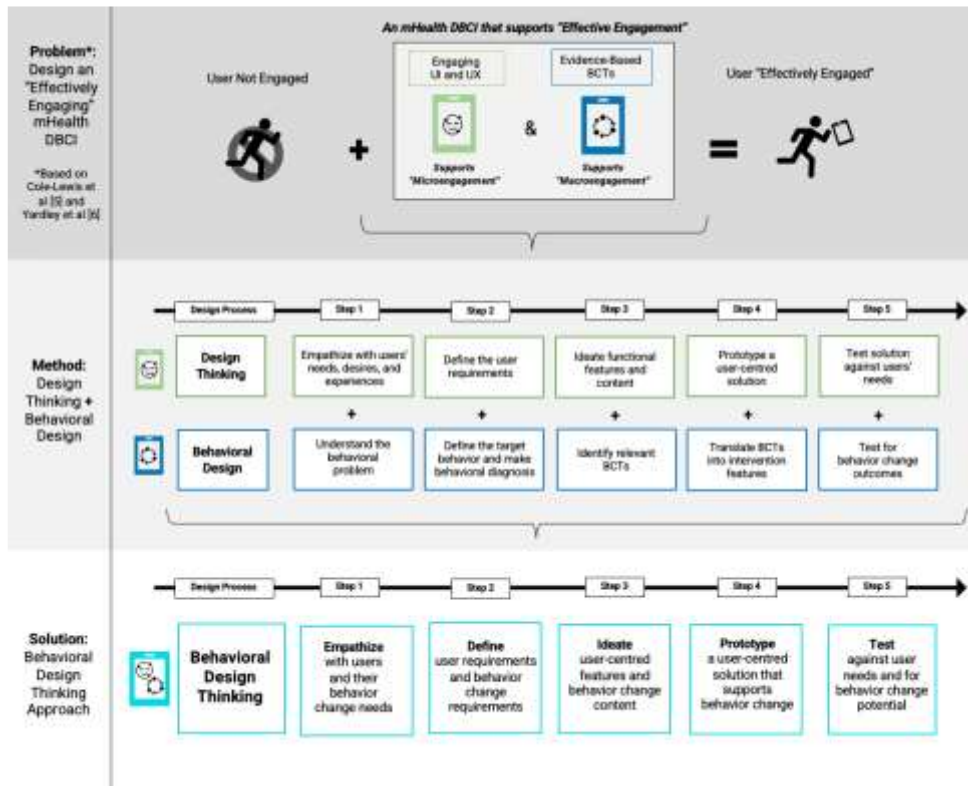


Nota. Fuente. Recuperado de <https://n9.cl/obfqp>

La metodología para la creación de prototipos sigue el proceso definido en la investigación realizada en Toronto, que trata sobre emplear dos metodologías para lograr un mejor resultado.

Figura 37

Methodology Behavioral Design Thinking Approach



Nota. Implementación de dos metodologías para el diseño de interfaces de usuario, Adaptado de (Voorheis et al., 2022).

3.1.2.5.1 Empatizar

Luego de realizar la investigación documentada, las encuestas y entrevistas ayudaron a identificar los procesos y empatizar la idea de una alternativa de adquirir los cilindros de gas mediante una aplicación móvil. Los usuarios y los distribuidores son los principales autores para entender las necesidades y los temores que tienen a estas tecnologías. Además, se socializó con aplicaciones referentes de pedidos a domicilio, taxis que trabajan de tipo InDriver y las facilidades que brindan estos servicios. Para esto se utilizó el mapa de empatía para conocer las necesidades de los usuarios.

Figura 38

Plantilla utilizada para la implementación del Mapa de Empatía



Nota. Fuente. Adaptado de: <https://n9.cl/ekruma>

3.1.2.5.2 Definir

El equipo de trabajo analizó la información obtenida para establecer los requisitos de la aplicación móvil y encontrar los aspectos y factores principales que influyeron en el problema. Posteriormente, se seleccionaron los requisitos considerados más adecuados en términos de viabilidad, relevancia y concordancia con las metas del proyecto y las demandas de los usuarios.

3.1.2.5.3 Idear

Se realizó una sesión de lluvia de ideas para idear conceptos de aplicaciones móviles relacionadas con pedidos y entregas a domicilio que funcionan en la localidad. Se compartieron las ideas con expertos en el área de aplicaciones móviles de la Universidad Técnica del Norte para

obtener retroalimentación adicional. El objetivo fue cubrir las necesidades de los usuarios, involucrando al cliente en la fase de pedidos y a los distribuidores en la parte de entrega y pagos.

Figura 39

Lluvia de ideas para empezar a prototipar la aplicación móvil mi gas



Nota. Fuente: Elaboración propia.

3.1.2.5.4 Prototipar

En esta fase, se crearon los prototipos utilizando herramientas tecnológicas y manuales para diagramas de flujo, diagramas de casos de uso, bocetos en papel; basados en la metodología XP que promueve la creación intuitiva de prototipos, de manera persuasiva y conductual. El equipo de trabajo supervisó de cerca el proceso para realizar ajustes y refinamientos según fuera necesario.

Figura 40

Boceto de inicio de sesión para el usuario

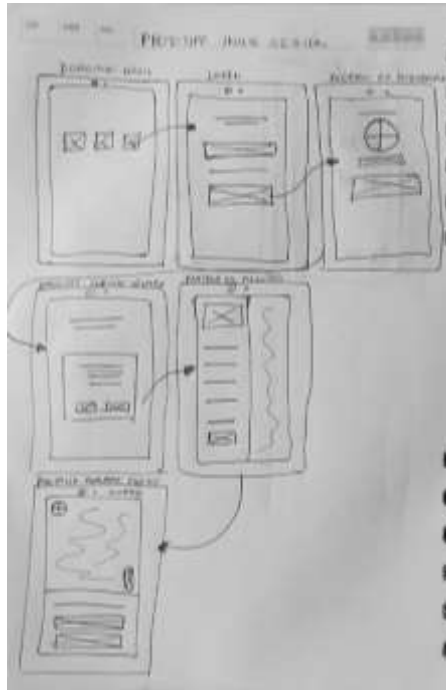


Figura 41

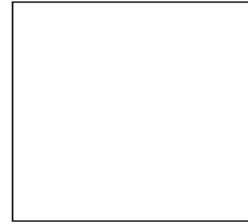
Boceto de inicio de sesión para el distribuidor e ingreso de documentos del distribuidor



Nota. Elaboración propia.

3.1.2.5.5 Colores principales

Los colores siguientes son los que se utilizaron en las pantallas de la aplicación desarrollada Flutter, tales como botones, letras, fondos e iconos.

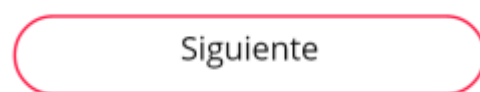


-
- backgroundColor: Colors.redAccent[400]/Color(0xFF50057)
 - backgroundColor: Color.fromRGBO(255, 23, 68, 1),
 - backgroundColor: Colors.white,
 - backgroundColor: Color.fromRGBO(255, 255, 255, 1)

Nota. *Fuente: Elaboración propia*

3.1.2.5.6 Colores de botones y posiciones de pantalla

A continuación, se detalla los colores utilizados para el desarrollo de los botones y las dimensiones que separan en las tallas.



backgroundColor:
Colors.redAccent[400]/
Color(0xFF50057)
color de letra: with
tipo de letra = Open Sans
radius = 30
fontSize: 23
fontWeight: FontWeight.normal,

backgroundColor: Colors.white,
color de letra: black
tipo de letra = Open Sans
borderColor: Color.fromRGBO(255,
23, 68, 1)
radius = 30
borderWidth = 2
fontSize: 23

fontWeight: FontWeight.normal,



ancho de botón = minimumSize: Size
(double.infinity, 50)

largo de botón = padding mínimo de
10 o container de 10



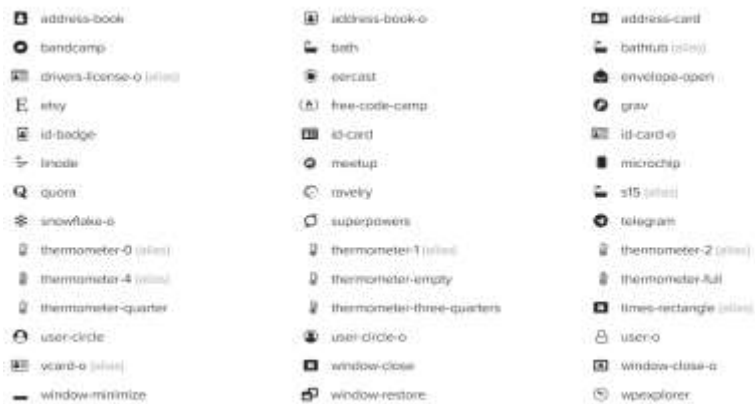
activeColor: Colors.blue,
trackColor: Colors.grey,

3.1.2.5.7 Iconos

Los iconos fueron tomados y adaptados de la página oficial de Flutter.

Figura 42

Iconos tomando de la página oficial de Flutter developers



Nota. Fuente: <https://api.flutter.dev/flutter/material/Icons-class.html>

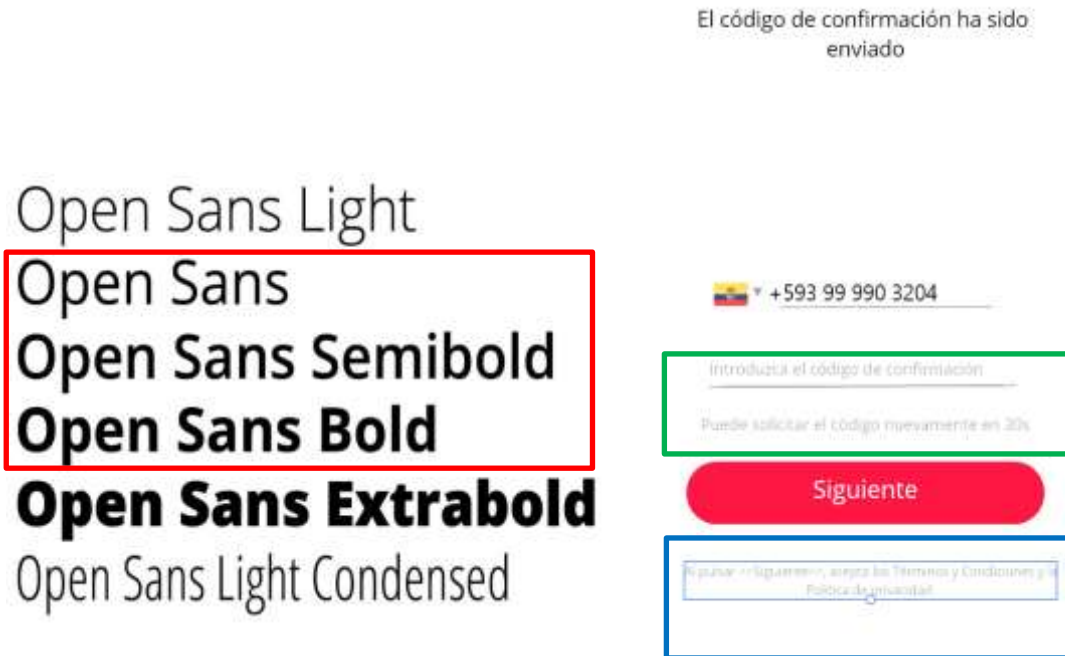
3.1.2.5.8 Tipo de fuente

Google creó la tipografía OpenSans específicamente para dispositivos móviles, siendo la fuente principal utilizada en el sistema operativo Android. Google la describe como una fuente de

naturaleza dual, con un esqueleto mecánico y formas predominantemente geométricas, al mismo tiempo que exhibe curvas amigables y abiertas.

Figura 43

Se muestra la fuente de texto recomendada por Google Fonts para Android



-Se utiliza las dos primeras, en caso de necesitar más negritas la **Bold**

-Cuadro rojo: Letra de títulos, pantallas = 23

-Cuadro verde: letra secundaria = 15

-Cuadro azul: letra nivel tres = 12/13

A continuación, se presenta una tabla donde se detalla el tamaño de los botones que recomienda Google para dispositivos Android.

Tabla 43

Tamaño de botones recomendados por Google

Tamaño táctil: Los botones deben ser lo suficientemente grandes para que los usuarios puedan tocarlos cómodamente con sus dedos sin activar accidentalmente botones cercanos. Se recomienda un tamaño mínimo de 48x48 píxeles como estándar para botones en pantallas táctiles.

Densidad de píxeles: Ten en cuenta la densidad de píxeles de los dispositivos móviles objetivo. Los tamaños en píxeles pueden variar dependiendo de la densidad de la pantalla (mdpi, hdpi, xhdpi, etc.). Puedes utilizar medidas en densidad independiente de píxeles (dp o dip) para asegurar que el tamaño sea coherente en diferentes dispositivos.

Espacio suficiente: Deja suficiente espacio entre los botones para evitar que los usuarios los toquen accidentalmente. Un margen mínimo de 8 píxeles alrededor de cada botón es una buena práctica.

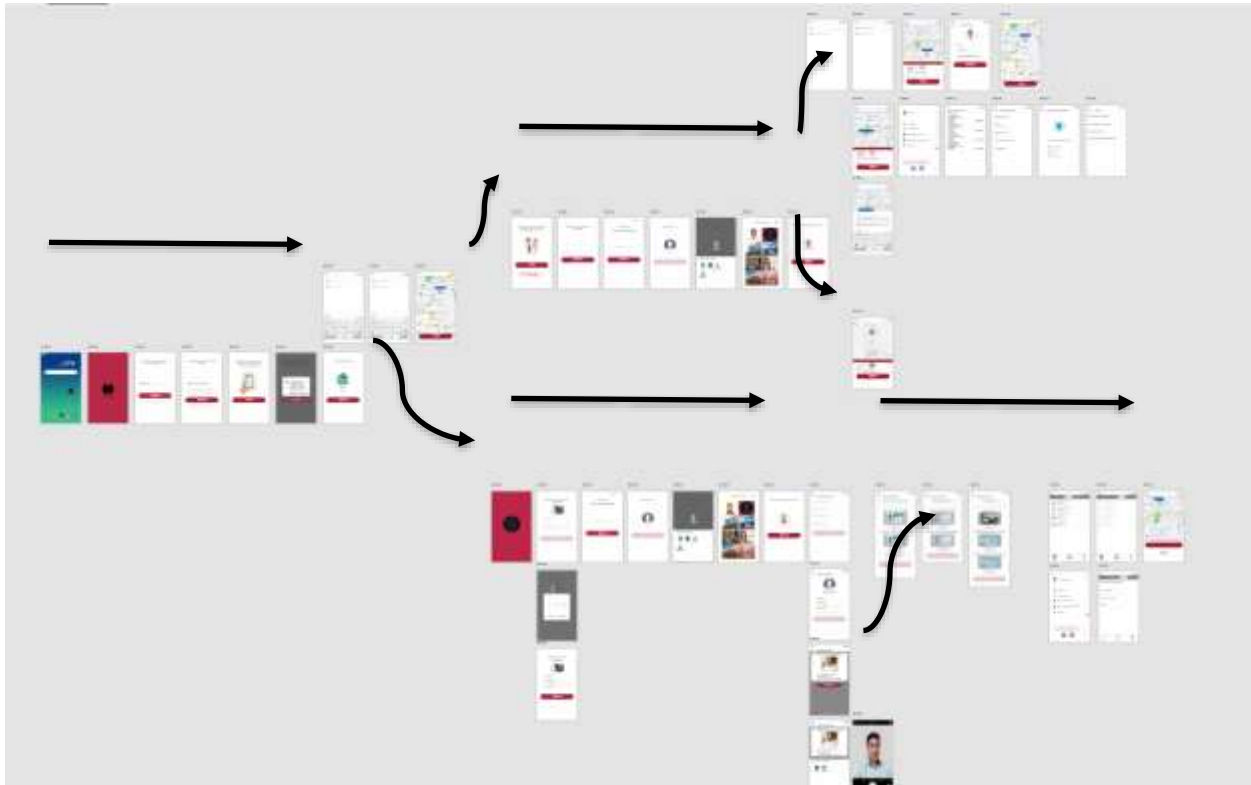
Nota. Fuente: (Google Help, 2023b).

3.1.2.5.9 Testear

En este proceso se realizaron pruebas piloto basadas en modelos de aplicaciones de Delivery, no se realizó pruebas exhaustivas como en la aplicación real. Esto se hizo con el objetivo de permitir posibles ajustes futuros, ya que algunos componentes no fueron completamente similares a la arquitectura Flutter para el desarrollo. Durante el desarrollo, se realizaron ajustes en elementos como fuentes de texto, botones, colores, espacios, márgenes y contenedores, que fueron diseñados en los prototipos. Aquí se muestra los prototipos iniciales realizados en el software abobe XD que fueron evaluados y aprobados para su desarrollo en Flutter.

Figura 44

Prototipo completo del funcionamiento de la aplicación Usuario y Distribuidor



Nota. En las figuras **44** se detalla el desarrollo completo de las pantallas finales que se crearon en Flutter y su respectivo funcionamiento.

3.1.3 Codificación

Antes de comenzar la etapa de codificación, es imprescindible contar con ciertos requisitos de desarrollo que revisten gran importancia para esta sección.

3.1.3.1 Requerimientos de desarrollo

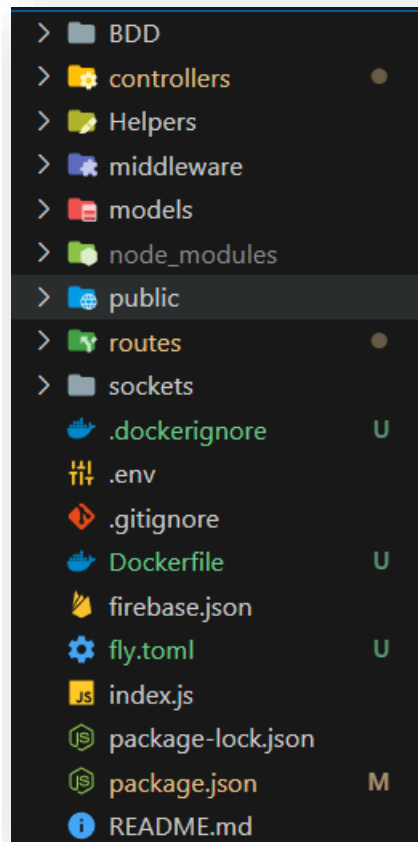
En este apartado se describe las herramientas que se utilizaron para el desarrollo del backend.

- Para el desarrollo del Backend
 - Firebase
 - El editor de código Visual Studio Code
 - Node.js
 - Postman
 - El framework Express

Para el desarrollo del backend, se ha implementado el patrón de diseño Modelo-Vista Controlador (MVC) con el objetivo de lograr una mejor organización y separación de responsabilidades (Danilo Gavilánez Álvarez et al., 2022). Este enfoque promueve modularidad y la legibilidad del código. En cuanto a la estructura del proyecto, se han establecido varias carpetas clave. La carpeta "Controllers" alberga la lógica de la API REST, donde se definen y gestionan las acciones y operaciones relacionadas con los datos. Por otro lado, la carpeta "Routes" se encarga de crear las rutas y URL correspondientes para procesar las solicitudes y establecer la comunicación con el backend. En esta sección se definen los puntos finales (endpoints) que se utilizaron para realizar las diversas consultas y operaciones. Asimismo, la carpeta "Models" desempeña un papel fundamental al permitir la representación y el modelado de los datos, especialmente si se trabaja con una base de datos no relacional. Esta estructura facilitó la organización de los diferentes modelos de datos y la interacción con la base de datos seleccionada. Además de estas carpetas principales, también se han incluido otras de gran importancia, como "Helpers" y "Middlewares". La carpeta "Helpers" contiene funcionalidades adicionales que brindan apoyo y ayudan en el desarrollo, mientras que la carpeta "Middlewares" proporciona capas de seguridad y validaciones para proteger la API REST.

Figura 45

Estructura Backend



Nota. Elaboración propia.

En base de datos se optó por Firebase, que es una excelente opción para el backend y la distribución de la aplicación móvil para el servicio de gas doméstico. Al hacer uso de Firebase Realtime Database para almacenar y sincronizar los datos en tiempo real, facilita el seguimiento de los pedidos de gas, la gestión de inventario y la actualización instantánea de la información para los usuarios y los repartidores (Marlen Treviño-Villalobos et al., 2019).

- Para el desarrollo del Frontend.
 - Flutter
 - Api de Google
 - Android Studio

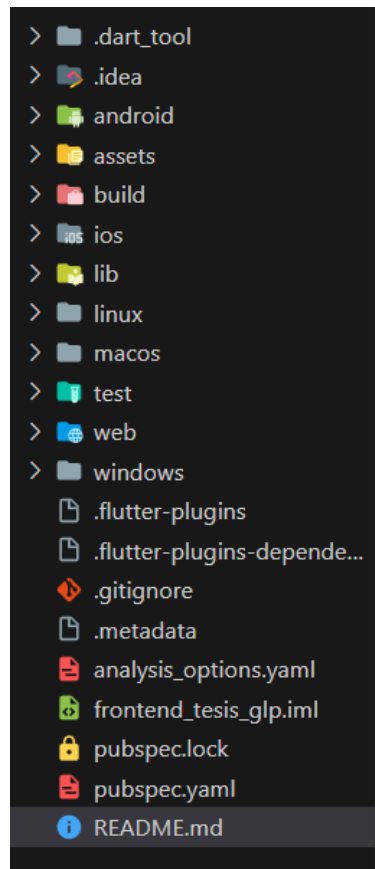
- Editor de código Visual Code

En el proceso de desarrollo del frontend, se ha adoptado y seguido rigurosamente la estructura de carpetas recomendada por Flutter, basada en las mejores prácticas de organización. Esta estructura, representada en la figura 34, presenta una serie de carpetas fundamentales para el proyecto. Dos de las carpetas más destacadas son "Android" e "IOS", las cuales contienen archivos de configuración necesarios para generar el archivo APK y permitir una instalación fluida en dispositivos Android e iOS respectivamente. Otra carpeta clave es "lib", que constituye el núcleo de la aplicación y alberga toda la lógica. Para lograr un mayor modularidad y una organización eficiente del código, se ha establecido una estructura de subcarpetas dentro de "lib", donde cada subcarpeta se enfoca en una funcionalidad específica o en un componente determinado de la aplicación. Asimismo, cabe resaltar la importancia del archivo "pubspec.yaml", que desempeña un papel esencial al listar las bibliotecas y dependencias necesarias para asegurar el correcto funcionamiento de la aplicación, evitando así errores en la ejecución (Corral et al., 2018). La implementación de esta estructura de carpetas en el desarrollo del frontend garantiza una gestión ordenada y comprensible del código, facilitando el mantenimiento, la escalabilidad y la colaboración en el proyecto.

En el proceso de desarrollo del frontend, se ha adoptado y seguido rigurosamente la estructura de carpetas recomendada por Flutter, basada en las mejores prácticas de organización. Esta estructura, representada en la figura 46, presenta una serie de carpetas fundamentales para el proyecto. Dos de las carpetas más destacadas son "Android" e "IOS", las cuales contienen archivos de configuración necesarios para generar el archivo APK y permitir una instalación fluida en dispositivos Android e iOS respectivamente. Otra carpeta clave es "lib", que constituye el núcleo de la aplicación y alberga toda la lógica. Para lograr un mayor modularidad y una organización eficiente del código, se ha establecido una estructura de subcarpetas dentro de "lib", donde cada subcarpeta se enfoca en una funcionalidad específica o en un componente determinado de la aplicación. Asimismo, cabe resaltar la importancia del archivo "pubspec.yaml", que desempeña un papel esencial al listar las bibliotecas y dependencias necesarias para asegurar el correcto funcionamiento de la aplicación, evitando así errores en la ejecución (Corral et al., 2018). La implementación de esta estructura de carpetas en el desarrollo del frontend garantiza una gestión ordenada y comprensible del código, facilitando el mantenimiento, la escalabilidad y la colaboración en el proyecto.

Figura 46

Estructura Frontend



Nota. Elaboración propia.

La utilización de la API de Google para geolocalización brinda a los desarrolladores acceso a funcionalidades avanzadas relacionadas con la ubicación y los mapas. A través de Google Maps Platform y Google Places API, se pueden integrar mapas interactivos, búsqueda de lugares, georreferenciación inversa y en lotes, obteniendo datos precisos y detallados como direcciones, reseñas y calificaciones (Zambrano et al., 2018). Esta API proporciona a los desarrolladores la capacidad efectiva de crear aplicaciones basadas en la ubicación y mejorar la experiencia del usuario de manera significativa.

3.1.4 Pruebas

Se realizaron varias pruebas en la aplicación móvil, en dichas pruebas se evaluaron los siguientes aspectos:

- Backend
 - Iniciar sesión
 - Registrarse
 - Registro adicional distribuidor
 - Realizar pedido
 - Aceptar pedido
 - Recepción de solicitudes de pedido
 - Editar información personal (distribuidor, cliente)
 - Editar número de teléfono (distribuidor, cliente)
- Frontend
 - Visualizar mapa con su ubicación
 - Visualizar la ruta para entrega del gas (cliente, distribuidor)
 - Visualizar lista de pedidos por cliente
 - Visualizar lista de pedidos por distribuidor
 - Visualizar nueva ubicación
 - Visualizar ubicación del distribuidor en la vista del cliente
 - Visualizar Información personal (cliente, distribuidor)

3.1.4.1 Inicio de sesión

La entrada a la pantalla principal de la aplicación móvil se restringe a usuarios que previamente se hayan autenticado. Cuando el usuario proporciona su número de teléfono de manera correcta, recibe un código de verificación a través de WhatsApp. Luego, debe ingresar este código para confirmar su autenticidad. En caso de que el usuario no esté registrado, se le guía a través del proceso de registro. Esto se puede observar en la figura 47, que muestra la pantalla donde se ingresan los datos del número de teléfono, y en la figura 48, que evidencia el resultado exitoso.

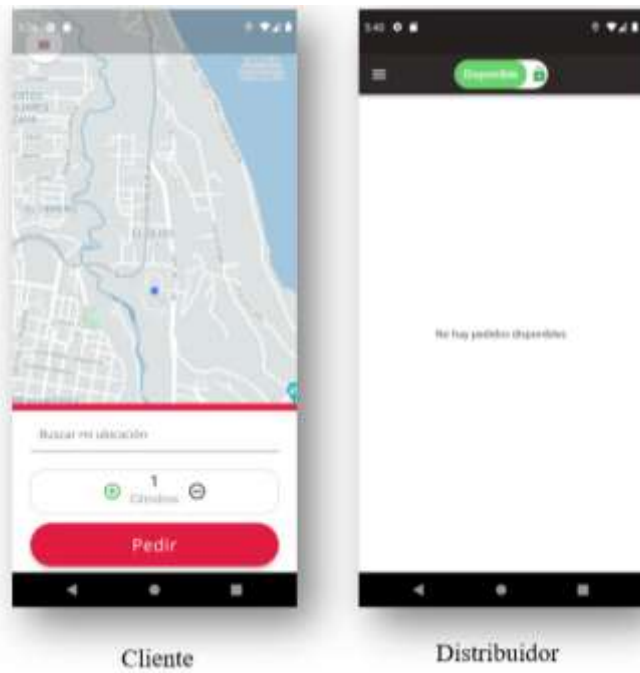
Figura 47

Pruebas, pantalla Inicio de Sesión



Figura 48

Pruebas, resultado de inicio de sesión



3.1.4.2 Registrarse

Si alguien inicia sesión sin tener un registro previo, debe completar un proceso de registro que consta de varios pasos. Estos pasos pueden involucrar la inclusión de información personal, algunos de los cuales son opcionales y otros obligatorios. La figura 49 muestra visualmente el proceso de registro para una mayor claridad, y en la figura 50 se puede ver el resultado exitoso del mismo.

Figura 49

Pruebas, registrarse

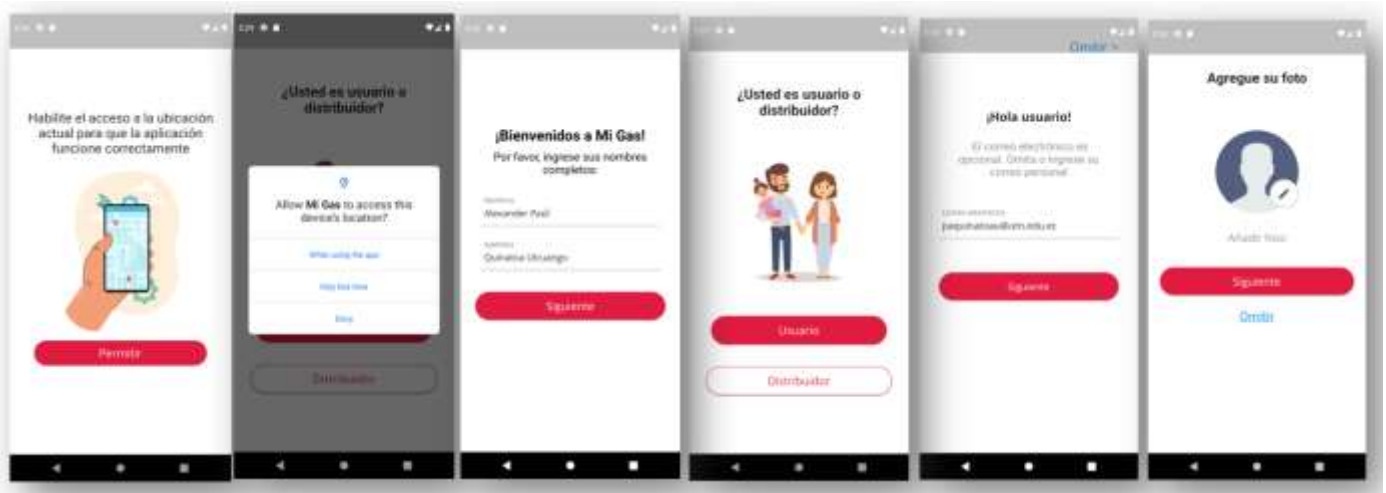
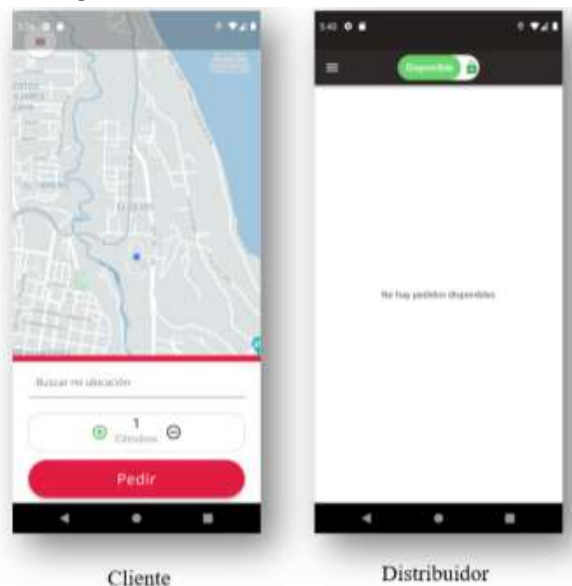


Figura 50

Pruebas, resultado de registrarse



3.1.4.3 Registro adicional distribuidor

Luego de finalizar el proceso de registro básico, a los distribuidores les corresponde incorporar datos adicionales específicos que les otorgan la acreditación necesaria para la distribución de gas doméstico. Esto, a su vez, les habilita para utilizar el módulo de distribución. En la Figura 51 se presentan las diversas pantallas que albergan la información pertinente requerida para activar el uso del módulo. Además, se destaca el éxito de este proceso, como se evidencia en la Figura 52.

Figura 51

Pruebas, registro adicional distribuidor

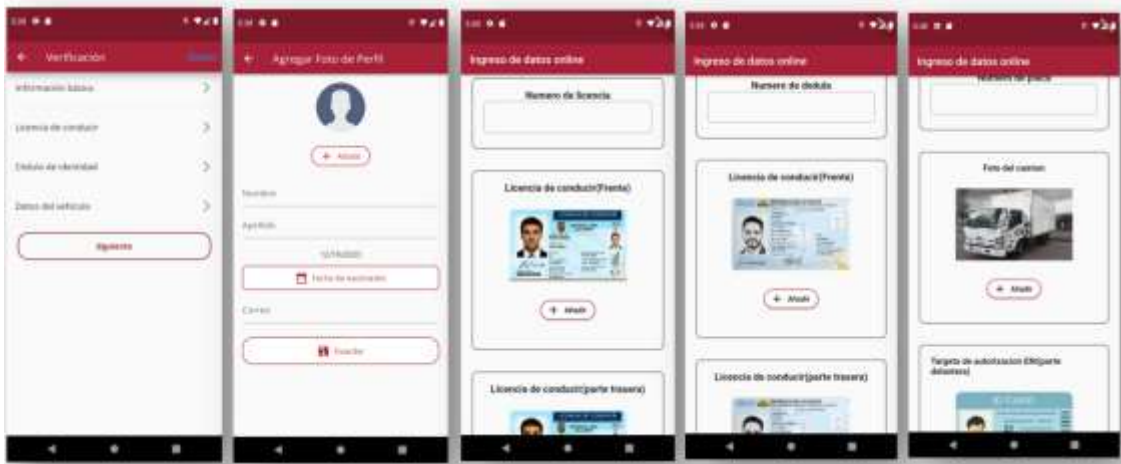
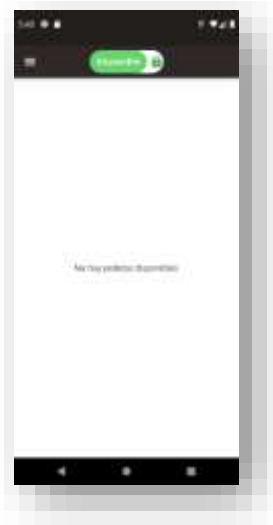


Figura 52

Pruebas, resultado de registro adicional distribuidor



3.1.4.4 Realizar pedido

Los clientes tienen la posibilidad de realizar un pedido de manera sencilla al especificar su ubicación y la cantidad de cilindros que requieren. En la figura 53, se puede observar una pantalla que exhibe un formulario con los requisitos necesarios para efectuar el pedido, junto con un mapa que muestra la ubicación del cliente. Además, en la figura 54, se puede apreciar el éxito del proceso, el cual incluye una notificación que indica que se está buscando un distribuidor disponible para brindar sus servicios.

Figura 53

Pruebas, realizar pedido

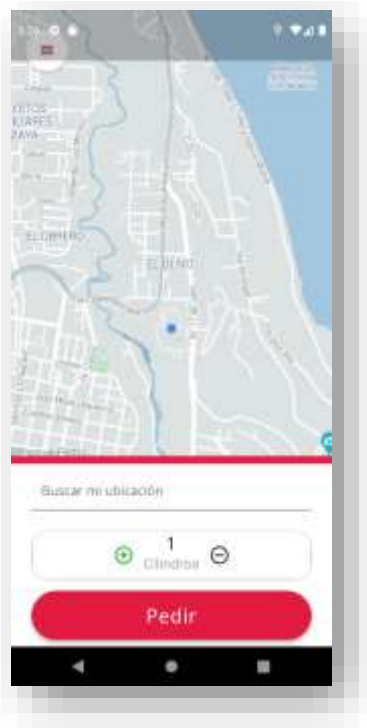


Figura 54

Pruebas, resultado realizar pedido



3.1.4.5 Recepción de solicitudes de pedido

En la pantalla principal, el distribuidor tiene la opción de recibir las solicitudes de pedido de los clientes, como se muestra en la figura 52. En la figura 55, se presenta el resultado exitoso que incluye una lista de los pedidos recibidos.

Figura 55

Pruebas, resultado de recepción de pedidos de pedidos



3.1.4.6 Aceptar pedo

En la lista de solicitudes de pedido, como se muestra en la figura 55, el distribuidor dispone de la capacidad para aceptar el pedido. Al hacerlo, se proporciona la información necesaria para generar una ruta que calcula tanto el tiempo estimado como la distancia en kilómetros entre la ubicación del distribuidor y la del cliente. Esto permite al distribuidor tener una perspectiva clara de la dirección a la que debe dirigirse, como se visualiza en la figura 56.

Figura 56

Pruebas, resultado de aceptar un pedido.



3.1.4.7 Editar información personal (cliente, distribuidor)

Tanto los distribuidores como los clientes cuentan con la posibilidad de editar su información personal. Esto se puede apreciar en la figura 57. Cuando finalizan el proceso de edición y, si es necesario, actualización de la información, se confirma con éxito, como se muestra en la figura 58.

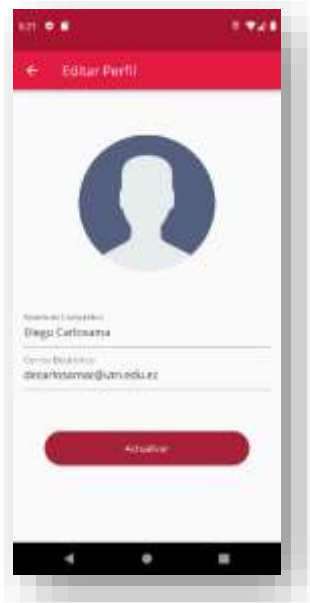
Figura 57

Pruebas, editar información personal (cliente, distribuidor)



Figura 58

Pruebas, resultado de editar información personal (cliente, distribuidor)



3.1.4.8 Editar número de teléfono

El número de teléfono es un dato de gran relevancia en la aplicación, por lo que se ha implementado una función específica para editar esta información. La funcionalidad de edición se visualiza en la figura 59, y el éxito al actualizar este dato se muestra en la figura 60.

Figura 59

Pruebas, editar número de teléfono.



Figura 60

Pruebas, resultado

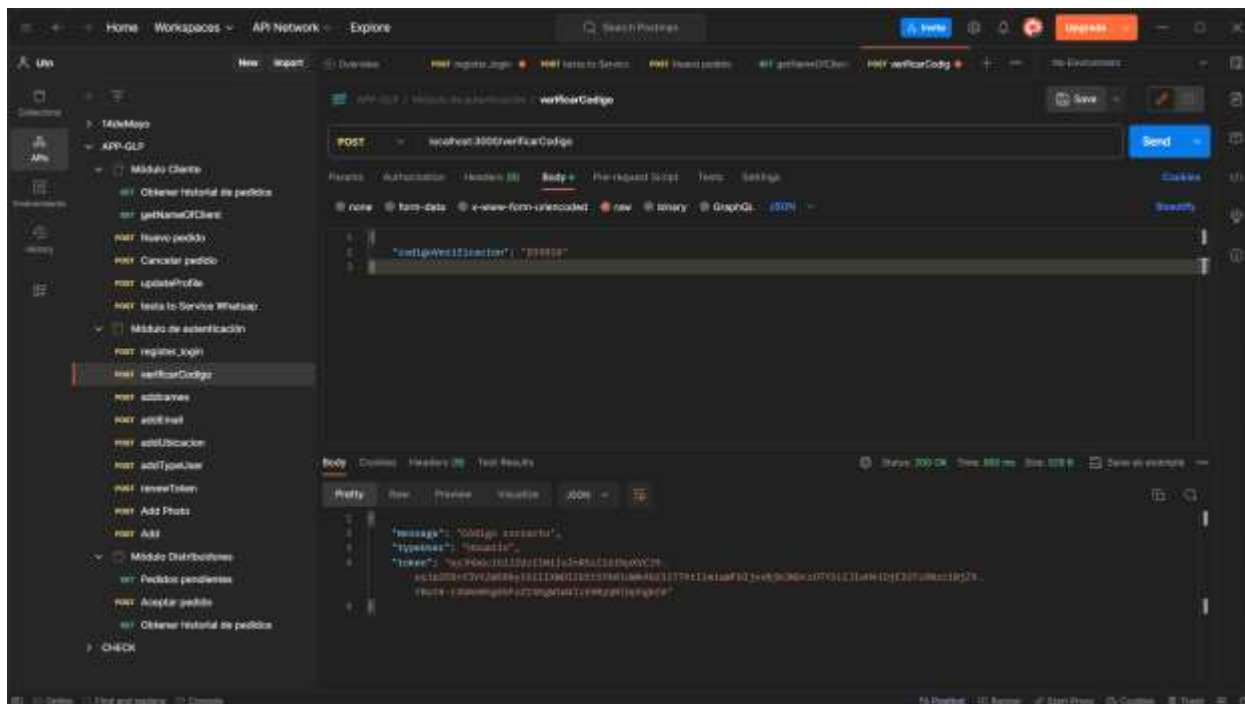


3.1.4.9 Api Rest

En la Figura 61, se pueden observar las diversas funcionalidades desarrolladas en el backend, junto con sus correspondientes pruebas de consumo HTTP. Estas pruebas se realizan utilizando la herramienta Postman, como se muestra en la misma figura.

Figura 61

Pruebas, API REST - verificación de código



3.1.4.10 Visualizar mapa con su ubicación

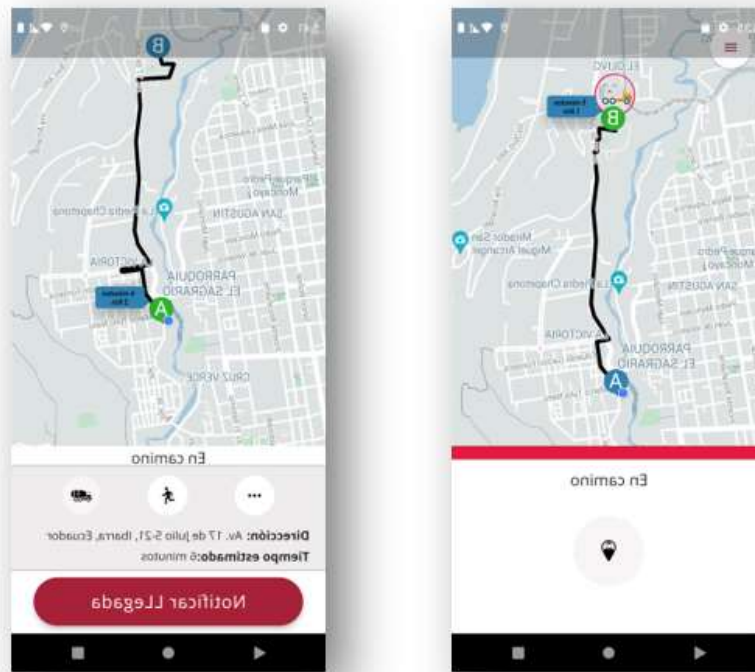
La ejecución de esta prueba se lleva a cabo a través de la aplicación móvil, permitiendo al cliente visualizar su ubicación en el mapa, tal como se muestra en la **Figura 53**.

3.1.4.11 Visualizar la ruta para entrega del gas (cliente, distribuidor)

La ejecución de esta prueba se realiza mediante la aplicación móvil, lo que permite tanto al cliente como al distribuidor visualizar la ruta desde la ubicación del cliente hasta la del distribuidor, como se muestra en la Figura 62.

Figura 62

Pruebas, visualizar la ruta en el mapa



Distribuidor

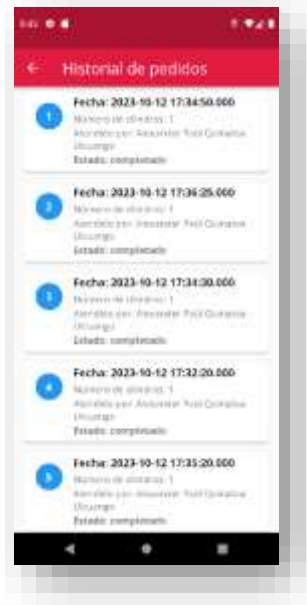
Cliente

3.1.4.12 Visualizar lista de pedidos por cliente

La ejecución de esta prueba se realiza mediante la aplicación móvil, la cual despliega un listado que incluye todas las solicitudes de pedido completadas al 100% por el cliente. Este proceso exitoso se puede apreciar claramente en la Figura 63.

Figura 63

Pruebas, visualizar listado de pedidos por cliente

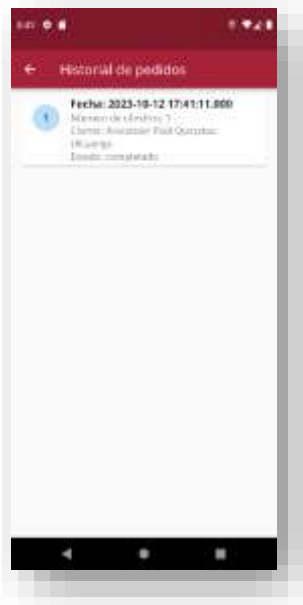


3.1.4.13 Visualizar lista de pedidos por distribuidor

La ejecución de esta prueba se realiza mediante la aplicación móvil, la cual despliega un listado que incluye todas las solicitudes de pedido completadas al 100% por el distribuidor. Este proceso exitoso se puede apreciar claramente en la Figura 64.

Figura 64

Pruebas, visualizar lista de pedidos por distribuidor



3.1.4.14 Visualizar nueva ubicación

La ejecución de esta prueba se realiza mediante la aplicación móvil, donde el cliente tiene la posibilidad de visualizar su nueva ubicación, la cual utilizará como referencia para su destino, tal y como se muestra en la Figura 65.

Figura 65

Pruebas, visualizar nueva ubicación

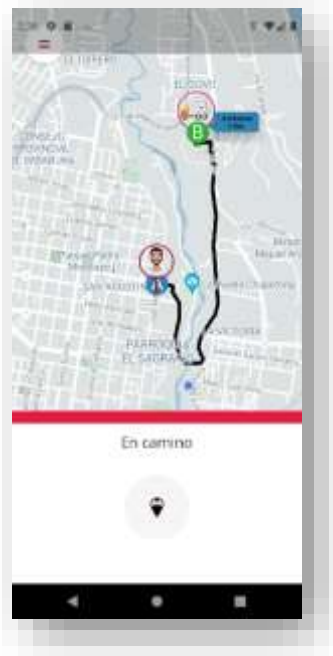


3.1.4.15 Visualizar ubicación del distribuidor en la vista del cliente

La ejecución de esta prueba se realiza mediante la aplicación móvil, donde el cliente puede visualizar en tiempo real la ubicación del distribuidor, como se puede observar en la Figura 66.

Figura 66

Pruebas, visualizar ubicación del distribuidor en la vista del cliente



3.1.4.16 Visualizar Información personal (cliente, distribuidor)

La ejecución de esta prueba se realiza mediante la aplicación móvil, donde se puede visualizar la información personal tanto del cliente como del distribuidor en sus respectivos módulos, como se muestra en detalle en la Figura 67.

Figura 67

Pruebas, visualizar Información personal (cliente, distribuidor)

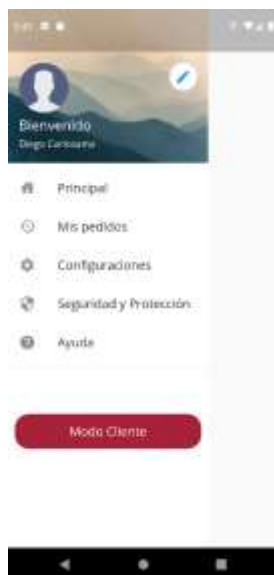


Tabla 44

Resumen de pruebas realizadas backend

Nombre de la Prueba	Descripción	Éxito (SI/NO)
Backend		
Iniciar sesión	Facilita a cualquier usuario la capacidad de iniciar sesión y acceder a la pantalla principal.	SI
Registrarse	Permite a cualquier usuario registrarse en la aplicación móvil y obtener los permisos necesarios para utilizarla.	SI
Registro adicional distribuidor	Permite registrar la información adicional necesaria para utilizar el módulo de distribuidor.	SI
Realizar pedido	Permite al cliente solicitar el servicio de entrega de gas a domicilio.	SI
Recepción de solicitudes de pedido	Permite a los distribuidores recibir los pedidos de los clientes en tiempo real.	SI
Aceptar pedido	Permite a los distribuidores aceptar los pedidos de las diversas solicitudes que aparecerán en su lista de pedidos.	SI

Editar información personal	Permite a los clientes y distribuidores la opción de editar su información personal a su conveniencia.	SI
Editar número de teléfono	Permite a los clientes y distribuidores la opción de actualizar su número de teléfono según sea necesario.	SI
Funcionalidad del Api Rest		SI

En el resumen de las pruebas realizadas a las funcionalidades del backend, se llevaron a cabo un total de nueve pruebas, todas las cuales resultaron exitosas, con un margen de éxito del 100%, como se puede apreciar en la figura 68.

Figura 68

Gráfico del resultado pruebas realizas backend



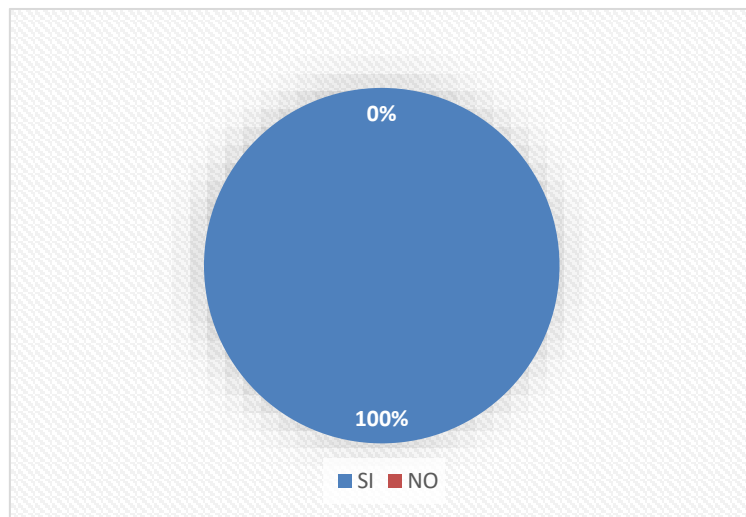
Tabla 45*Resumen de pruebas realizadas frontend*

Frontend		
Visualizar mapa con su ubicación	Permite a clientes visualizar su ubicación actual.	SI
Visualizar la ruta para la entrega de gas	Permite a clientes y distribuidores visualizar una ruta desde la ubicación del cliente hasta el distribuidor.	SI
Visualizar lista de pedidos por cliente	Permite visualizar todos los pedidos completados por parte del cliente.	SI
Visualizar lista de pedidos por distribuidor	Permite visualizar todos los pedidos completados por parte del distribuidor	SI
Visualizar nueva ubicación	Permite visualizar una nueva ubicación alterna a su ubicación actual.	SI
Visualizar ubicación del distribuidor en la vista del cliente	La ubicación del distribuidor se puede visualizar en tiempo real en la vista del cliente.	SI
Visualizar información personal	Posibilita a clientes y distribuidores la visualización de su información personal.	SI

En el resumen de las pruebas realizadas a las funcionalidades del frontend, se llevaron a cabo un total de siete pruebas, todas las cuales resultaron exitosas, con un margen de éxito del 100%, como se puede apreciar en la figura 69.

Figura 69

Gráfico del resultado pruebas realizadas backend



CAPITULO 4

4 Análisis de Usabilidad y Eficiencia en Aplicación Móvil con ISO 25010

En este capítulo se aborda dos temas importantes para la evaluación y seguimiento de buenas prácticas en el entorno de desarrollo de las aplicaciones móviles, esta evaluación se centró a dos subtemas principales, el de usabilidad y la eficiencia de desempeño.

En el tema 4.1 se habló de la Usabilidad, se detalló temas enfocados al mejorar la experiencia al usuario cuando utilice la aplicación, como: los colores, el tamaño de letras y los botones. Mientras que en el tema 4.2 se habló de la eficiencia de desempeño, se expuso la aplicación a cargas de trabajo prolongadas, la cantidad de usuarios que interactúan al mismo tiempo y los recursos que necesita para la operabilidad en los dispositivos.

4.1 Evaluación de Cumplimiento de Usabilidad

La funcionalidad y el rendimiento adecuados de una aplicación móvil son fundamentales para su éxito. Si los usuarios encuentran problemas, es probable que la desinstalen de inmediato. Según estudios, el 52% de los usuarios de dispositivos móviles experimentan dificultades con las aplicaciones y es poco probable que las sigan utilizando. Por lo tanto, es esencial que las empresas realicen pruebas antes de lanzar una aplicación para garantizar una experiencia satisfactoria para los usuarios (Arévalo & Mirón, 2017). Esto implica pruebas metodológicas que verifiquen la usabilidad de la aplicación y la implementación de características específicas en las vistas para mejorar su funcionamiento.

4.1.1 Evaluación de Pruebas Metodológicas de Usabilidad

Existen tres categorías de métodos que se pueden emplear para evaluar la usabilidad: inspección, indagación y pruebas (Mahrin et al., 2009). Dentro de la categoría de inspección se incluyen la evaluación heurística, el recorrido cognitivo y la inspección basada en estándares (Villota & Herrera, 2021), siendo esta última adecuada para evaluar la importancia de la usabilidad dentro de la calidad de software (Norma ISO 25010 - sección Usabilidad). Se utilizó dos metodologías que ayudaron a mejorar este proceso, la metodología de los 10 Principios Heurísticos de Jacob Nielsen y La Guía Fundamental Para La Usabilidad Mobile de la página web Designers.

Figura 70

Metodologías para evaluar la usabilidad

VARIABLES DE USABILIDAD SEGÚN ISO 25010	VALIDACIONES		
	LOS 10 PRINCIPIOS HEURÍSTICOS (PH) DE JACOB NIELSEN	LA GUÍA FUNDAMENTAL PARA LA USABILIDAD MÓVIL DE LA PÁGINA WEB DESIGNERS	MATERIAL DESIGN
INTELIGIBILIDAD	#2: PH Coincidencia entre el sistema y el mundo real	The Fundamental Guide to Mobile Usability	Applying color to UI and button layout
APRENDIZAJE	#1: PH Visibilidad del estado del sistema		
OPERABILIDAD	#3: PH Control y libertad del usuario		
PROTECCIÓN FRENTE A ERRORES	#5: PH Prevención de errores		
ESTÉTICA	#8: Diseño estético minimalista		
ACCESIBILIDAD (COLORIMETRÍA)			

Nota. Fuente: Elaboración propia y adaptado de (Esoldo Christine, 2023b).

4.1.2 Inteligibilidad

Para cumplir con este principio y aportar a que la norma ISO 25010, sección usabilidad, Nielsen recomienda que el diseño de una interfaz se comunique de manera efectiva con los usuarios. La forma en que debe tener la aplicación depende mucho de sus usuarios específicos (Nielsen, 2020).

Tabla 46

Segundo principio heurístico de usabilidad

-
- Utilizar un lenguaje que sea comprensible y familiar para ellos, evitando el uso de términos técnicos o jerga interna. Además, es recomendable seguir las convenciones y patrones que se encuentran en el mundo real, presentando la información de manera ordenada y lógica, siguiendo un flujo natural.
-
- Los usuarios pueden desconocer o confundirse con los términos, conceptos, íconos e imágenes que son claros para usted y sus colegas.
-

-
- La conformidad de los controles de diseño a las convenciones del mundo real y su correspondencia con los resultados esperados (conocido como mapeo natural) facilita el aprendizaje y la memorización de los usuarios sobre el funcionamiento de la interfaz. Esto contribuye a crear una experiencia que se percibe como intuitiva.
 - Se aseguro que los usuarios puedan comprender el significado de una palabra sin tener que buscar su definición.
 - No asuma que su entendimiento de las palabras o conceptos será igual al de sus usuarios.
 - La investigación de usuarios descubrirá la terminología familiar de sus usuarios, así como sus modelos mentales en torno a conceptos importantes.
-

Nota. Fuente: adaptado de (Nielsen, 2020).

Par este principio se toma en cuenta desde el inicio de la aplicación en donde con un solo clic en la pantalla de la aplicación avanza y se crea un nuevo usuario con el número celular activo, esto para asegurar que no se creen cuentas que no se encuentren activos los números de celular, ya sea por bloqueo de numero o perdidas de la línea del celular.

4.1.1 Aprendizaje

En este punto se destaca la importancia de la visibilidad del estado de la aplicación, que se refiere a la efectividad con la que se comunica el estado de la aplicación a los usuarios (Nielsen, 2020).

Tabla 47

Principio Heurístico de la Visibilidad del estado del sistema

-
- El diseño siempre debe mantener informados a los usuarios sobre lo que está sucediendo, a través de comentarios apropiados dentro de un período de tiempo razonable.
-

-
- Cuando los usuarios conocen el estado actual del sistema, aprenden el resultado de sus interacciones anteriores y determinan los próximos pasos. Las interacciones predecibles crean confianza en el producto y en la marca.
 - Comunique claramente a los usuarios cuál es el estado del sistema: no se debe realizar ninguna acción con consecuencias para los usuarios sin informarles.
 - Presente comentarios al usuario lo más rápido posible (idealmente, de inmediato).
 - Genere confianza a través de una comunicación abierta y continua.
-

Nota. Fuente: adaptado de (Nielsen, 2020).

4.1.2 Operabilidad

En este ítem se consideró como la aplicación debe funcionar en diferentes dispositivos y entornos. Entre ellos destácala la facilidad de uso, la comprensibilidad, interfaces intuitivas y funcionalidades claras.

Tabla 48

Control y libertad del usuario

-
- Los usuarios suelen realizar acciones por error. Necesitan una "salida de emergencia" claramente marcada para salir de la acción no deseada sin tener que pasar por un proceso prolongado.
-
- La aplicación debe permitir a las personas dar marcha atrás en un proceso o deshacer una acción, se fomenta una sensación de libertad y confianza. Las salidas permiten a los usuarios mantener el control del sistema y evitar quedarse atascados y sentirse frustrados.
 - La aplicación debe admitir Deshacer y Rehacer.
 - La aplicación debe mostrar una forma clara de salir de la interacción actual, como un botón Cancelar.
 - Asegúrese de que la salida esté claramente etiquetada y sea reconocible.
-

Nota. Fuente: adaptado de (Nielsen, 2020).

4.1.3 *Protección frente a errores*

En este tema se mencionó las actividades comunes que el usuario realiza al usar una aplicación y que se debe prevenir para evitar errores, es una forma de brindar una experiencia satisfactoria, estable y segura al usuario.

Tabla 49

Quinto principio heurístico, protección de errores

-
- La aplicación debe contener un control de mensajes de error importantes.
-
- La aplicación debe presentar a los usuarios una opción de confirmación antes de que se comprometan con la acción.
 - La aplicación debe estar preparada a dos categorías de errores: deslices y equivocaciones. Los deslices son errores que ocurren de forma inconsciente debido a la falta de atención. Por otro lado, las equivocaciones son errores conscientes que surgen cuando hay una discrepancia entre el modelo mental del usuario y el diseño.
 - Evite los deslices proporcionando restricciones útiles y buenos valores predeterminados.
 - Evite errores eliminando cargas de memoria, admitiendo deshacer y advirtiendo a sus usuarios.
-

Nota. Fuente: adaptado de (Nielsen, 2020).

4.1.4 *Estética*

En este tema se destacó la forma visual con la que percibe el usuario. La tipografía, los elementos, los espacios, los contenedores y la disposición de los elementos.

Tabla 50

Principio heurístico ocho, Diseño estético minimalista

-
- Las interfaces de la aplicación no deben contener información que sea irrelevante o que rara vez se necesite.
-
- Esta heurística no implica que debas utilizar un diseño plano; su objetivo es asegurarse de que el contenido y el diseño visual se centren en lo esencial. Es importante que los elementos visuales de la interfaz apoyen los principales objetivos del usuario.
 - La aplicación debe mantener el contenido y el diseño visual de la interfaz de usuario centrado en lo esencial.
-

-
- La aplicación no debe permitir que elementos innecesarios distraigan a los usuarios de la información que realmente necesitan.
 - La aplicación debe priorizar el contenido y las funciones para respaldar los objetivos principales.
-

Nota. Fuente: adaptado de (Nielsen, 2020).

4.1.3 Accesibilidad (Colorimetría)

Si bien es cierto la accesibilidad es muy amplia, en este caso se ha considerado la colorimetría de la aplicación. Ya que la accesibilidad pretende que la aplicación debe ser inclusiva en muchos factores y es muy amplio el campo.

Tabla 51

Tabla de ítems para la colorimetría adecuada de la aplicación móvil

<p>Proporciones del esquema de color:</p> <p>60-30-10</p>	<p>Es una norma que suelen usar los diseñadores de interiores y decoradores para escoger, estructurar y darles proporción a los colores escogidos. En este caso, podemos aplicarlo para diseñar el “interior” de la aplicación móvil.</p> <ul style="list-style-type: none"> - 60% es la proporción del color dominante, en nuestro caso pasaría a ser el color de fondo, en muchos casos, el color blanco o gris claro. - 30% proporción del secundario, el color más estacado de la aplicación. - 10% el “accent color”, para dar un toque de color diferente y complementario/opuesto a los demás, normalmente suele usarse para botones de confirmación o para añadir algún elemento.
<p>Aplicar y luego iterar</p>	<ul style="list-style-type: none"> - Una vez que haya utilizado la regla 60-30-10, modifique sus colores para mejorar la estética y también la prominencia de lo que es importante en su diseño. No existe un color estándar,

<p>Use colores consistentemente en su interfaz</p>	<p>cada color individual se agrega dependiendo del producto al que se enfoca.</p> <ul style="list-style-type: none"> - Si se usa azul brillante para las llamadas a la acción en una pantalla, ese mismo color debe usarse para las llamadas a la acción en todas partes. - Si usa el rojo como color de advertencia en una pantalla, no debe usarse para significar algo diferente en otro lugar. - La consistencia es clave para ayudar a sus usuarios a comprender el uso del color.
---	--

Nota. Fuente: adaptado de (Gordon, 2021).

Evaluación experimental de usabilidad

En este subtema se detalla cómo se realizó las pruebas de Usabilidad experimentales en usuarios reales con la ayuda de un el software Maze, que es una plataforma líder en pruebas rápidas que capacita a los equipos ágiles para realizar pruebas, adquirir conocimientos y tomar medidas de manera ágil y eficiente (Capterra, 2021). Para realizar esta prueba se inició con la selección del número de personas que sugiere la herramienta <5 sugiere que el nivel de confianza es bajo, entre 5-20 se encuentra en un nivel aceptable y se identifique los problemas comunes, entre 21-100 se presentaran datos más confiables porque se identificaran patrones y >100 ya se tiene un nivel alto de evaluación. creación de la tarea en la herramienta Maze, para esto se debe tener el prototipo de Adobe XD organizado por secciones que necesite ser evaluada, en este caso se evaluó el **inicio de sesión, configuración, realizar un pedido, cancelar el pedido, ver historial de pedidos y buscar las direcciones**. Estas tareas fueron escritas en la herramienta para que el usuario lea y cumpla con lo que se pide, el evaluador da las instrucciones correctamente y se procedió a evaluar al voluntario, si no encuentra el camino, se le dificulta realizar la tarea se registra los datos para luego ser analizados y realizar mejoras en el diseño de la aplicación y que el usuario no tenga un nivel alto de frustración cuando utilice la aplicación de pedido de gas.

Tabla 52

Análisis de las tareas de usabilidad

Tarea	Descripción	Resultados
Inicio de sesión	Inicie sesión en la aplicación con su número de celular y agregue sus datos personales.	-5% clic fuera de lugar. -28.0s del tiempo promedio que se lograron logear. -100% d ellos usuario terminaron la prueba. -0% de los usuarios abandonaron la prueba.
Configuración	En el menú de configuraciones, cambie el número de celular y guarde.	-5% clic fuera de lugar. -38.0s del tiempo promedio que se lograron cambiar. -100% d ellos usuario terminaron la prueba. -0% de los usuarios abandonaron la prueba.
Realizar un pedido	Realice un pedio de gas a su dirección actual.	-25% clic fuera de lugar. -20.0s del tiempo promedio que se lograron realizar el pedido. -100% d ellos usuario terminaron la prueba. -0% de los usuarios abandonaron la prueba.
Cancelar el pedido	Cancele el pedido que realizo anteriormente.	-2% clic fuera de lugar. -5.0s del tiempo promedio que se lograron cancelar el pedido. -95% de los usuarios terminaron la prueba. -5% de los usuarios abandonaron la prueba.
Ver historial de pedidos	Revise el historial de pedidos que ha realizado en la aplicación.	-5% clic fuera de lugar. -10.0s del tiempo promedio que se lograron ver el historial de pedidos. -100% de los usuarios terminaron la prueba. -0% de los usuarios abandonaron la prueba.
Menú de seguridad	Encuentre la opción de números de ayuda y seguridad.	-10% clic fuera de lugar. -10s del tiempo promedio que se lograron llegar al menú de seguridad. -100% d ellos usuario terminaron la prueba.

-0% de los usuarios abandonaron la prueba.

Nota. Fuente. Adaptado de: (Capterra, 2021; Aguilera G., 2021).

Al finalizar las pruebas la herramienta presentó una serie de información e imágenes de mapas de calor que indican en donde las personas dieron los clics erróneos, esto sirvió para el desarrollador mejora el tamaño de las letras, la ubicación y tamaño de los botones entre otras.

4.2 Evaluación de eficiencia de desempeño

En esta sección, se realiza una evaluación exhaustiva de la API REST utilizada en la aplicación móvil de distribución de gas doméstico, con un enfoque específico en las subcaracterística de eficiencia y desempeño. El propósito principal es analizar el rendimiento y la eficiencia de la API en diversas situaciones de carga de trabajo, con el fin de garantizar un funcionamiento óptimo. Para realizar estas pruebas, utilizaremos software especializado que facilite el proceso. Con el objetivo de determinar la herramienta más adecuada, se realizó un análisis comparativo de varias opciones, incluyendo Apache JMeter, LoadRunner y Gatling. Los resultados y detalles de esta comparativa se presentan en la Tabla 63.

Tabla 53

Características de las herramientas para evaluación de software

Característica	Apache JMeter	LoadRunner	Gatling
Escalabilidad	Alta escalabilidad y capacidad para manejar grandes volúmenes de carga.	Puede manejar cargas masivas de usuarios simultáneos.	Eficiencia en cuanto a recursos, puede manejar cargas intensivas con un impacto mínimo.
Flexibilidad y personalización	Amplia gama de plugins y extensiones para personalizar y extender sus funcionalidades.	Ofrece opciones de configuración y personalización.	Personalizable con capacidad de definir escenarios de prueba.

Interfaz de usuario	de Interfaz gráfica de usuario intuitiva y fácil de usar.	Interfaz amigable con herramientas visuales para facilitar la creación y ejecución de pruebas.	Requiere conocimientos técnicos más avanzados debido a su enfoque basado en código.
Soporte y documentación	Amplia documentación oficial y comunidad activa de usuarios.	Soporte técnico dedicado.	Comunidad activa que proporciona soporte y recursos.

Después de analizar la tabla comparativa, se ha determinado que Apache JMeter es la herramienta más adecuada para realizar las pruebas de eficiencia y desempeño en la aplicación móvil de distribución de gas doméstico. Apache JMeter destaca por su alta escalabilidad, lo que significa que puede manejar grandes volúmenes de carga de manera eficiente. Además, ofrece una amplia gama de plugin y extensiones que permiten una gran flexibilidad y personalización en la creación de pruebas. Su interfaz gráfica de usuario intuitiva y fácil de usar facilita el proceso de configuración y ejecución de las pruebas. Además, cuenta con una amplia documentación oficial y una comunidad activa de usuarios que proporcionan soporte y recursos adicionales. Todo esto, sumado a su condición de herramienta de código abierto, con actualizaciones y mejoras constantes, hacen que Apache JMeter sea la elección ideal para garantizar un funcionamiento óptimo de la API REST en diferentes situaciones de carga de trabajo.

En la Ciudad de Ibarra, según un informe de (La Hora, 2019), se realiza la distribución diaria de 3000 cilindros de gas doméstico. Esta cifra demuestra la relevancia y la demanda que existe en la comunidad ibarreña por este servicio esencial.

Con base en la información recopilada, se plantea el diseño de escenarios de prueba utilizando JMeter para evaluar la eficiencia de desempeño de la API REST para la distribución de gas doméstico en la ciudad de Ibarra. En estos escenarios, se empleará una aproximación de usuarios virtuales, considerando cada solicitud de cilindro como un usuario en las pruebas.

Para distribuir la carga de trabajo de manera real, se tendrán en cuenta diferentes factores. En primer lugar, se considerará el número de cilindros de gas doméstico distribuidos diariamente, que es de 3000. Además, se tomará en cuenta la información obtenida de

entrevistas realizadas en el capítulo 1, donde se menciona que la venta promedio de cilindros por distribuidor oscila entre 40 y 60 unidades.

Con estos datos, se presentan los siguientes escenarios de prueba:

Escenario 1: Carga máxima con 3000 usuarios.

Escenario 2: Carga promedio con 1500 usuarios

Escenario 3: Carga mínima con 500 usuarios

En la planificación de pruebas de eficiencia y rendimiento, es esencial asignar ciertas funcionalidades clave a cada escenario, de acuerdo con criterios específicos que reflejen su importancia y su impacto en la experiencia del usuario y en el funcionamiento general del negocio. Estas funcionalidades requerirán una evaluación exhaustiva.

La selección de estas funcionalidades se rige por tres criterios fundamentales que permiten una priorización objetiva y fundamentada:

- **Valor de Negocio:** Evalúa el aporte de cada funcionalidad a los objetivos del negocio y a la generación de ingresos. Se califican más alto las funcionalidades que tienen un impacto directo en el modelo de negocio y en la obtención de beneficios.
- **Riesgo:** El criterio de riesgo se enfoca en la probabilidad de problemas o fallos asociados a la implementación y operación de una funcionalidad. Se otorgan calificaciones más altas a las funcionalidades cuyo mal funcionamiento podría tener un impacto significativo en la seguridad, integridad o experiencia del usuario.
- **Frecuencia de Uso:** Este criterio evalúa cuán a menudo los usuarios interactúan con cada funcionalidad. Se otorgan calificaciones más altas a las funcionalidades utilizadas con mayor frecuencia, ya que tienen un impacto más pronunciado en la experiencia del usuario.

Estos criterios proporcionan una base sólida para priorizar las pruebas de eficiencia y rendimiento en las funcionalidades más críticas.

Tabla 54

Priorización de Funcionalidades

Funcionalidad	Valor de Negocio	Riesgo	Frecuencia de Uso
----------------------	-------------------------	---------------	--------------------------

Iniciar Sesión	5	3	5
Realizar Pedido	5	3	4
Aceptar Pedido	4	3	3
Registrarse	4	3	3
Registro Adicional			
Distribuidor	3	3	2
Recepción de			
Solicitudes de	4	3	3
Pedido			
Editar información			
personal	3	2	3
Editar Número de			
Teléfono	2	2	2

Tras una cuidadosa evaluación como se observa en la figura 53 de las funcionalidades de la aplicación utilizando los criterios de Valor de Negocio, Riesgo y Frecuencia de Uso, se ha identificado que las funcionalidades prioritarias para someter a pruebas de eficiencia y rendimiento son las siguientes:

- **Iniciar Sesión:** Esta funcionalidad ha demostrado ser crítica para el negocio, ya que tiene un alto impacto en la retención de usuarios y en la seguridad de la aplicación. Además, los usuarios la utilizan con una alta frecuencia. Garantizar su eficiencia es fundamental para una experiencia de usuario óptima.
- **Realizar Pedido:** La capacidad de realizar pedidos es esencial para el modelo de negocio, generando ingresos directos. Los usuarios la utilizan con frecuencia, y cualquier problema con esta funcionalidad podría afectar negativamente las ventas y la satisfacción del cliente.
- **Aceptar Pedido:** Esta funcionalidad es importante tanto para los distribuidores como para la satisfacción del cliente. Aunque se utiliza con menos frecuencia que las anteriores, su impacto en la operatividad y la satisfacción del cliente la convierte en una funcionalidad crítica.

Una vez que se han definido los escenarios y las funcionalidades principales para su evaluación, se procede de la siguiente manera: La norma ISO 25010 define la característica de eficiencia de desempeño, que consta de diversas subcaracterísticas. En este caso, se enfocará en dos de ellas, el comportamiento temporal y la capacidad. Estas subcaracterísticas serán el enfoque principal de las pruebas de eficiencia.

4.2.1 Escenario 1

4.2.1.1 Iniciar sesión

Figura 71

Escenario 1 – Iniciar sesión

Label	# Samples	Average	Min	Max	Std. Dev.	Min %	Throughput	Received KB/s	Sent KB/s
Iniciar sesión	3000	498	0	3362	118,19	0,00%	14,89884	5,12	4,03
TOTAL	3000	498	0	3362	118,19	0,00%	14,89884	5,12	4,03

Fuente: Elaboración propia.

Los resultados que se muestran en la figura 71 revelan un comportamiento temporal eficiente en la acción de inicio de sesión. El tiempo de respuesta promedio de 498 milisegundos, con un tiempo mínimo de 0 milisegundos y un tiempo máximo de 3362 milisegundos, demuestran una respuesta rápida y aceptable del sistema ante las solicitudes de inicio de sesión. Aunque se observa cierta variabilidad en los tiempos de respuesta, con una desviación estándar de 118,19 milisegundos, la estabilidad del sistema queda respaldada por la ausencia de errores registrados durante las 3000 muestras realizadas. Estos resultados respaldan un comportamiento temporal eficiente y confiable en el proceso de inicio de sesión.

Además, los resultados indican una capacidad sólida en la acción de inicio de sesión. Con un promedio de 14,89884 solicitudes por segundo, el sistema demuestra su capacidad para manejar una carga considerable de solicitudes de inicio de sesión. Las tasas de transferencia de datos recibidos y enviados, de 5,12 KB/s y 4,03 KB/s respectivamente, resaltan la eficiente transferencia de datos durante el proceso. Estos hallazgos respaldan la confiabilidad y eficiencia del sistema en términos de capacidad, asegurando una experiencia fluida y satisfactoria para los usuarios durante el proceso de inicio de sesión.

4.2.1.2 Solicitar pedido

Figura 72

Escenario 1 – Solicitar pedido



Label	# Samples	Average	Min	Max	Std. Dev.	Err. %	Throughput	Received KB/s	Sent KB/s
Solicitar pedido	3000	171	0	746	42,70	0,00%	14,91req	4,87	7,81
TOTAL	3000	171	0	746	42,70	0,00%	14,91req	4,87	7,81

Fuente: Elaboración propia.

Los resultados que se muestra en la figura 72 revelan un comportamiento temporal eficiente en la acción de realizar pedidos. Con un tiempo de respuesta promedio de 171 milisegundos, con un tiempo mínimo de 0 milisegundos y un tiempo máximo de 746 milisegundos, el sistema muestra una respuesta rápida y aceptable frente a las solicitudes de pedidos. La desviación estándar de 42,70 milisegundos indica una cierta variabilidad en los tiempos de respuesta. Además, no se registraron errores durante las 3000 muestras realizadas, lo que destaca la estabilidad y confiabilidad del sistema en el proceso de realización de pedidos.

En términos de capacidad, los resultados indican un promedio de 14,91780 solicitudes por segundo, lo que demuestra la capacidad del sistema para manejar una carga considerable de solicitudes de pedidos. Además, las tasas de transferencia de datos recibidos y enviados fueron de 4,87 KB/s y 7,81 KB/s, respectivamente, mientras que el tamaño promedio de bytes fue de 334,0 bytes. Estos hallazgos respaldan la eficiencia del sistema en términos de capacidad de procesamiento y transferencia de datos durante el proceso de realizar pedidos.

4.2.1.3 Aceptar pedido

Figura 73

Escenario 1 – Aceptar pedido



Label	# Samples	Average	Min	Max	Std. Dev.	Err. %	Throughput	Received KB/s	Sent KB/s
Aceptar pedido	3000	329	0	861	85,78	0,00%	14,91req	4,87	7,81
TOTAL	3000	329	0	861	85,78	0,00%	14,91req	4,87	7,81

Fuente: Elaboración propia.

Los resultados que se muestran en la figura 73 revelan un comportamiento temporal eficiente en la acción de aceptar pedidos. Con un tiempo de respuesta promedio de 329 milisegundos, con un tiempo mínimo de 0 milisegundos y un tiempo máximo de 861 milisegundos, el sistema muestra una respuesta rápida y aceptable frente a las solicitudes de aceptar pedidos. La desviación estándar de 85,78 milisegundos indica una cierta variabilidad

en los tiempos de respuesta. Además, no se registraron errores durante las 3000 muestras realizadas, lo que destaca la estabilidad y confiabilidad del sistema en el proceso de aceptación de pedidos.

En términos de capacidad, los resultados indican un promedio de 14,91128 solicitudes por segundo, lo que demuestra la capacidad del sistema para manejar una carga considerable de solicitudes de aceptación de pedidos. Además, las tasas de transferencia de datos recibidos y enviados fueron de 4,82 KB/s y 7,48 KB/s, respectivamente, mientras que el tamaño promedio de bytes fue de 331,0 bytes. Estos hallazgos respaldan la eficiencia del sistema en términos de capacidad de procesamiento y transferencia de datos durante el proceso de aceptar pedidos.

4.2.2 Escenario 2

4.2.2.1 Iniciar sesión

Figura 74

Escenario 2 – Iniciar sesión

Label	# Samples	Average	Min	Max	Std. Dev.	Unit %	Throughput	Received KB/s	Sent KB/s
Inicio sesión	1500	545	0	2626	261,86	0,0%	1,4/s	2,56	2,01
TOTAL	1500	545	0	2626	261,86	0,0%	1,4/s	2,56	2,01

Fuente: Elaboración propia.

Los resultados que se muestran en la figura 74 revelan un comportamiento temporal eficiente en la acción de inicio de sesión. Con un tiempo de respuesta promedio de 545 milisegundos, con un tiempo mínimo de 0 milisegundos y un tiempo máximo de 2626 milisegundos, el sistema muestra una respuesta aceptable frente a las solicitudes de inicio de sesión. La desviación estándar de 261,86 milisegundos indica cierta variabilidad en los tiempos de respuesta. Además, no se registraron errores durante las 1500 muestras realizadas, lo que respalda la confiabilidad del sistema en el proceso de inicio de sesión.

En términos de capacidad, los resultados indican un promedio de 7,44524 solicitudes por segundo, lo que demuestra la capacidad del sistema para manejar una carga adecuada de solicitudes de inicio de sesión. Además, las tasas de transferencia de datos recibidos y enviados fueron de 2,56 KB/s y 2,01 KB/s, respectivamente, mientras que el tamaño promedio de bytes fue de 352,0 bytes. Estos hallazgos respaldan la eficiencia del sistema en términos de capacidad de procesamiento y transferencia de datos durante el proceso de inicio de sesión.

4.2.2.2 Solicitar pedido

Figura 75

Escenario 2 – Solicitar pedido

Label	# Samples	Average	Min	Max	Std. Dev.	Err %	Throughput	Received KB/s	Sent KB/s
Solicitar pedido	1500	341	0	2230	207,17	0,00%	7.51458	2,45	3,93
100%	1500	341	0	2230	207,17	0,00%	7.51458	2,45	3,93

Fuente: Elaboración propia.

Los resultados que se muestran en la figura 75 revelan un comportamiento temporal eficiente en la acción de realizar pedidos. Con un tiempo de respuesta promedio de 243 milisegundos, con un tiempo mínimo de 0 milisegundos y un tiempo máximo de 2230 milisegundos, el sistema muestra una respuesta rápida y aceptable frente a las solicitudes de realizar pedidos. La desviación estándar de 207,17 milisegundos indica cierta variabilidad en los tiempos de respuesta. Además, no se registraron errores durante las 1500 muestras realizadas, lo que respalda la confiabilidad del sistema en el proceso de realizar pedidos.

En términos de capacidad, los resultados indican un promedio de 7,51458 solicitudes por segundo, lo que demuestra la capacidad del sistema para manejar una carga adecuada de solicitudes de realizar pedidos. Además, las tasas de transferencia de datos recibidos y enviados fueron de 2,45 KB/s y 3,93 KB/s, respectivamente, mientras que el tamaño promedio de bytes fue de 334,0 bytes. Estos hallazgos respaldan la eficiencia del sistema en términos de capacidad de procesamiento y transferencia de datos durante el proceso de realizar pedidos.

4.2.2.3 Aceptar pedido

Figura 76

Escenario 2 – Aceptar pedido

Label	# Samples	Average	Min	Max	Std. Dev.	Err %	Throughput	Received KB/s	Sent KB/s
Aceptar pedido	1500	364	0	2064	155,70	0,00%	7.31458	2,45	3,75
100%	1500	364	0	2064	155,70	0,00%	7.31458	2,45	3,75

Fuente: Elaboración propia.

Los resultados que se muestran en la figura 76 revelan un comportamiento temporal eficiente en la acción de aceptar pedidos. Con un tiempo de respuesta promedio de 364 milisegundos, con un tiempo mínimo de 0 milisegundos y un tiempo máximo de 2064 milisegundos, el sistema muestra una respuesta rápida y aceptable frente a las solicitudes de aceptar pedidos. La desviación estándar de 155,70 milisegundos indica cierta variabilidad en

los tiempos de respuesta. Además, no se registraron errores durante las 1500 muestras realizadas, lo que respalda la confiabilidad del sistema en el proceso de aceptación de pedidos.

En términos de capacidad, los resultados indican un promedio de 7,51326 solicitudes por segundo, lo que demuestra la capacidad del sistema para manejar una carga adecuada de solicitudes de aceptar pedidos. Además, las tasas de transferencia de datos recibidos y enviados fueron de 2,43 KB/s y 3,77 KB/s, respectivamente, mientras que el tamaño promedio de bytes fue de 331,0 bytes. Estos hallazgos respaldan la eficiencia del sistema en términos de capacidad de procesamiento y transferencia de datos durante el proceso de aceptar pedidos.

4.2.3 Escenario 3

4.2.3.1 Iniciar sesión

Figura 77

Escenario 3 – Iniciar sesión

Label	# Samples	Average	Min	Max	Std. Dev	Unit/s	Throughput	Received KB/s	Sent KB/s
Iniciar sesión	500	541	0	1127	100,14	0,00%	2,53/s	0,86	0,68
TOTAL	500	541	0	1127	100,14	0,00%	2,53/s	0,86	0,68

Fuente: Elaboración propia.

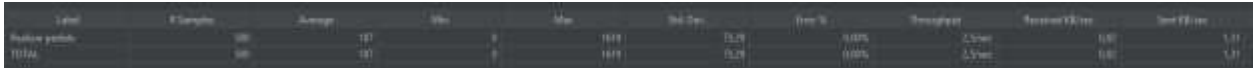
Los resultados que se muestran en la figura 77 revelan un comportamiento temporal eficiente en la acción de inicio de sesión. Con un tiempo de respuesta promedio de 541 milisegundos, con un tiempo mínimo de 0 milisegundos y un tiempo máximo de 1127 milisegundos, el sistema muestra una respuesta aceptable frente a las solicitudes de inicio de sesión. La desviación estándar de 100,14 milisegundos indica cierta variabilidad en los tiempos de respuesta. Además, no se registraron errores durante las 500 muestras realizadas, lo que respalda la confiabilidad del sistema en el proceso de inicio de sesión.

En términos de capacidad, los resultados indican un promedio de 2,49928 solicitudes por segundo, lo que demuestra la capacidad del sistema para manejar una carga adecuada de solicitudes de inicio de sesión. Además, las tasas de transferencia de datos recibidos y enviados fueron de 0,86 KB/s y 0,68 KB/s, respectivamente, mientras que el tamaño promedio de bytes fue de 352,0 bytes. Estos hallazgos respaldan la eficiencia del sistema en términos de capacidad de procesamiento y transferencia de datos durante el proceso de inicio de sesión.

4.2.3.2 Solicitar pedido

Figura 78

Escenario 3 – Solicitar pedido



Label	# Samples	Average	Min	Max	Std. Dev.	Err %	Throughput	Received Kbits	Send Kbits
Solicitar pedido	500	187	0	1619	73,29	0,00%	2,50sec	0,82	1,31
TOTAL	500	187	0	1619	73,29	0,00%	2,50sec	0,82	1,31

Fuente: Elaboración propia.

Los resultados que se muestran en la figura 78 revelan un comportamiento temporal eficiente en la acción de realizar pedidos. Con un tiempo de respuesta promedio de 187 milisegundos, con un tiempo mínimo de 0 milisegundos y un tiempo máximo de 1619 milisegundos, el sistema muestra una respuesta rápida y aceptable frente a las solicitudes de realizar pedidos. La desviación estándar de 73,29 milisegundos indica cierta variabilidad en los tiempos de respuesta. Además, no se registraron errores durante las 500 muestras realizadas, lo que respalda la confiabilidad del sistema en el proceso de realizar pedidos.

En términos de capacidad, los resultados indican un promedio de 2,50262 solicitudes por segundo, lo que demuestra la capacidad del sistema para manejar una carga adecuada de solicitudes de realizar pedidos. Además, las tasas de transferencia de datos recibidos y enviados fueron de 0,82 KB/s y 1,31 KB/s, respectivamente, mientras que el tamaño promedio de bytes fue de 334,0 bytes. Estos hallazgos respaldan la eficiencia del sistema en términos de capacidad de procesamiento y transferencia de datos durante el proceso de realizar pedidos.

4.2.3.3 Aceptar pedido

Figura 79

Escenario 3 - Aceptar pedido



Label	# Samples	Average	Min	Max	Std. Dev.	Err %	Throughput	Received Kbits	Send Kbits
Aceptar pedido	500	361	0	719	82,79	0,00%	2,30sec	0,82	1,31
TOTAL	500	361	0	719	82,79	0,00%	2,30sec	0,82	1,31

Fuente: Elaboración propia.

Los resultados que se muestra en la figura 79 revelan un comportamiento temporal eficiente en la acción de aceptar pedidos. Con un tiempo de respuesta promedio de 361 milisegundos, con un tiempo mínimo de 0 milisegundos y un tiempo máximo de 719 milisegundos, el sistema muestra una respuesta rápida y aceptable frente a las solicitudes de aceptar pedidos. La desviación estándar de 82,79 milisegundos indica cierta variabilidad en

los tiempos de respuesta. Además, no se registraron errores durante las 500 muestras realizadas, lo que respalda la confiabilidad del sistema en el proceso de aceptar pedidos.

En términos de capacidad, los resultados indican un promedio de 2,50095 solicitudes por segundo, lo que demuestra la capacidad del sistema para manejar una carga adecuada de solicitudes de aceptar pedidos. Además, las tasas de transferencia de datos recibidos y enviados fueron de 0,81 KB/s y 1,26 KB/s, respectivamente, mientras que el tamaño promedio de bytes fue de 331,0 bytes. Estos hallazgos respaldan la eficiencia del sistema en términos de capacidad de procesamiento y transferencia de datos durante el proceso de aceptar pedidos.

CAPITULO 5

5 Validación de los Resultados de la Investigación

5.1 Resultados

Además de evaluar la funcionalidad y calidad de la aplicación móvil, es esencial tener en cuenta la aceptación del cliente simulado en un entorno de producción. Este enfoque nos permite verificar si la aplicación cumple con los requisitos de los usuarios y si logra satisfacer sus expectativas. Al evaluar la aceptación del cliente, se obtiene una evaluación integral que considera tanto aspectos técnicos como la satisfacción del usuario, garantizando que la aplicación cumpla con los estándares de calidad y brinde una experiencia satisfactoria para el cliente.

Análisis de resultados

Para analizar los resultados, se utiliza como referencia el Cuestionario de Usabilidad de Sistemas Informáticos (CSUQ), que consta de 16 preguntas diseñadas para evaluar la usabilidad de la aplicación (Lewis, 2018). Estas preguntas están enfocadas en evaluar la utilidad del sistema, la calidad de la información y la calidad de las interfaces (Isolde et al., 2015).

Encuesta CSUQ

Las preguntas de este cuestionario están valoradas en la escala de Likert de 7 opciones. Según (Lewis, 2018) los tres aspectos que mide CSUQ son los siguientes:

- Utilidad del sistema (abarca las preguntas 1 - 6)
- Calidad de la información (abarca las preguntas 7 - 12)
- Calidad de las interfaces (abarca las preguntas 13 - 16)

En la TABLA 65 se muestra los valores cuantitativos de cada punto de la escala mencionada y en la TABLA 66 se muestran las 16 preguntas que componen a la encuesta CSUQ.

Tabla 55

Escala de Likert

Valor Cuantitativo	Valor Cualitativo
7	Totalmente de acuerdo

6	De acuerdo
5	Ligeramente en acuerdo
4	Neutral
3	Ligeramente en desacuerdo
2	En desacuerdo
1	Totalmente en desacuerdo

Tabla 56.

Encuesta CSUQ

Nro.	Pregunta
1	En general, estoy satisfecho con lo fácil que es utilizar esta aplicación móvil.
2	Fue simple usar esta aplicación móvil.
3	Soy capaz de completar mi trabajo rápidamente utilizando esta aplicación móvil.
4	Me siento cómodo utilizando esta aplicación móvil.
5	Fue fácil aprender a utilizar esta aplicación móvil.
6	Creo que me volví experto rápidamente utilizando esta aplicación móvil.
7	Esta aplicación móvil muestra mensajes de error que me dicen claramente como resolver los problemas.
8	Cada vez que cometo un error utilizando la aplicación móvil, lo resuelvo fácil y rápidamente.
9	La información (como ayuda en línea, mensajes en pantalla y otra documentación) que provee esta aplicación móvil es clara.
10	Es fácil encontrar en la aplicación móvil la información que necesito.
11	La información que proporciona la aplicación móvil fue efectiva ayudándome a completar las tareas.

12	La organización de la información de la aplicación móvil en la pantalla fue clara.
13	La interfaz de la aplicación móvil fue placentera.
14	Me gustó utilizar la aplicación móvil.
15	La aplicación móvil tuvo todas las herramientas que esperaba que tuviera.
16	En general, estuve satisfecho con la aplicación móvil.

Nota. Fuente: (Isolde et al., 2015).

5.2 Tamaño de Muestra

Para realiza los análisis de los encuestados se realizó un cálculo de la muestra poblacional del sector, dando como resultado 384 habitantes que se obtuvieron en las encuestas iniciales para el levantamiento de requisitos, antes de que el software fuera desarrollado o puesto a disposición. Esta muestra fue obtenida con ayuda del software Epidat. Lo resultados que arroja el software utilizan esta fórmula que sirvió para calcular la muestra. La siguiente formula es adaptada de (QuestioPro,2023):

$$n = \frac{N \times Z^2 \times p \times q}{e^2 \times (N - 1) + Z^2 \times p \times q}$$

Donde:

- n: Es la muestra obtenida
- N: Es el número de la población en este caso
- Z: Es nuestro factor de confiabilidad en este caso 1.96
- p: Es la porción de la población con las mismas características en este caso 0.5
- q: Es la porción de la población que no cumplen con las mismas características en este caso $1 - p = 0.5$
- e: Es el error de la muestra que se lo calcula mediante $1 - k = 5\% \Rightarrow 0.05$

Remplazando los datos en cada término de la formula se obtiene lo siguiente:

$$n = \frac{170549 \times (1.96)^2 \times (0.5) \times (0.5)}{0.05^2 \times (170549 - 1) + 1.96^2 \times (0.5) \times (0.5)}$$

$$n = 384$$

Utilizando el proceso de cálculo de la muestra, se pudo inferir que, de una población de 170,549 habitantes en la zona urbana de la ciudad de Ibarra, es requerido aplicar la encuesta a 384 individuos, de los cuales se llevarán a cabo los análisis estadísticos correspondientes. Para lograr una mayor exactitud se solicitó que la encuesta fuera contestada por una persona que realiza los quehaceres del hogar.

5.3 Análisis de la Encuesta CSUQ

Para realizar el análisis de la encuesta CSUQ aplicada a los usuarios de la aplicación móvil, se tiene en cuenta que los parámetros que se maneja dentro de las preguntas tienen una caracterización de la información, interfaz y con el sistema en general, de esta forma permite tener un análisis más profundo y concreto.

Por otra parte, existe una relación con las sub características de usabilidad de la norma 25010 con las preguntas CSUQ, las cuales consisten en:

- Inteligibilidad, se relaciona con las preguntas 3, 11 y 15
- Aprendizaje, tiene una conexión con las preguntas 5, 9 y 16
- Operabilidad, se asemeja con las preguntas 1, 2, 4, 6 y 10
- Protección ante errores de usuario, se relaciona con las preguntas 7 y 8
- Estética, que hace referencia a las preguntas 12, 13 y 14

Una vez definida la relación de las preguntas del cuestionario CSUQ con la norma ISO/IEC 25010, se realiza un análisis general de los resultados de la encuesta, así como se muestra en la Tabla 66. Por otra parte, para evaluar cada una de las preguntas del cuestionario CSUQ, se utiliza la escala de Likert, teniendo en cuenta que esta tiene dos versiones la que está en español que se indica en la Tabla 57 y la versión en inglés que se define a 1 como “totalmente de acuerdo” hasta 5 que es “totalmente en desacuerdo” (De la Cruz, 2022).

Tabla 57

Tabla de frecuencias de respuestas CSUQ

Respuesta	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16
-----------	----	----	----	----	----	----	----	----	----	-----	-----	-----	-----	-----	-----	-----

Totalmente de acuerdo	44	57	22	49	41	14	52	11	60	22	42	33	59	60	49	43
De acuerdo	30	28	49	28	25	44	20	49	19	45	39	42	7	20	27	36
Ligeramente en acuerdo	8	1	14	8	14	24	9	22	6	15	5	10	18	4	9	5
Neutral	4	1	0	2	7	5	6	4	1	4	1	1	2	1	1	3
Ligeramente en desacuerdo	1	0	1	0	0	0	0	0	1	0	0	1	1	1	0	0
En desacuerdo	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0	0
Totalmente en desacuerdo	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
Total	87	87	87	87	87	87	87	87	87	87	87	87	87	87	87	87

Es fundamental evaluar la confiabilidad de la encuesta CSUQ antes de llevar a cabo el análisis estadístico. La fórmula 2 del Alpha de Cronbach es conocida para realizar estos análisis, cuyo resultado indicará el nivel de confiabilidad de la encuesta realizada.

Formula 2 de Alpha de Cronbach

$$\alpha = \frac{K}{K-1} \left(1 - \frac{\sum Vi}{Vt} \right)$$

Fuente (Alonso & Santacruz, 2015)

Donde:

- K: Es el número de ítems en este caso la encuesta CSUQ tiene 16 ítems
- Vi: Es la varianza de cada ítem, de la encuesta realizada es de 6.97
- Vt: Varianza del total, de la encuesta realizada es de 18.43

Remplazando los datos en la formula se obtiene que el Alpha de Cronbach es de:

$$\alpha = \frac{16}{16-1} \left(1 - \frac{\sum 6.97}{18.43} \right)$$

$$\alpha = 0.7$$

De acuerdo con (Posso Pacheco & Bertheau, 2020) existen rangos mediante los cuales se puede interpretar el resultado del cálculo realizado:

- Menor a 0.5: La encuesta es de confiabilidad Inaceptable
- Mayor a 0.5: La encuesta es de confiabilidad Pobre
- Mayor a 0.6: La encuesta es de confiabilidad Cuestionable
- Mayor a 0.7: La encuesta es de confiabilidad Aceptable
- Mayor a 0.8: La encuesta es de confiabilidad Buena
- Mayor a 0.9: La encuesta es de confiabilidad Excelente

Tomando en cuenta estos aspectos es posible concluir que el nivel de confiabilidad de la encuesta realizada indica que las respuestas obtenidas se encuentran en el rango confiable con un 0.7 de confiabilidad mencionada antes.

En la siguiente tabla se analizó la respuesta de la encuesta, la primera columna contiene la subcaracterística que tienen con la norma ISO 25010, en la columna dos el número de la pregunta y la columna tres es el análisis de los resultados.

Tabla 58

Análisis de encuesta CSUQ e interpretación de resultados

Subcaracterísticas ISO 25010	Preguntas relacionadas	Análisis
Inteligibilidad	3	El 59% de los participantes de la encuesta expresó su total acuerdo con la finalización rápida del trabajo, mientras que el 41% manifestó su acuerdo. Esto indica que el sistema se encuentra en un nivel estándar en cuanto a agilizar el proceso de toma de decisiones gracias a su comprensión clara.
	11	El 80% de los participantes de la encuesta mostró total acuerdo en cuanto a la utilidad de la información proporcionada por el sistema para llevar a cabo sus tareas, mientras que el 20% expresó acuerdo. Esto permite afirmar que el sistema cuenta con la información adecuada para el proceso de comercialización y cotización.

Aprendizaje	15	<p>El 66% de los encuestados expresó total acuerdo, mientras que el 34% mostró acuerdo, lo que indica que los usuarios están altamente satisfechos con las herramientas disponibles en la aplicación móvil y el sitio web.</p>
	5	<p>El 73% de los participantes de la encuesta manifestaron total acuerdo en cuanto a que el sistema web fue fácil de aprender, mientras que el 27% estuvo de acuerdo en este aspecto. Esto indica que los sistemas son altamente intuitivos y fáciles de usar.</p>
	9	<p>El 77% de los encuestados expresaron total acuerdo, mientras que el 22% estuvo de acuerdo, en relación con la claridad de la información del sistema. Estas dos respuestas positivas demuestran que la información proporcionada permite a los usuarios comprender fácilmente lo que deben hacer de manera intuitiva.</p>
	16	<p>El 74% de los participantes de la encuesta expresó estar completamente satisfechos, mientras que el 16% estuvo de acuerdo en general con el sitio web. Estos resultados indican que el sistema cumple con las expectativas de los usuarios y brinda una experiencia satisfactoria al utilizarlo.</p>
	1	<p>El 70% de los participantes de la encuesta expresaron total acuerdo, mientras que el 30% estuvo de acuerdo, en relación con la facilidad de uso del sistema. Estos indicadores altos demuestran que los usuarios están satisfechos al utilizar el sitio web.</p>
	Operabilidad	2
4		<p>El 90% de los participantes de la encuesta expresaron total acuerdo, mientras que el 10% estuvo de acuerdo, en relación a sentirse cómodos al utilizar el sistema. Estas dos respuestas extremadamente positivas indican que el sistema es</p>

		altamente amigable para el usuario, sin causar fatiga o hastío durante su uso.
	6	El 75% de los participantes de la encuesta manifestaron total acuerdo, mientras que el 25% estuvo de acuerdo, en relación con considerarse expertos rápidamente al utilizar el sitio web y la aplicación móvil. Estos valores positivos nos permiten concluir que más de la mitad de los encuestados lograron ser productivos en poco tiempo al utilizar el sistema, mostrando una rápida adaptación y dominio de las herramientas.
	10	El 70% de los participantes de la encuesta manifestaron total acuerdo, mientras que el 30% estuvo de acuerdo, en relación con la facilidad de encontrar la información que necesitan en el sistema y la aplicación móvil. Estos indicadores positivos demuestran que la búsqueda y navegación dentro del sistema son fáciles y efectivas, permitiendo a los usuarios encontrar la información necesaria de manera rápida y sencilla.
	7	El 50% de los participantes de la encuesta expresan total acuerdo, mientras que el 50% están solo de acuerdo, en relación con la indicación de mensajes de error por parte del sistema. Estos dos valores se consideran medianamente positivos, lo que sugiere que se puede mejorar el sistema aumentando las alertas de mensajes de error existentes.
Protección ante errores	8	El 66% de los participantes de la encuesta manifestaron total acuerdo, mientras que el 34% estuvo de acuerdo, en relación con la capacidad de recuperación rápida al cometer un error en el sistema. Estas dos respuestas positivas demuestran que los usuarios pueden resolver los errores de forma ágil y eficiente.
Estética	12	El 69% de los participantes de la encuesta expresaron un alto grado de acuerdo en cuanto a la claridad de la organización del sitio web mientras lo utilizaban, mientras que el 31% estuvo de acuerdo. Estos resultados indican que los usuarios tienen una buena experiencia en términos de la disposición y estructura de la información en el sitio web.

El 68% de los participantes de la encuesta expresaron un acuerdo bastante alto, mientras que el 32% estuvo de acuerdo, en relación con el uso del sitio web. Estos dos factores son positivos, lo que indica que la mayoría de los usuarios tuvieron una experiencia favorable y les gustó el sitio web.

Nota. Fuente. Adaptado de: (Ávila, 2022; De la Cruz & Salazar, 2022).

5.4 Pruebas de experimentación

En esta sección se realizó una prueba de usabilidad con personas voluntarias, estas personas se les dará una serie de indicaciones que se deberán completar para evaluar su experiencia de usuario y mejorar su funcionamiento. Par cumplir esto se divide por estepas que están detalladas en la tabla 68.

Tabla 59

Pruebas experimentales de los prototipos

Definir los objetivos de la prueba		
Descripción	Usuario	Distribuidor
Establece claramente los objetivos que deseas lograr con la prueba de usabilidad. ¿Qué aspectos de la aplicación deseas evaluar? ¿Qué mejoras esperas obtener?	<ul style="list-style-type: none"> • Inicio de sesión • Registro de datos • Realizar un pedido • Ver configuraciones • Seleccionar nueva ruta 	<ul style="list-style-type: none"> • Inicio de sesión • Creación perfil de usuario • Recibir pedido • Registro de datos • Ver historial de pedidos • Ver configuraciones
Seleccionar a los participantes		
Identifica el perfil de los usuarios a quienes deseas que participen en la prueba. Deben ser representativos	<ul style="list-style-type: none"> • Amas de casa • Estudiantes • Personas voluntarias 	<ul style="list-style-type: none"> • Distribuidor de gas

del público al que se destina la aplicación.

Preparar el entorno

Se aseguro de contar con un espacio adecuado para llevar a cabo las pruebas. Debe ser tranquilo y libre de distracciones. Además, se preparó el dispositivo móvil con la versión de la aplicación que se va a evaluar.

- Biblioteca universitaria
- Hogar
- Almacén “Provemadera Ibarra”
- Almacén “Provemadera Ibarra”
- Estación de abastecimiento Agip Gas (Ibarra. Aduana).
- Calles de la ciudad de Ibarra

Explicar el proceso a los participantes

Se informó a los participantes sobre el propósito de la prueba y las tareas que deben realizar. Se aseguró de que comprendan que están siendo evaluados y que no están siendo evaluados ellos como personas.

- Explicar que realizaran tareas.
- Las tareas deben hacer y narrar el proceso.
- Explicar que realizaran tareas.
- Las tareas deben hacer y narrar el proceso.

Observar y registrar

Se observó atentamente a los participantes mientras realizaron las tareas. Se registró sus acciones, comentarios, gestos y cualquier dificultad que encuentren. Se uso grabaciones de pantalla y



Se realizo la toma de datos a los participantes realizando las tareas mencionadas en los distribuidores voluntarios en las tareas

grabaciones de audio para “definir los objetivos de prueba”, donde 4 de cada 5 usuarios no tuvieron problemas para realizar las tareas asignadas.



mencionadas en “definir los objetivos de prueba”, donde 3 de cada 5 usuarios no tuvieron problemas para realizar las tareas asignadas.

Recopilar retroalimentación verbal

Después de cada tarea, se pidió a los participantes que expresen sus pensamientos, impresiones y sugerencias sobre la aplicación. Esto proporcionó información valiosa sobre su experiencia subjetiva.

- Al primer prototipo de pidió realizar más grande los botones.
- Poner un color menos brillante.
- Poner el texto más grande
- Que no se pida código de confirmación por mensajes de texto.
- Poner un color menos brillante.
- Poner el texto más grande
- Que no distraiga al conductor.
- Permitir seleccionar más cilindros de gas.
- Ver la ruta trazada más clara.

Analizar los resultados

Se revisó las grabaciones y notas para identificar patrones de comportamiento y áreas problemáticas. Se clasificó los problemas por su gravedad y prioridad.

Cada tarea fue ejecutada por separado.

Se pedía que haga una cosa y se toman los apuntes.

Las grabaciones generan nuevos indicios para realizar una aplicación más intuitiva por parte de los distribuidores

Generar un informe de usabilidad

Se elaboró un informe que resume los resultados de las pruebas de usabilidad.

Incluye una descripción de los problemas identificados, recomendaciones de mejora y ejemplos de éxito. Este informe será útil para la toma de decisiones y la implementación de cambios en la aplicación.

El informe se detalla la hora, la fecha, que tipo de modulo analizó y cuáles son las inquietudes que presentan al ejecutar la tarea.

El informe se detalla la hora, la fecha, que tipo de modulo analizó y cuáles son las inquietudes que presentan al ejecutar la tarea.

Iterar y mejorar

Con base en los resultados de las pruebas de usabilidad, se realizó mejoras en la aplicación. Luego, se repitió el proceso de pruebas de usabilidad para evaluar si las mejoras han tenido un impacto positivo en la experiencia del usuario

Personas que nos les gusta el color muy claro, se cambió a otro tipo de color menos claro

Con base a los botones se realizó un aumento de tamaño y cambio de colores.

Por parte de la pantalla se cambió la fuente a una puntuación más.

Personas que nos les gusta el color muy claro, se cambió a otro tipo de color menos claro

Con base a los botones se realizó un aumento de tamaño y cambio de colores.

Por el tipo de fuente en pantalla se cambió la fuente a una puntuación más.

Cambio de

Nota. Elaboración propia.

5.1 Discusión de resultados

Los trabajos relacionados que respaldan el desarrollo de aplicaciones móviles para la adquisición de gas doméstico y la prestación de servicios relacionados se describen a continuación y se presenta una discusión basada en los aspectos clave de cada referencia que se encuentra detallada en el 1.6.16 del capítulo uno; como trabajos relacionados.

En el ámbito Nacional e internacional se encontró el desarrollo previo a este proyecto aplicaciones similares que utilizaron el servicio de geolocalización (Ávila, 2022). Pero no se encontró el programa desarrollado en con el entorno Dart y Flutter. Sin embargo (Camacho, 2020) & (Acurio Javier, 2017) realizaron la aplicación similar con enfoque a dispositivos móviles Android, pero en distinto lenguaje de programación.

Durante la definición del proyecto y la revisión de documentos (SLR) en repositorios web, se logró comprender la situación actual, lo que permitió establecer los requerimientos de acuerdo con las normas ISO/IEC-IEEE 29148. Esto confirma que la documentación sigue un proceso estructurado, lo que facilita la definición de requerimientos, tanto funcionales como no funcionales, que posteriormente se utilizaron para crear historias de usuario. Sin embargo, se observó una carencia de procesos estandarizados en investigaciones previas relacionadas con este proyecto, lo que dificulta la orientación del investigador en el enfoque de ingeniería de software.

Eras Camacho desarrolló una aplicación móvil para la adquisición de gas licuado de petróleo (GLP) doméstico en la ciudad de Quito (Frank Camacho, 2020). Utilizó componentes de geolocalización y APIs de Google para ubicar al distribuidor más cercano. Sin embargo, no se encontró similitud de aplicar normas de usabilidad que exigen los estándares internacionales ISO para buenas prácticas de experiencia de usuario. Por otro lado, (Kamoun et al., 2020) presentó una solución móvil para la solicitud de suministro de emergencia de cilindros de oxígeno. El utilizó técnicas como Design Thinking y modelos de casos de uso UML para el análisis y diseño. Por lo tanto, podemos decir que utilizar metodologías, guías normas o estándares; ayudan a diseñar aplicaciones intuitivas que mejoran la experiencia de usuario.

De igual manera en referencia a Álvaro Marcelo Guerrón Paspuel que desarrolló una aplicación móvil utilizando técnicas como el geoposicionamiento y rastreo satelital, se realizaron pruebas de campo y encuestas para validar los resultados. Esto tiene similitud a (Voorheis et al., 2022) realizó una revisión sistemática que explora la integración de Behavioral Design (BD) y Design Thinking (DT) en el desarrollo de aplicaciones móviles de salud. Se identificaron estudios que demuestran la efectividad de combinar estos enfoques en la mejora de la experiencia del usuario y el diseño de interfaces amigables.

Tras finalizar el desarrollo de la aplicación móvil, se aplicaron principios de usabilidad y se observó un aumento en el tamaño de la aplicación en términos de megabytes (MB). Este incremento pasó de 67 MB a 115 MB. Como resultado, podemos concluir que la implementación de una interfaz de usuario (UI) y experiencia de usuario (UX) más elaboradas sí conlleva un consumo adicional de recursos. Sin embargo, es importante destacar que este aumento no tuvo un impacto significativo en el rendimiento de la aplicación. Gracias a su compilación en el lenguaje Dart, la aplicación sigue siendo ágil y eficiente en la ejecución de sus actividades.

Finalmente se realizaron pruebas de funcionamiento y rendimiento para determinar los límites y mejorar la visualización en tiempo real de los distribuidores (Acurio Javier, 2017). Sin embargo, la geolocalización ha evolucionado desde el 2017 hasta el 2013, en precisión (Leandro Sacco, 2023), velocidad (@cienciaplus, 2023), tiempo de respuesta (@cienciaplus, 2023) y nuevas tecnologías (digitalelement.com, 2023); lo que hace que este proyecto utilice herramientas actuales como bases de datos no relacionales, aplicaciones directamente instaladas en los dispositivos, nuevos lenguajes de programación que hacen más fácil el desarrollo y lanzamiento de aplicaciones para distintas plataformas sin tener que codificar para cada plataforma independientemente.

CONCLUSIONES

La elaboración del marco teórico, conjunto la metodología SLR fue fundamental para estructurar el proyecto, seleccionar la arquitectura, metodologías, herramientas y comprender los casos relacionados en el ámbito, local, nacional e internacional que implica al desarrollo de una aplicación móvil. Se identificó los procesos locales de distribución de gas y se obtuvo el contexto claro para definir los requisitos de los usuarios. Además, proporcionó información para aplicar normas y estándares de desarrollo de software, que ayudan a aplicar las mejores prácticas de la industria.

Al definir los requerimientos para la aplicación móvil bajo la norma ISO/IEC/IEEE 29148-2011, facilitó identificar los requisitos funcionales, no funcionales, documentación (Informe de Requisitos (IR), Especificación de Requisitos (ER)) y comprensión del problema para abordar las demás fases del ciclo de vida de software. Además, permitió entender de las necesidades de los Stakeholders estipulado en el capítulo 5.2.2, identificar los procesos estipulados en el capítulo 6, los requerimientos de funcionalidad del capítulo 9.4.4 y requerimientos de usabilidad en el capítulo 9.4.5.

El marco de trabajo Extreme Programming (XP) permitió gestionar el proceso de desarrollo en parejas desde el diseño hasta la programación. mientras uno programa el otro se aseguró de que el desarrollo se ejecute de manera correcta, esto ayuda a realizar la retroalimentación, ya que XP es una metodología flexible para proyectos pequeños que permite ajustar las tareas a conveniencia de los desarrolladores.

La integración de las Apis de Google para la geolocalización, no se necesita de mucho conocimiento para utilizar en pruebas piloto de proyectos académicos, ya que se encontró una gran cantidad de documentación relacionada, y una de las características de desarrollar en Flutter es que se aprovechan al máximo la compatibilidad del lenguaje dart al ser provenientes de una misma empresa se relacionan sin ningún problema.

Al evaluar la API REST utilizando indicadores de calidad basados en la norma ISO 25010 y la herramienta JMeter, se pudo analizar la eficiencia, el desempeño y posibles fallas que se pueda ocasionar, ya que en un entorno real las aplicaciones con Apis de prueba se limitan mucho la velocidad de carga cuando so alojados en servidores en la nube gratuitos. Durante este proceso, se evaluaron aspectos como el comportamiento temporal, la capacidad

y las funciones principales de la API. Los resultados obtenidos permitieron identificar oportunidades de mejora y brindar recomendaciones para optimizar el rendimiento, asegurando así una experiencia satisfactoria y eficiente para los usuarios

Finalmente, en el capítulo 5 se pudo observar cómo es el entorno real y la aceptación del cliente a unos nuevos productos de software. Existe una incertidumbre y desconfianza para utilizar. Se pudo observar en los usuarios que sienten curiosidad, pero no al 100% para utilizar como una aplicación confiable. Gracias al encuesta CSUQ se pudo evaluar la aceptación del cliente en donde 4 de cada 5 personas se mostraron interesadas y no tuvieron problemas para realizar tareas asignadas por los evaluadores.

RECOMENDACIONES

Se recomienda desarrollar una aplicación web especializada en la gestión de la información relacionada con la distribución de gas doméstico. Esta aplicación permitirá centralizar y organizar de manera eficiente los datos relevantes, como perfiles de usuarios, registros de transacciones y detalles de los distribuidores. Al implementar esta solución, se logrará un manejo más efectivo y seguro de la información, lo cual contribuirá a mejorar la eficiencia y la calidad en el proceso de distribución de gas doméstico.

Se recomienda implementar un nuevo módulo de contabilidad en la aplicación móvil que permita la realización de pagos en línea, como transferencias bancarias. Este módulo brindará a los usuarios la opción de realizar pagos de forma segura y conveniente a través de la aplicación, eliminando la necesidad de transacciones en efectivo. Al incorporar esta funcionalidad, se agilizará el proceso de pago y se proporcionará mayor comodidad tanto para los usuarios como para los distribuidores.

Se recomienda implementar certificaciones especializadas en diseño de interfaces y experiencia del usuario (UI/UX) como parte de la formación de profesionales en el campo de aplicaciones móviles. La forma en que los usuarios interactúan con las aplicaciones móviles es crucial para su éxito. Al ofrecer estas certificaciones, se proporcionará a los futuros profesionales los conocimientos y habilidades necesarios para crear interfaces fáciles de usar y brindar experiencias positivas a los usuarios. Esto los preparará para cumplir con las demandas del mercado laboral y contribuir al desarrollo de aplicaciones móviles de alta calidad y centradas en las necesidades de los usuarios.

Para llevar a cabo la evaluación de pruebas de usabilidad, se recomienda contar con un grupo de personas expertas en el tema, así como voluntarios que participen activamente. Durante estas pruebas, se les proporcionarán instrucciones y se observará su ejecución con el objetivo de recopilar información sobre el comportamiento humano y determinar si se requieren realizar cambios en la aplicación.

Otra recomendación adicional sería llevar a cabo evaluaciones con grupos de personas en un entorno de laboratorio. Durante estas evaluaciones, se les proporcionaría una instrucción específica y se observaría su comportamiento mediante cámaras para medir los niveles de estrés experimentados al utilizar la aplicación.

BIBLIOGRAFÍA

- Acens. (2013). Bases de datos NO SQL.
- Acurio Javier. (2017). UNIVERSIDAD POLITÉCNICA SALESIANA SEDE QUITO.
- Alonso, G., & Santacruz, P. (2015). Cálculo e interpretación del Alfa de Cronbach para el caso de validación de la consistencia interna de un cuestionario, con dos posibles escalas tipo Likert. <https://nbn-resolving.org/urn:nbn:de:0168-ssoar-423821>
- Alonso-Arévalo, J., & Mirón-Canelo, A. (2017). Aplicaciones móviles en salud: potencial, normativa de seguridad y regulación Mobile health applications: potential, regulation and security (Vol. 28, Issue 3). <http://scielo.sld.cu>
- Amodeo Enrique. (2013). Principios de diseño de APIs REST. http://leanpub.com/introduccion_apis_rest
- Apache Software Foundation. (2023a, June 6). Apache JMeter™. APACHE. <https://jmeter.apache.org/#top>
- Apache Software Foundation. (2023b, June 6). Apache JMeter™. APACHE. <https://jmeter.apache.org/#top>
- Ashvin Kumbhani. (2023, May 5). Node.js Statistics: The Updated Guide on Node.js Usage and Trends. BACANCY. <https://www.bacancytechnology.com/blog/nodejs-statistics#:~:text=According%20to%20the%20stats%20available,Fermium%2C%20Erbium%2C%20and%20Gallium.>
- Avila Ricardo. (2022). DESARROLLO DE UNA APLICACIÓN WEB PARA LA GEOLOCALIZACIÓN DE MASCOTAS USANDO GPS APLICANDO LA NORMA ISO-IEC-IEEE 29148-2011 PARA LA LICITACIÓN DE REQUISITOS. <http://repositorio.utn.edu.ec/handle/123456789/13189>
- AWS. (2023, May 5). ¿Qué es un marco en programación e ingeniería? Marcos de Desarrollo Móvil. <https://aws.amazon.com/es/what-is/framework/#:~:text=Los%20marcos%20de%20desarrollo%20m%C3%B3vil,utilizarlos%20para%20crear%20aplicaciones%20multiplataforma.>
- Bellavista, P., Küpper, A., & Helal, S. (2008). Location-Based Services: Back to the Future. IEEE Pervasive Computing, 7(02), 85–89. <https://doi.org/10.1109/MPRV.2008.34>

Beltran Gersón. (2016). Geolocalización online: La importancia del dónde - Gersón Beltrán López - Google Libros. <https://books.google.es/books?id=5FLeDQAAQBAJ&printsec=frontcover&d#v=onepage&q&f=false>

Capterra. (2021, January 1). Maze. ¿Qué Es Maze?

Carrizo, D., & Moller, C. (2018). Estructuras metodológicas de revisiones sistemáticas de literatura en Ingeniería de Software: un estudio de mapeo sistemático Methodological structures of systematic literature review in software engineering: a systematic mapping study. <https://www.scielo.cl/pdf/ingeniare/v26s1/0718-3305-ingeniare-26-00045.pdf>

Casanova Leonardo. (2002). Sistema de Posicionamiento Global GPS. http://www.serbi.ula.ve/serbiula/libros-electronicos/Libros/topografia_plana/pdf/CAP-10.pdf

@cienciaplus. (2023, May 19). Geolocalización bajo tierra con partículas donde no llega el GPS.

Consellería de Sanidade, Xunta de Galicia, E., Organización Panamericana de la Salud (OPS-OMS), & Universidad CES, Colombia. (2016, July). EPIDAT. Programa Para Análisis Epidemiológico de Datos. Versión 4.2, Julio 2016. <https://www.sergas.es/Saude-publica/EPIDAT-4-2?idioma=es>

Corral, A., Víctor, L., & Rodríguez, V. (2018). Desarrollo de aplicaciones móviles multiplataforma con Flutter.

Danilo Gavilánez Alvarez, O. I., Layedra, N. I., & Ramos III, V. (2022). Ciencias Técnicas y Aplicadas Artículo de Investigación. 7(7), 2145–2565. <https://doi.org/10.23857/pc.v7i7>

DATAREPORTAL. (2022, March 15). DIGITAL 2022: ECUADOR.

De la Cruz Andrés, & Salazar Fausto. (2022). Desarrollo de una aplicación móvil Android (tienda virtual) para la microempresa “import mag” de la ciudad de Ibarra, que permita fortalecer el proceso de comerciante de ventas. <http://repositorio.utn.edu.ec/handle/123456789/13189>

Developers. (2023a, May 4). Introducción a Android Studio. Android Estudio. <https://developer.android.com/studio/intro?hl=es-419>

Developers. (2023b, May 5). Android Studio Giraffe | 2022.3.1. Android Estudio. <https://developer.android.com/studio/releases?hl=es-419>

Deyde DataCentric. (2018, March 3). How to distinguish between geolocation and georeferencing.

digitalelement.com. (2023). Geolocalización en dispositivos móviles. Digital Element. https://www.digitalelement.com/wp-content/uploads/2020/01/Target-mobile_es.pdf

El Comercio. (2020, January 15). Aplicativo móvil para vender gas de uso doméstico en la frontera del Ecuador. Washington Benalcázar.

Eras Frank. (2020). ESCUELA POLITÉCNICA NACIONAL FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA.

Erika Manzaba. (2020). UNIVERSIDAD AGRARIA DEL ECUADOR FACULTAD DE CIENCIAS AGRARIAS. https://cia.uagraria.edu.ec/Archivos/MANZABA%20ALCIVAR%20ERIKA%20GUADALUPE_compressed.pdf

Esoldo Christine. (2023a, April 1). Designers. The Fundamental Guide to Mobile Usability.

Esoldo Christine. (2023b, April 1). Designers. The Fundamental Guide to Mobile Usability.

Evaluando Software.com. (2021). Qué es la geolocalización y cómo funciona - Evaluando Software. <https://www.evaluandosoftware.com/la-geolocalizacion-funciona/>

Flutter. (2023, April 4). Android Studio e IntelliJ. Installation and Setup. <https://docs.flutter.dev/tools/android-studio>

Fly.io. (2023a, January 1). Fly.io Docs. Fly.io Developer Documentation.

Fly.io. (2023b, June 6). Fly.io developer documentation. Fly.io Docs. <https://fly.io/docs/>

Francisco Aguilera G. (2021, April 21). Cómo Hacer un Test de Usabilidad en Maze. YouTube.

Frank Camacho. (2020). DESARROLLO DE UNA APLICACIÓN MÓVIL EN ANDROID PARA LA ADQUISICIÓN DE GAS LICUADO DE PETRÓLEO (GLP) DE USO DOMÉSTICO.

García Adrian. (2019). Mejoramiento de un sistema de cálculo de orientación en tiempo real basado en GPS no dedicados. https://oa.upm.es/56646/1/TFG_ADRIAN_DEL_AMO_GARCIA.pdf

García María. (2015). Investigación de usos y tipos de aplicaciones móviles.

GitHub Docs. (2023, January 1). Hello World. Get Started. <https://docs.github.com/en/get-started/quickstart/hello-world>

Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra. (2022). Plan de Desarrollo y Ordenamiento Territorial del Cantón Ibarra. <https://www.ibarra.gob.ec/site/planificacion-estrategica-1/pdyot/>

Google for Developers Korea Blog: Windows용 Flutter를 소개합니다. (n.d.). Retrieved May 24, 2023, from <https://developers-kr.googleblog.com/2022/03/announcing-flutter-for-windows.html>

Google Help. (2023a, January 1). Ayuda de Play Console. Crea y Configura Tu App.

Google Help. (2023b, July 1). Tamaño de los objetivos táctiles. Ayuda de Accesibilidad de Android. <https://support.google.com/accessibility/android/answer/7101858?hl=es>

Gordon Kelley. (2021, June 6). Grupo Nielsen Norman. Using Color to Enhance Your Design.

Guerrón Alvaro. (2018). ALTERNATIVA TECNOLÓGICA CON GEO POSICIONAMIENTO Y RASTREO SATELITAL PARA REDUCIR LA CONTAMINACIÓN ACÚSTICA PRODUCIDA POR PARTE DE LOS DISTRIBUIDORES DE GAS DOMÉSTICO DE LA CIUDAD DE IBARRA.

Holzinger, A., Errath, M., Searle, G., Thurnher, B., & Slany, W. (2005). From Extreme Programming and Usability Engineering to Extreme Usability in Software Engineering Education (XP+UEXU).

Hooper, S. (2017). Design for Fingers, Touch, and People, Part 1. <https://www.uxmatters.com/mt/archives/2017/03/design>

Hostinger. (2023, January 10). Qué es GitHub y cómo empezar a usarlo. Hostinger Tutoriales. <https://www.hostinger.es/tutoriales/que-es-github>

IBM. (2021, March 5). Estándares de servicio Web. SOAP.

INEC. (2022). Tecnologías de la información y comunicación. https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas_Sociales/TIC/2022/202207_Tecnología%20de%20la%20Información%20y%20Comunicación%20-%20TICs.pdf

ISO/IEC/IEEE 29148. (2011). ISO/IEC/IEEE 29148:2011(E), Systems and software engineering — Life cycle processes — Requirements engineering. www.iso.org

Isolde, M., Aguilar, H., De La Garza González, A., Patricia, M., Miranda, S., Abigail, A., & Villegas, G. (2015). Adaptación al español del Cuestionario de Usabilidad de Sistemas Informáticos CSUQ Spanish language adaptation of the Computer Systems Usability Questionnaire CSUQ. 4.

Jakob Nielsen. (2020, November 15). Logotipo del grupo Nielsen NormanGrupo Nielsen Norman. 10 Heurísticas de Usabilidad Para El Diseño de Interfaces de Usuario. <https://www.nngroup.com/articles/ten-usability-heuristics/>

John S. Rigden. (2000). Science & Technology. <https://www.hup.harvard.edu/catalog.php?isbn=9780674004351>

Jonathan Briskman. (2022, December 1). ASO Guide: 15 Expert Strategies to Master App Store Optimization. Data.Ai. <https://www.data.ai/en/insights/user-acquisition/app-store-optimization-guide/>

Jose Luis Yujra Tito. (2017). UNIVERSIDAD MAYOR DE SAN ANDRÉS FACULTAD DE CIENCIAS PURAS Y NATURALES CARRERA DE INFORMÁTICA LICENCIA DE USO. <https://repositorio.umsa.bo/bitstream/handle/123456789/16758/T-3363.pdf?sequence=1&isAllowed=y>

Julen Castellano, & David Casamichana. (2014). Deporte con dispositivos de posicionamiento global (GPS), Aplicaciones y limitaciones. *Revista de Psicología de Deporte*, 23, 355–364. <https://www.redalyc.org/articulo.oa?id=235131674015>

Kamoun, F., Barachi, M. El, Hachani, A., Belqasmi, F., Said, A. Ben, & Amri, I. (2020). A geolocation-aware mobile crowdsourcing solution for the emergency supply of oxygen cylinders. *Procedia Computer Science*, 170, 153–160. <https://doi.org/10.1016/j.procs.2020.03.020>

Kos, T., Grgic, M., & Sisul, G. (2006). Mobile user positioning in GSM/UMTS cellular networks. *Proceedings Elmar - International Symposium Electronics in Marine*, 185–188. <https://doi.org/10.1109/ELMAR.2006.329545>

Kumar, A., Kumar, S., Lal, P., Saikia, P., Srivastava, P. K., & Petropoulos, G. P. (2021). Introduction to GPS/GNSS technology. *GPS and GNSS Technology in Geosciences*, 3–20. <https://doi.org/10.1016/B978-0-12-818617-6.00001-9>

La Hora. (2019, June 20). Verifican distribución y venta de gas en Ibarra. <https://www.lahora.com.ec/noticias/verifican-distribucion-y-venta-de-gas-en-ibarra/>

Leandro Sacco. (2023, May 1). Sistemas de geolocalización en dispositivos móviles. *Geolocalización, Dispositivos Móviles*. <https://peritosinformaticos.ar/dispositivos-moviles/geolocalizacion/sistemas-de-geolocalizacion/>

Lewis, J. R. (2018). Measuring Perceived Usability: The CSUQ, SUS, and UMUX. *International Journal of Human-Computer Interaction*, 34(12), 1148–1156. <https://doi.org/10.1080/10447318.2017.1418805>

Lionel Sujay. (2023, March 27). Number of developers using the GitHub platform worldwide from 2013 to 2023. Statista. <https://www.statista.com/statistics/1363004/developer-github-worldwide/>

Lionel Sujay Vailshery. (2023, August 29). Programming/development tools used by software developers worldwide from 2018 to 2022. Statista. <https://www.statista.com/statistics/869106/worldwide-software-developer-survey-tools-in-use/>

MacHado, C., & Campos, J. C. (2021). Towards the integration of user interface prototyping and model-based development. *ICGI 2021 - 2021 International Conference on Graphics and Interaction, Proceedings*. <https://doi.org/10.1109/ICGI54032.2021.9655284>

Madleňák, A. (2021). Geolocation Services and Marketing Communication from a Global Point of View. *SHS Web of Conferences*, 92, 02040. <https://doi.org/10.1051/shsconf/20219202040>

Mahrin, M. N. ri, Strooper, P., & Carrington, D. (2009). Selecting usability evaluation methods for software process descriptions. *Proceedings - Asia-Pacific Software Engineering Conference, APSEC*, 523–529. <https://doi.org/10.1109/APSEC.2009.33>

Marlen Treviño-Villalobos, Leonardo Vízque-Acuña, Rocio Quiroz-Oviedo, & Gaudy Esquiél-Vega. (2019). Vista de Una comparación de rendimiento entre bases de datos NoSQL.

Martin Heller. (2022, July 8). What is Visual Studio Code? Microsoft's extensible code editor. InfoWorld.

Matt Rae. (2020, October 26). ¿Qué es Adobe XD y para qué se emplea? Adobe. <https://www.adobe.com/co/products/xd/learn/get-started/what-is-adobe-xd-used-for.html>

Maza Maria, & Bohorquez Yesica. (2012). DISPOSITIVOS MÓVILES: EVOLUCIÓN Y USO.

Morales Gustavo. (2020). Introducción a Node.js.

Munoz, D., Bouchereau, F., Vargas, C., & Enriquez, R. (2009). The Position Location Problem. Position Location Techniques and Applications, 1–22. <https://doi.org/10.1016/B978-0-12-374353-4.00007-7>

OPS-OMS, & CES. (2016). Programa para análisis epidemiológico de datos (Epidat). In Rev Panam Salud Publica (Vol. 27, Issue 1). <http://dxsp.sergas.es>.

Pinos kevin. (2017). Diseño de un juego para el rescate cultural identitario en adolescentes ecuatorianos.

Postman. (2023, May 5). 2023 State of the API Report. Tooling for APIs and Development. <https://www.postman.com/state-of-api/tooling-for-apis-and-development/#api-platforms-and-tools>

Quiroz Carlolina. (2022). IMPLEMENTACIÓN DE UN MENÚ DIGITAL PARA RESTAURANTES DE ESPECIALIDAD BASADO EN EL ESTÁNDAR ISO/IEC/IEEE 29148.

Rebecca Johnston-Gilbert. (2023, February 13). There's an API for that. Postman. <https://blog.postman.com/theres-an-api-for-that/>

Salazar Steven. (2017). ANÁLISIS COMPARATIVO ENTRE LOS SISTEMAS OPERATIVOS DE DISPOSITIVOS MÓVILES CON MAYOR DEMANDA EN EL MERCADO.

Schlosser, S., Toninelli, D., & Cameletti, M. (2021). Comparing methods to collect and geolocate tweets in Great Britain. Journal of Open Innovation: Technology, Market, and Complexity, 7(1), 1–20. <https://doi.org/10.3390/joitmc7010044>

Stack Overflow. (2023, May 5). 2023 Developer Survey. Web Frameworks and Technologies. <https://survey.stackoverflow.co/2023/#most-popular-technologies-webframe>

StatCounter. (2023, October 3). Operating System Market Share Worldwide. Operating System Market Share Worldwide - September 2023. <https://gs.statcounter.com/os-market-share>

Statista, & Lionel Sujay Vailshery. (2023, June 1). Cross-platform mobile frameworks used by developers worldwide 2019-2022. Cross-Platform Mobile Frameworks Used by Software Developers Worldwide from 2019 to 2022. <https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/>

StudySmarter. (2023, August 4). Operating Systems. <https://www.studysmarter.co.uk/explanations/computer-science/computer-systems/operating-systems/>

SWEBOK. (2014). Guide to the Software Engineering Body of Knowledge SWEBOK ® A Project of the IEEE Computer Society. <https://cs.fit.edu/~kgallagher/Schtick/Serious/SWEBOKv3.pdf>

Ubicalo. (2021). 6 TIPOS DE EQUIPOS DE RASTREO SATELITAL. <https://www.ubicalo.com.mx/blog/equipos-de-rastreo-satelital/>

Ulloa Diana. (2014). ESTUDIO DE METODOLOGÍAS PARA ESTANDARIZAR EL DESARROLLO DE SOFTWARE EN EL DEPARTAMENTO DE INFORMÁTICA EN LA PASTORAL SOCIAL CARITAS DE LA DIÓCESIS DE AMBATO. https://repositorio.uta.edu.ec/bitstream/123456789/8524/1/Tesis_t937si.pdf

UNR. (2019). Georreferenciacion. https://www.fceia.unr.edu.ar/gps/cursos/Georreferenciacion_2019.pdf

Valencia Luis. (2021). Introducción a SQL.

Vázquez Víctor. (2018). Desarrollo de aplicaciones móviles multiplataforma con Flutter.

Velazco Raúl. (2017). Aplicación Android para el entrenamiento cognitivo de personas bajo tutela jurídica con discapacidad intelectual o del desarrollo.

Villota Oyarvide, W. R., & Parrales Herrera, S. C. (2021). Usabilidad de aplicaciones móviles para pedidos a domicilio: COVID-19 y emergencia sanitaria en Guayaquil. Question/Cuestión, 3(69). <https://doi.org/10.24215/16696581e551>

Visual Studio Code. (2023, August 8). Visual Studio Code FAQ. <https://code.visualstudio.com/docs/supporting/FAQ>

Voorheis, P., Zhao, A., Kuluski, K., Pham, Q., Scott, T., Sztur, P., Khanna, N., Ibrahim, M., & Petch, J. (2022). Integrating Behavioral Science and Design Thinking to Develop Mobile Health Interventions: Systematic Scoping Review. In *JMIR mHealth and uHealth* (Vol. 10, Issue 3). JMIR Publications Inc. <https://doi.org/10.2196/35799>

Yanina Muradas. (2019, June 3). Qué es Postman y primeros pasos. OpenWebinars. <https://openwebinars.net/blog/que-es-postman/>

Yasvi, M. A., & Yadav, K. S. (2019). Review On Extreme Programming-XP Hybrid Text Summarization View project Ontology Design Recommendation View project Review On Extreme Programming-XP. <https://www.researchgate.net/publication/332465869>

Zambrano, J., David, E., Freire, G., Xavier, E., Mena, P., & Maribel, V. (2018). Herramientas de Google Maps en aplicaciones móviles con georreferenciación.

ANEXOS

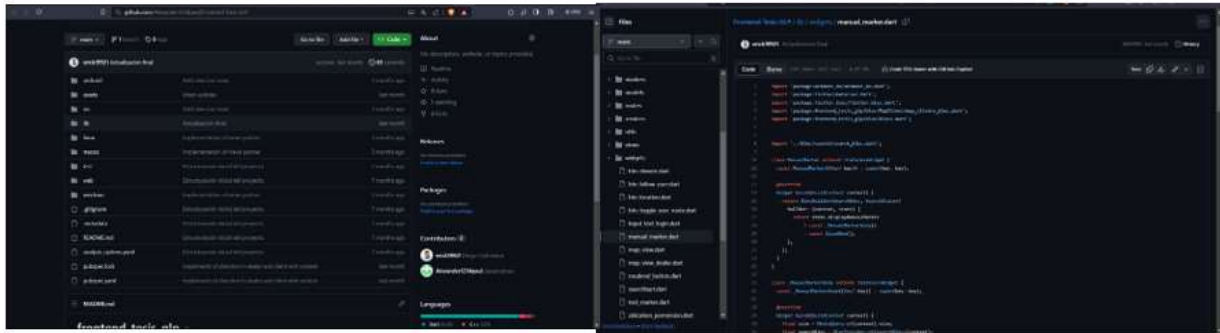
Anexo A: Código fuente de la Aplicación móvil

Diríjase al siguiente enlace: <https://github.com/Alexander1234paul/Frontend-Tesis-GLP.git>

Anexo B:

Figura 80

Proyecto del FrontEnd subido en el repositorio de GitHub



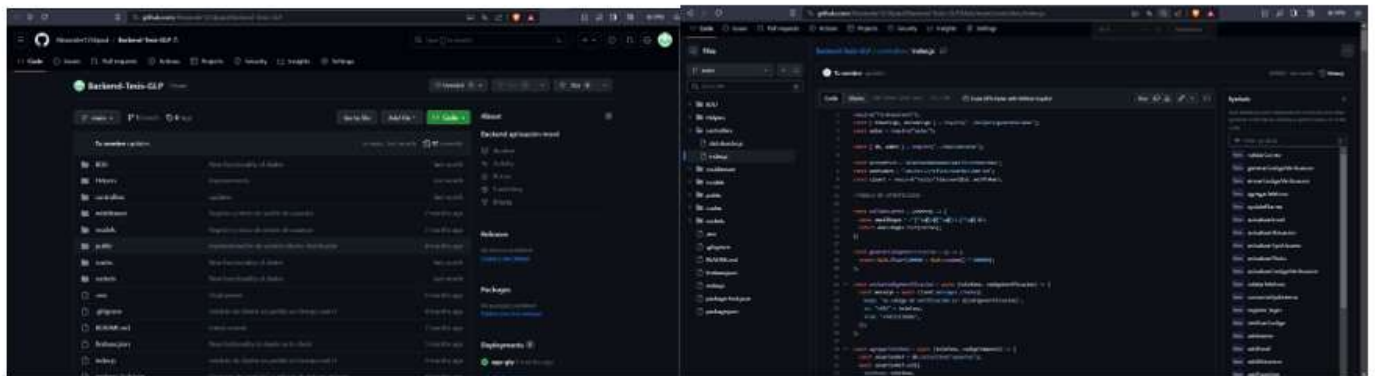
Anexo C: Código fuente de la Api Rest

Diríjase al siguiente enlace: <https://github.com/Alexander1234paul/Backend-Tesis-GLP.git>

Anexo D:

Figura 8160

Proyecto del Backend subido en el repositorio de GitHub



Anexo E:

Figura 82

Entrevista a distribuidores en la empresa ENI



Anexo F:

Figura 83

Uso de la Aplicación Móvil por los Distribuidores



Anexo G:

Figura 84

Uso de la Aplicación Móvil por los Clientes

