



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE TELECOMUNICACIONES

INFORME FINAL DEL TRABAJO DE INTEGRACIÓN
CURRICULAR

TEMA:

“SISTEMA DE MONITOREO DE CONTENEDORES DE RESIDUOS
INFECCIOSOS CON NOTIFICACIÓN DE RECOLECCIÓN
INDICANDO LA MEJOR RUTA PARA LABORATORIOS CLÍNICOS
DE LA CIUDAD DE IBARRA”

**Trabajo de titulación previo a la obtención del título de Ingeniera en
Telecomunicaciones**

Línea de investigación: Innovación Tecnológica y de Productos

AUTOR:

Melannie Adriana Calero Vallejos

DIRECTOR:

Ing. Jaime Roberto Michilena Calderón, Msc.

IDENTIFICACIÓN DE LA OBRA

La Universidad Técnica del Norte dentro del proyecto Repositorio Digital Institucional, determinó la necesidad de disponer de textos completos en formato digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la Universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	171849533-4		
APELLIDOS Y NOMBRES:	Calero Vallejos Melannie Adriana		
DIRECCIÓN:	Av. Camilo Pompeyo Guzmán y Magnolias		
EMAIL:	macalero@utn.edu.ec		
TELÉFONO FIJO:	xxxxxxxxxx	TELF. MOVIL	0985571632

DATOS DE LA OBRA	
TÍTULO:	Sistema de monitoreo de contenedores de residuos infecciosos con notificación de recolección indicando la mejor ruta para laboratorios clínicos de la ciudad de Ibarra.
AUTOR (ES):	Calero Vallejos Melannie Adriana
FECHA: AAAAMMDD	2025-02-10
SOLO PARA TRABAJOS DE INTEGRACIÓN CURRICULAR	
CARRERA/PROGRAMA:	<input checked="" type="checkbox"/> GRADO <input type="checkbox"/> POSGRADO
TÍTULO POR EL QUE OPTA:	Ingeniera en Telecomunicaciones
DIRECTOR:	Ing. Jaime Roberto Michilena Calderón, Msc.

AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Melannie Adriana Calero Vallejos, con cédula de identidad Nro. 171849533-4, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de integración curricular descrito anteriormente, hago entrega del ejemplar respectivo en formato digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión; en concordancia con la Ley de Educación Superior Artículo 144.

Ibarra, a los 10 días del mes de febrero de 2025

LA AUTORA:



.....

Melannie Adriana Calero Vallejos

CONSTANCIAS

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

Ibarra, a los 10 días, del mes de febrero de 2025

LA AUTORA:

A handwritten signature in blue ink, appearing to read "Melannie Calero", written over a horizontal dotted line.

Melannie Adriana Calero Vallejos

CERTIFICACIÓN DEL DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Ibarra, 10 de febrero de 2025

ING. JAIME ROBERTO MICHILENA CALDERÓN, MSC.

DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

CERTIFICA:

Haber revisado el presente informe final del trabajo de Integración Curricular, el mismo que se ajusta a las normas vigentes de la Universidad Técnica del Norte; en consecuencia, autorizo su presentación para los fines legales pertinentes.



.....
Ing. Jaime Roberto Michilena Calderón, Msc.

C.C.: 100219843-8

APROBACIÓN DEL COMITÉ CALIFICADOR

El Comité Calificador del trabajo de Integración Curricular “Sistema de monitoreo de contenedores de residuos infecciosos con notificación de recolección indicando la mejor ruta para laboratorios clínicos de la ciudad de Ibarra” elaborado por Melannie Adriana Calero Vallejos, previo a la obtención del título de Ingeniera en Telecomunicaciones, aprueba el presente informe de investigación en nombre de la Universidad Técnica del Norte:



.....
Ing. Jaime Roberto Michilena Calderón, Msc.
C.C.: 100219843-8



.....
Ing. Luis Edilberto Suárez Zambrano, Msc.
C.C.: 100230429-1

DEDICATORIA

Dedico el presente trabajo de titulación con profunda gratitud a las personas que, con su inquebrantable apoyo, amor y dedicación, han sido pilares fundamentales en mi vida y en el desarrollo de este trabajo.

A mi padre Xavier Calero, por estar siempre a mi lado, brindándome su fuerza y motivación, me enseñó a no rendirme frente a los desafíos y a encontrar la fortaleza en los momentos difíciles. A mi madre Sandra Vallejos, que, con su amor infinito y sus sabios consejos, fue mi mayor guía en los momentos de incertidumbre, recordándome siempre que era capaz de superar cualquier obstáculo. A ambos, por su amor, sacrificio y el ejemplo de vida que me han dado, les dedico este esfuerzo, pues sin ustedes este sueño no habría sido posible, y todo lo que soy se los debo a ustedes.

A mi hermana, Samantha Calero, por ser una parte esencial de mi vida y ocupar un lugar irremplazable en mi corazón, aunque no siempre compartimos los mismos puntos de vista, valoro profundamente su apoyo y su compañía. El cariño y la conexión que compartimos me recuerdan cuánto la quiero y lo importante que es para mí.

A mi novio, Paúl Ramírez, por ser mi compañero incondicional, siempre ayudándome, apoyándome y motivándome en cada paso de este camino. Le agradezco profundamente por su amor constante y por estar siempre a mi lado, brindándome la fuerza necesaria para alcanzar esta meta. Su apoyo ha sido mi mayor inspiración, y por ello le dedico este logro con todo mi corazón.

A mi abuelita Luz Amaya, por todo lo que compartimos durante tantos años, por su amor y su presencia en mi vida. Cuánto me hubiera gustado que estuviera aquí para compartir este logro, porque su ausencia me deja un vacío inmenso, pero también una gran cantidad de recuerdos que siempre llevaré conmigo.

A mi perrito, Alberto, por acompañarme en las largas noches de estudio y por ser mi mayor motivación. Sus juegos siempre lograron desestresarme y llenar mis días de alegría.

Melannie Adriana Calero Vallejos

AGRADECIMIENTO

Agradezco a Dios por darme la fortaleza, las oportunidades y la guía necesarias para completar esta etapa, acompañándome en cada paso de este camino.

Agradezco a la Universidad Técnica del Norte por ser el pilar de mi formación profesional, por brindarme una educación de calidad y un entorno donde crecí como persona, rodeada de valores, retos y oportunidades que marcaron mi vida.

Al Ing. Jaime Michilena, en calidad de tutor, le agradezco profundamente por su apoyo incondicional durante todo este proceso, por estar siempre presente brindándome guía, motivación y confianza, y por inspirarme a superar los retos con dedicación y perseverancia.

Al Ing. Luis Suárez, en calidad de asesor, le expreso mi gratitud por su valiosa orientación, sus observaciones precisas y por aportar de manera significativa al desarrollo y calidad de este trabajo, contribuyendo con su conocimiento a perfeccionar cada detalle.

Al equipo del Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura, especialmente al Tecnólogo Edwin Vásquez Mera, a la Doctora Fressia Rosas, a la señora Consuelo Sandoval y al señor Patricio Quiñaluiza, les expreso mi gratitud por su invaluable apoyo, por facilitarme los recursos necesarios y por su colaboración en la realización de las pruebas esenciales para este trabajo.

Finalmente, a mis amigos, Magali, Edison, Andrés y Marlon quienes estuvieron a mi lado durante toda mi carrera universitaria, compartiendo risas, desafíos y momentos inolvidables. Su compañía y palabras de aliento en los momentos más desafiantes me dieron la fuerza y motivación necesarias para seguir adelante y alcanzar este logro.

Melannie Adriana Calero Vallejos

RESUMEN EJECUTIVO

El presente trabajo de titulación aborda la ineficiencia en la recolección de desechos infecciosos en laboratorios clínicos de Ibarra, Ecuador, un problema que incrementa los riesgos de propagación de enfermedades y contaminación ambiental. La principal causa es la falta de monitoreo adecuado de los niveles de llenado de los contenedores y una planificación deficiente de las rutas de recolección. El objetivo principal de este proyecto es diseñar un sistema de monitoreo inteligente para los contenedores de desechos infecciosos, utilizando sensores inalámbricos y una aplicación móvil que notifique la necesidad de recolección y proporcione la ruta más eficiente para su retiro, mejorando así la seguridad y rapidez en la gestión de residuos. Para lograrlo, se adoptó un enfoque metodológico basado en el modelo en cascada, que abarca las fases de análisis, diseño, implementación, verificación y servicio. En la fase de análisis, se identificaron deficiencias en el proceso actual, como la falta de datos en tiempo real sobre el nivel de llenado y las ineficiencias en las rutas de recolección. El sistema propuesto incorpora una aplicación móvil que permite consultar niveles de llenado, recibir notificaciones en tiempo real y, mediante el algoritmo de Dijkstra, calcular y mostrar las rutas más cortas para optimizar la recolección al reducir desplazamientos innecesarios. Las pruebas realizadas confirmaron que el sistema mide y registra con precisión los niveles de llenado, proporcionando información en tiempo real y optimizando las operaciones. Este proyecto demuestra cómo la tecnología puede transformar sistemas críticos contribuyendo a la salud pública y la sostenibilidad ambiental.

Palabras clave: gestión de desechos infecciosos, monitoreo, sensores inalámbricos, algoritmo de Dijkstra, optimización de rutas, salud pública.

ABSTRACT

This research focuses on the inefficiencies in infectious waste collection from clinical laboratories in Ibarra, Ecuador, a critical issue that heightens the risks of disease transmission and environmental contamination. The root causes of this problem are inadequate monitoring of container fill levels and poor planning of collection routes. The primary objective of this study is to design an intelligent monitoring system for infectious waste containers. This system employs wireless sensors and a mobile application to notify collection needs and provide the most efficient routes for waste removal, improving both the safety and efficiency of waste management processes. To achieve this, a waterfall methodology was adopted, comprising the phases of analysis, design, implementation, verification, and service. During the analysis phase, deficiencies in the current process were identified, including the absence of real-time data on container fill levels and inefficiencies in route optimization. The proposed system integrates a mobile application that enables users to monitor fill levels, receive real-time notifications, and utilize Dijkstra's algorithm to calculate and display the shortest routes. This functionality minimizes unnecessary travel and optimizes collection processes. Testing demonstrated that the system accurately measures and records container fill levels, providing real-time data that significantly enhances operational planning. This project exemplifies how advanced technology can address critical inefficiencies in waste management systems, making substantial contributions to public health and environmental sustainability.

Keywords: Infectious Waste Management, Monitoring, Wireless Sensors, Dijkstra Algorithm, Route Optimization, Public Health

LISTA DE SIGLAS

MSP. Ministerio de Salud Pública.

GADM-I. Gobierno Autónomo Descentralizado Municipal de Ibarra.

OMS. Organización Mundial de la Salud.

ACCESS: Agencia de Aseguramiento de la Calidad de los Servicios de Salud y Medicina Propagada.

INEC. Instituto Nacional de Estadísticas y Censos.

SENPLADES. Secretaría Nacional de Planificación y Desarrollo.

APP. Aplicación Móvil.

API. Interfaz de Programación de Aplicaciones.

APK. Paquete de Aplicación Android (Android Application Package)

SDK. Kit de Desarrollo de Software.

IoT. Internet de las Cosas.

WPAN. Wireless Personal Area Network (Red de Área Personal Inalámbrica).

WLAN. Wireless Local Area Network (Red Local Inalámbrica).

WMAN. Wireless Metropolitan Area Network (Red Metropolitana Inalámbrica).

WWAN. Wireless Wide Area Network (Red de Área Amplia Inalámbrica).

INDICE DE CONTENIDOS

1. Capítulo I: Antecedentes	28
1.1. Tema	28
1.2. Planteamiento del Problema	28
1.3. Objetivos.....	29
1.3.1. Objetivo General.....	29
1.3.2. Objetivos Específicos.....	30
1.4. Alcance	30
1.5. Justificación.....	33
2. Capítulo II: Fundamentos Teóricos.....	35
2.1. Internet de las Cosas	35
2.1.1. Arquitectura IoT.....	35
2.2. Redes de Sensores Inalámbricos.....	37
2.2.1. Elementos que Componen una Red de Sensores Inalámbricos.....	37
2.2.1.1. <i>Red de Nodos de Sensores</i>	38
2.2.1.2. <i>Gateway</i>	38
2.2.1.3. <i>Estación Base</i>	38
2.2.2. Topología de las Redes de Sensores Inalámbricos.....	39
2.2.2.1. <i>Topología Tipo Estrella</i>	39
2.2.2.2. <i>Topología Tipo Árbol</i>	40
2.2.2.3. <i>Topología Tipo Malla</i>	40

2.2.3.	Aplicaciones de las Redes de Sensores Inalámbricos	40
2.3.	Tecnologías de Comunicación Inalámbrica	42
2.3.1.	Clasificación de Redes Inalámbrica	42
2.3.1.1.	<i>Redes Inalámbricas de Área Personal (WPAN)</i>	42
2.3.1.2.	<i>Redes Inalámbricas de Área Local (WLAN)</i>	42
2.3.1.3.	<i>Redes Inalámbricas de Área Metropolitana (WMAN)</i>	43
2.3.1.4.	<i>Redes Inalámbricas de Área Amplia (WWAN)</i>	43
2.3.2.	Ventajas de las Tecnologías de Comunicación Inalámbrica.....	43
2.4.	Tecnología Wi-Fi.....	44
2.4.1.	Estándares que certifica la Alianza Wi-Fi.....	45
2.4.2.	Características de la tecnología Wi-Fi	46
2.4.3.	Ventajas del uso de la Tecnología Wi-Fi	46
2.4.4.	Aplicaciones de la Tecnología Wi-Fi.....	46
2.5.	Desechos de Residuos Infecciosos	47
2.5.1.	Clasificación de los Desechos de Residuos Infecciosos	48
2.5.2.	Fases de la Gestión Integral de Desechos Infecciosos	49
2.5.3.	Métodos de Recolección	49
2.5.3.1.	<i>Método de Parada Fija</i>	50
2.5.3.2.	<i>Método de Acera</i>	50
2.5.3.3.	<i>Método de Contenedores</i>	51
2.6.	Algoritmo Dijkstra.....	51
2.6.1.	Características	52

	14
2.6.2. Grafos	52
2.7. Metodología en Cascada.....	53
3. Capítulo III: Diseño del Sistema	55
3.1. Descripción del Área de Estudio	55
3.1.1. La Ciudad de Ibarra.....	56
3.1.2. Agencia de Aseguramiento de la Calidad de los Servicios de Salud y Medicina Prepagada (ACCESS-Imbabura)	58
3.1.3. Protocolo de Manejo de Recolección de Desechos infecciosos.....	61
3.1.4. Métodos de Recolección de Desechos Infecciosos de la Ciudad de Ibarra ..	62
3.1.4.1. <i>Servicio Público de Recolección de Desechos Infecciosos</i>	62
3.1.4.2. <i>Servicio Privado de Recolección de Desechos Infecciosos</i>	70
3.1.5. Métodos Interno de Recolección de Desechos Infecciosos realizado en los Laboratorios Clínicos.....	74
3.2. Requisitos del Sistema.....	78
3.2.1. Stakeholders	78
3.2.2. Requerimientos de Stakeholders	79
3.2.3. Requerimientos del Sistema	81
3.2.4. Requerimientos de la Arquitectura.....	82
3.3. Componentes del Sistema.....	84
3.3.1. <i>Etapa de Sensado</i>	84
3.3.2. <i>Etapa de Monitoreo</i>	91
3.3.3. <i>Etapa de Almacenamiento de Datos</i>	93

	15
3.3.4. <i>Etapa de Visualización</i>	96
3.4. Diseño del Sistema	99
3.4.1. Diagrama de Bloques	100
3.4.2. Selección de la Tecnología de Comunicación.....	101
3.4.3. Arquitectura del Sistema	104
3.4.4. Diagramas de Interconexión.....	105
3.4.4.1. <i>Nodo Sensor</i>	105
3.5. Desarrollo del Software	107
3.5.1. Diagrama de Flujo del Sistema	107
3.5.2. Configuración de la Base de Datos	108
3.5.3. Configuración del Nodo Sensor	112
3.5.3.1. <i>Programación ESP32</i>	113
3.5.3.2. <i>Diagrama de flujo del Nodo Sensor</i>	121
3.5.4. Configuración de la Aplicación Móvil con Flutter	123
3.5.4.1. <i>Autenticación de Usuarios</i>	123
3.5.4.2. <i>Niveles de Llenado de los Contenedores de Desechos Infecciosos</i>	124
3.5.4.3. <i>Informes de Laboratorios</i>	126
3.5.4.4. <i>Sistema de Notificaciones</i>	128
3.5.4.5. <i>Cálculo de Rutas mediante el Algoritmo de Dijkstra</i>	131
3.5.4.6. <i>Diagrama de Flujo de la Aplicación Móvil</i>	137
3.6. Diseño y Montaje de la Carcasa para el Nodo Sensor.....	139
3.6.1. Distribución de Componentes en el Nodo Sensor.....	139

3.6.2.	Ubicación del Nodo Sensor en el Contenedor	140
3.7.	Funcionamiento de la Aplicación Móvil	141
3.7.1.	Página Principal de Aplicación Móvil	142
3.7.2.	Usuarios Recolectores de Desechos Infecciosos.....	142
3.7.3.	Usuarios de Laboratorios Clínicos	145
3.7.4.	Usuarios de Administración de Aplicación Móvil	146
3.7.4.1.	<i>Registro de Usuarios de Laboratorio, Recolectores, Administradores...</i>	147
3.7.4.2.	<i>Informes de Laboratorios</i>	149
3.7.4.3.	<i>Informes de Recolectores.....</i>	152
3.7.4.4.	<i>Envío de Notificaciones</i>	153
4.	Capítulo IV: RESULTADOS Y ANÁLISIS	155
4.1.	Evaluación del Funcionamiento General del Sistema	156
4.1.1.	Verificación de Funcionamiento de Inicio de Sesión de Usuarios.....	156
4.1.2.	Verificación de Registro, Edición y Eliminación de Usuarios.....	158
4.1.3.	Verificación de la capacidad del nodo sensor para adaptarse a contenedores de diferentes tamaños.....	160
4.2.	Evaluación de Logros del Sistema.....	161
4.2.1.	Evaluación del Sistema de Monitoreo de Contenedores de Residuos Infecciosos	161
4.2.2.	Validación de Mapas para Cálculo de la Ruta más corta entre Origen y Destino	162
4.2.3.	Verificación del Envío de Notificaciones Personalizadas.....	165

4.2.3.1. Envío de Notificaciones desde el usuario Recolector hacia el usuario de Laboratorio Clínico	166
4.2.3.2. Envío de Notificaciones desde el usuario de Laboratorio Clínico hacia el usuario Recolector	168
4.2.3.3. Envío de Notificaciones desde el usuario Administrador hacia todos los usuarios	169
4.2.4. Evaluación de Informes de los Contenedores	170
4.2.4.1. Encuesta realizada por el personal del Laboratorio Clínico	170
4.2.4.2. Análisis comparativo con los Informes obtenidos en la APP	172
4.3. Discusión de resultados	174
Conclusiones y recomendaciones	177
Conclusiones.....	177
Recomendaciones	178
Referencias Bibliográficas.....	180
ANEXO 1 – Encuesta realizada al personal del Laboratorio Clínico	189
ANEXO 2 – Resultados obtenidos de la encuesta	195
ANEXO 3 – Certificado de Validación de Pruebas del Funcionamiento de la Sistema Emitido por el Sindicato Provincial de Choferes Profesionales de Imbabura	199
ANEXO 4 – Certificado de Validación del Sistema de Monitoreo tras Verificación Personal de la Doctora Fressia Rosas del Sindicato Provincial de Choferes Profesionales de Imbabura	200
ANEXO 5 – Código del Nodo Sensor	201

INDICE DE TABLAS

Tabla 1	45
<i>Cuadro comparativo de los estándares IEEE 802.11 más relevante</i>	45
Tabla 2	48
<i>Clasificación de los desechos de residuos infecciosos</i>	48
Tabla 3	60
<i>Listado de los Laboratorios Clínicos Ubicados en el centro de la Ciudad de Ibarra</i>	60
Tabla 4	68
<i>Listado de los Laboratorios Clínicos de la Ciudad de Ibarra, que se encuentran registrados dentro del sistema de recolección de desechos infecciosos brindado por el GADM Ibarra</i>	68
Tabla 5	72
<i>Listado de los Laboratorios Clínicos de la Ciudad de Ibarra, que hacen uso de gestores externos privados que brindan servicio de recolección de desechos infeccioso</i>	72
Tabla 6	76
<i>Etiquetas de los Desechos Sanitarios Peligrosos</i>	76
Tabla 7	78
<i>Lista de Requerimientos con sus respectivas abreviaturas</i>	78
Tabla 8	79
<i>Lista de Requerimientos con sus respectivas abreviaturas</i>	79
Tabla 9	80
<i>Requerimientos de Stakeholders</i>	80
Tabla 10	81
<i>Requerimientos del Sistema</i>	81

Tabla 11	83
<i>Requerimientos de la arquitectura</i>	83
Tabla 12	86
<i>Tabla comparativa de Sensores Medidores de Distancia</i>	86
Tabla 13	89
<i>Tabla comparativa de Módulos Microcontroladores</i>	89
Tabla 14	94
<i>Tabla comparativa de Bases de Datos</i>	94
Tabla 15	97
<i>Tabla comparativa de Softwares para el desarrollo de Aplicaciones Móviles</i>	97
Tabla 16	102
<i>Tabla comparativa de las Tecnologías de Comunicación Inalámbrica</i>	102
Tabla 17	133
<i>Aplicación del Algoritmo de Dijkstra</i>	133
Tabla 18	155
<i>Evaluación del Funcionamiento General del Sistema</i>	155
Tabla 19	155
<i>Evaluación de logros del sistema</i>	155
Tabla 20	196
<i>Días de mayor afluencia de pacientes (mayor generación de desechos infecciosos)</i> ...	196
Tabla 21	196
<i>Temporadas de mayor afluencia de pacientes</i>	196
Tabla 22	198
<i>Sugerencias de las personas entrevistadas</i>	198

INDICE DE FIGURAS

Figura 1	36
<i>Capas de la Arquitectura IoT</i>	36
Figura 2	37
<i>Elementos de una red de sensores inalámbricos</i>	37
Figura 3	39
<i>Tipos de Topologías de las Redes de Sensores</i>	39
Figura 4	50
<i>Métodos de recolección de desechos infecciosos</i>	50
Figura 5	53
<i>Partes del algoritmo de Dijkstra</i>	53
Figura 6	54
<i>Metodología en cascada</i>	54
Figura 7	57
<i>Ubicación geográfica de la Ciudad de Ibarra junto con sus límites y sus Parroquias Urbanas</i>	57
Figura 8	58
<i>Total de permisos de funcionamiento otorgado a establecimientos de salud del cantón Ibarra, de la provincia de Imbabura</i>	58
Figura 9	59
<i>Permisos de funcionamiento otorgado a establecimientos de salud del cantón Ibarra, de la provincia de Imbabura, divididos en niveles de atención</i>	59
Figura 10	64
<i>Pesaje previo a la recolección de los residuos infecciosos</i>	64
Figura 11	66

<i>Información del estado actual de la recolección de desechos infecciosos de la ciudad de Ibarra “Reunión con la Ingeniera Karla Lima Paredes, Técnico de Gestión Ambiental de la Dirección de Gestión Ambiental/Residuos Sólidos”</i>	66
Figura 12	70
<i>Ubicación de los Laboratorios Clínicos Registrados el Sistema de Recolección del Municipio de la ciudad de Ibarra</i>	70
Figura 13	74
<i>Ubicación de los Laboratorios Clínicos de la ciudad de Ibarra que utilizan el método de recolección de desechos infecciosos proporcionado por gestores privados</i>	74
Figura 14	75
<i>Clasificación de desechos infecciosos generados en los laboratorios clínicos</i>	75
Figura 15	77
<i>Áreas de almacenamiento temporal de los laboratorios clínicos</i>	77
Figura 16	88
<i>Funcionamiento del Sensor Ultrasónico HC-SR04</i>	88
Figura 17	91
<i>Funcionamiento del Módulo Microcontrolador ESP32</i>	91
Figura 18	92
<i>Funcionamiento del Sistema de Monitoreo</i>	92
Figura 19	96
<i>Funcionamiento de Firebase Realtime Database</i>	96
Figura 20	101
<i>Diagrama de Bloques del Sistema</i>	101
Figura 21	104
<i>Tecnología de Comunicación Wifi</i>	104

Figura 22	105
<i>Arquitectura del Sistema</i>	105
Figura 23	106
<i>Diagrama de interconexión del nodo sensor</i>	106
Figura 24	108
<i>Diagrama de Flujo del Sistema</i>	108
Figura 25	109
<i>Creación de nuevo Proyecto</i>	109
Figura 26	110
<i>Creación de Base de Datos</i>	110
Figura 27	111
<i>Configuración del SDK</i>	111
Figura 28	112
<i>Bibliotecas de conexión con Firebase</i>	112
Figura 29	112
<i>Conexión con Firebase (Base de Datos)</i>	112
Figura 30	113
<i>Importación de Librerías Wifi y credenciales de conexión</i>	113
Figura 31	114
<i>Conexión con Base de Datos</i>	114
Figura 32	115
<i>Configuración de Sensor Ultrasónico HC-SR04</i>	115
Figura 33	116
<i>Conexión con la red Wi-Fi</i>	116
Figura 34	117

<i>Conexión con la base de datos Firebase y servidor NTP.....</i>	<i>117</i>
Figura 35.....	118
<i>Cálculo del tamaño del contenedor.....</i>	<i>118</i>
Figura 36.....	119
<i>Medición de nivel de llenado con Sensor Ultrasónico HC-SR04.....</i>	<i>119</i>
Figura 37.....	120
<i>Cálculo del porcentaje de llenado con Sensor Ultrasónico HC-SR04.....</i>	<i>120</i>
Figura 38.....	120
<i>Datos guardados temporalmente.....</i>	<i>120</i>
Figura 39.....	121
<i>Envío de datos a Firebase</i>	<i>121</i>
Figura 40.....	123
<i>Diagrama de Flujo del Nodo Sensor.....</i>	<i>123</i>
Figura 41.....	124
<i>Función de Inicio de Sesión.....</i>	<i>124</i>
Figura 42.....	125
<i>Obtención de datos de los Contenedores (Llamado a base de datos Firebase).....</i>	<i>125</i>
Figura 43.....	126
<i>Obtención del último dato guardado (Porcentajes y distancias).....</i>	<i>126</i>
Figura 44.....	127
<i>Conexión con Base de datos.....</i>	<i>127</i>
Figura 29.....	127
<i>Presentación de todos los datos guardados en Base de datos</i>	<i>127</i>
Figura 30.....	128
<i>Verificación en dos pasos.....</i>	<i>128</i>

Figura 31	129
<i>Contraseñas de Aplicaciones</i>	129
Figura 32	129
<i>Conexión de Correo electrónico con Código para Aplicación Móvil</i>	129
Figura 33	130
<i>Envío de correo electrónico (Notificación)</i>	130
Figura 34	131
<i>Mensajes de Notificaciones</i>	131
Figura 35	132
<i>Grafo con nodos dirigidos</i>	132
Figura 36	134
<i>Nodos y sus ubicaciones</i>	134
Figura 37	135
<i>Definición de distancias entre nodos (aristas)</i>	135
Figura 38	135
<i>Inicialización de las distancias de todos los nodos</i>	135
Figura 39	136
<i>Selección de nodo con menor distancia conocida</i>	136
Figura 40	137
<i>Reconstrucción de la ruta más corta</i>	137
Figura 41	138
<i>Diagrama de Flujo de la Aplicación Móvil</i>	138
Figura 42	140
<i>Distribución de componentes del nodo sensor</i>	140
Figura 43	141

<i>Plataforma metálica que sujetará al nodo sensor</i>	141
Figura 44	141
<i>Colocación del nodo sensor</i>	141
Figura 45	142
<i>Pantalla principal de la aplicación móvil</i>	142
Figura 46	143
<i>Página de inicio de sesión de usuarios recolectores e ingreso a página principal</i>	143
Figura 47	144
<i>Página de laboratorio clínico</i>	144
Figura 48	145
<i>Página de inicio de sesión de usuarios de laboratorios clínicos</i>	145
Figura 49	146
<i>Pantalla principal de usuarios de laboratorio clínico</i>	146
Figura 50	147
<i>Inicio de sesión como usuario administrador</i>	147
Figura 51	148
<i>Registro de usuarios de laboratorio clínico</i>	148
Figura 52	149
<i>Registro de usuarios y funciones de edición y eliminación de información registrada</i>	149
Figura 53	150
<i>Verificación de informes de los contenedores de los laboratorios clínicos</i>	150
Figura 54	151
<i>Informes de los contenedores de los laboratorios clínicos</i>	151
Figura 55	152

<i>Análisis de los contenedores de desechos infecciosos ubicados en los laboratorios clínicos</i>	152
Figura 56	153
<i>Listado de Recolecciones Realizadas</i>	153
Figura 57	154
<i>Envío de notificaciones desde usuario administrador</i>	154
Figura 58	157
<i>Inicio de sesión correcto</i>	157
Figura 59	158
<i>Ingreso de sesión incorrecto</i>	158
Figura 60	159
<i>Registro de nuevos usuarios</i>	159
Figura 61	159
<i>Edición y eliminación de usuarios</i>	159
Figura 62	161
<i>Medición de tamaño del contenedor</i>	161
Figura 63	162
<i>Sistema de monitoreo de contenedores de desechos infecciosos</i>	162
Figura 64	163
<i>Verificación de la ruta más corta</i>	163
Figura 65	165
<i>Rutas de recolección 1 – Mapa de rutas utilizado por el GADM-I (Filtrado por laboratorios clínicos)</i>	165
Figura 66	166

<i>Envío de notificaciones desde el usuario Recolector hacia el usuario de Laboratorio Clínico</i>	166
Figura 67	167
<i>Envío de notificaciones Recolector - Laboratorio Clínico</i>	167
Figura 68	168
<i>Envío de notificaciones desde el usuario de Laboratorio Clínico hacia el usuario Recolector</i>	168
Figura 69	169
<i>Envío de notificaciones desde el usuario Administrador hacia todos los usuarios registrados</i>	169
Figura 70	171
<i>Resultados de las preguntas realizadas sobre los días de mayor generación de desechos infecciosos</i>	171
Figura 71	171
<i>Resultados de las preguntas realizadas (nivel de llenado de los contenedores)</i>	171
Figura 72	173
<i>Informes de los datos tomados en el Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura – Contenedor 1</i>	173
Figura 73	174
<i>Informes de los datos tomados en el Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura – Contenedor 2</i>	174
Figura 74	193
<i>Encuesta realizada a la Fisioterapeuta Johana Chacón Bustamante</i>	193
Figura 75	194
<i>Encuesta realizada a la Dra. Fressia Rosas Médico General</i>	194

Capítulo I: Antecedentes

1.1. Tema

Sistema de monitoreo de contenedores de residuos infecciosos con notificación de recolección indicando la mejor ruta para laboratorios clínicos de la ciudad de Ibarra.

1.2. Planteamiento del Problema

En el pasado, en Ecuador, la forma de recolectar y desechar los desechos infecciosos de los laboratorios clínicos no siempre cumplía con los estándares adecuados. Esto representaba riesgos para la salud pública y el medio ambiente, ya que existía la posibilidad de propagar enfermedades y contaminar el suelo y el agua. Sin embargo, las autoridades sanitarias y los laboratorios han tomado conciencia de esta problemática y han colaborado para establecer regulaciones y procedimientos que aseguren una gestión segura y eficiente de los desechos infecciosos (Pupiales, 2023).

En la actualidad, el Ministerio de Salud Pública es la institución responsable de supervisar y regular la gestión de los desechos infecciosos. Esta entidad ha establecido pautas y regulaciones precisas que los laboratorios clínicos deben considerar para llevar a cabo de manera adecuada la recolección, almacenamiento, transporte y eliminación final de dichos desechos. De modo que se establece que los laboratorios clínicos tienen la responsabilidad de clasificar de manera adecuada los desechos infecciosos y emplear contenedores seguros y debidamente etiquetados. Estos desechos deben ser recolectados de forma periódica por compañías autorizadas encargadas del transporte y tratamiento de residuos peligrosos. Estas empresas deben cumplir con los requisitos establecidos por la legislación de Ecuador y asegurar la correcta eliminación de los desechos infecciosos (MSP, 2017).

En la ciudad de Ibarra los laboratorios clínicos se rigen al cumplimiento de la normativa propuesta por el Ministerio de Salud Pública en lo que conlleva a la recolección

y almacenamiento de desechos tóxicos de manera interna, sin embargo, a pesar de que estos parámetros se cumplan, el problema se presenta cuando la recolección de los contenedores por parte del personal encargado de forma externa es ineficiente, considerando que debido a que no tienen conocimiento sobre el nivel de llenado de los contenedores estos pueden demorar bastante tiempo en acudir al retiro de estos o en otros casos pasar cuando los contenedores aún no han alcanzado su capacidad máxima, de igual manera otro de los factores presentes es el tráfico generado en el centro de la ciudad de Ibarra para lo cual los encargados de la recolección toman diferentes rutas para llegar al destino, que en varios casos no son las más adecuadas (Granda, 2022).

A medida que se prolonga el tiempo de recolección de los desechos infecciosos, aumenta el riesgo de infecciones, accidentes o lesiones. Un ejemplo de ello son las agujas y jeringas que, debido a su naturaleza puntiaguda, pueden representar un peligro si no se manipulan correctamente. Asimismo, existe la posibilidad de derrames de líquidos u otros materiales infecciosos si los residuos se acumulan y no se manejan adecuadamente. Es por ello que para el desarrollo de este proyecto se propone implementar un sistema de monitoreo de los residuos infecciosos de los laboratorios clínicos de Ibarra, en donde se realizarán pruebas de almacenamiento máximo en sus contenedores para proceder a enviar una notificación hacia el personal encargado de la recolección y a su vez informar acerca de la ruta más rápida de llegada hacia el laboratorio clínico que lo requiera.

1.3. Objetivos

1.3.1. Objetivo General

Diseñar un sistema de monitoreo de contenedores de residuos infecciosos, mediante el uso de una red de sensores inalámbricos que presenten información hacia una aplicación móvil con notificación de recolección para indicar la mejor ruta hacia los laboratorios clínicos de Ibarra.

1.3.2. Objetivos Específicos

- Analizar mediante el compendio de referencias bibliográficas, el proceso de recolección de desechos infecciosos generados en los laboratorios clínicos, para comprender el método utilizado por el Municipio de San Miguel de Ibarra.
- Verificar el mapa de rutas que realizan los carros de recolección de desechos infecciosos brindado por el Municipio de San Miguel de Ibarra, tomando en cuenta el algoritmo de Dijkstra, para poder evaluar la mejor ruta.
- Diseñar un sistema de monitoreo de contenedores de residuos infecciosos, utilizando sensores inalámbricos y una aplicación móvil para el envío de información y notificación donde se especifique la ruta más corta de llegada.
- Realizar pruebas del funcionamiento del sistema implementado, para poder analizar y predecir el tiempo que tarda en llenarse un contenedor con desechos infecciosos.

1.4. Alcance

El presente proyecto tiene como objetivo el desarrollo de un sistema de monitoreo de contenedores de residuos infecciosos ubicados en los laboratorios clínicos de la ciudad de Ibarra, para poder evaluar el nivel de llenado de estos, con el fin de proporcionar información mediante una aplicación móvil que indique la ruta más corta por medio de una notificación, cuando el contenedor ya haya alcanzado su capacidad máxima. Con el sistema se pretende lograr que no ocurran derrames en el área de almacenamiento de contenedores para evitar la propagación de enfermedades debido a la recolección de desechos tardada.

En el proyecto se hará uso de la metodología en cascada que consiste en un procedimiento secuencial y lineal, dividiéndose en distintas fases que permiten alcanzar

los objetivos y actividades definidos para el desarrollo del proyecto. Las fases que conforman esta metodología son las siguientes: análisis, diseño, implementación, verificación y servicio (Agency, 2018).

En la etapa de análisis, se realizará un estudio de la información bibliográfica acerca de los métodos de recolección de desechos infecciosos de los laboratorios clínicos de la ciudad de Ibarra; de modo que se pueda comprender el procedimiento utilizado por los encargados de recolectar este tipo de desechos. En primer lugar, resulta primordial reconocer los laboratorios clínicos donde se producen los desechos infecciosos, considerándose cada uno de ellos como un punto destinado a la recolección. Mediante la ayuda de un sistema de información geográfica, es necesario señalar en el mapa todas las ubicaciones de los puntos de recolección para facilitar la visualización y planificación de las rutas, con lo cual se hará un estudio del algoritmo de Dijkstra, el mismo que se aplicará para el cálculo de la ruta más corta tomando en cuenta como nodo origen a los encargados de recolectar los desechos infecciosos y como nodos destino a cada uno de los tres laboratorios clínicos considerados, y posteriormente proceder a guardar las rutas dentro de una base de datos.

En la etapa de diseño, se llevará a cabo el desarrollo del sistema de monitoreo del nivel de contenedores junto con la aplicación móvil, en el cual se utilizará un sensor ultrasónico que mida la distancia entre la tapa del contenedor y los residuos almacenados para poder calcular el nivel de llenado del mismo e inmediatamente enviar dichos datos hacia la aplicación para su visualización, una vez se haya alcanzado la capacidad máxima de llenado se procederá a enviar una notificación en la cual se especificarán los datos generales del laboratorio clínico que requiera su recolección y la ruta más corta de llegada, la misma que se encontrará previamente establecida y guardada en la base de datos; de igual manera se va a considerar la hora del día en el cual exista mayor tráfico en el centro

de la ciudad de Ibarra para que mediante la notificación de recolección se muestre la segunda ruta más corta como una ruta alterna a la primera, evaluando primeramente si la ruta más corta es afectada por este problema planteado. Cabe recalcar que todas las rutas serán calculadas una sola vez y se mostrarán mediante la notificación de recolección, solo cuando los contenedores ya hayan alcanzado su capacidad máxima.

En la etapa de implementación, se instalará y configurará la red de sensores inalámbricos tomando en cuenta los laboratorios seleccionados para la recolección de los desechos producidos, con el objetivo de generar datos del nivel mediante los sensores que serán conectados a una placa microcontroladora para el procesamiento de estos. Especialmente, se requiere obtener dichos datos para informar a través de la aplicación móvil cuando los contenedores ya se encuentren llenos y requieran ser recolectados enviando una notificación junto con la especificación de la ruta más corta previamente calculada. También se realizará el análisis de la frecuencia con la que se generan los desechos infecciosos en cada sitio, de modo que se pueda predecir un tiempo estimado en el que los contenedores de los laboratorios clínicos se encontrarán llenos para brindar dicha información en la APP.

En la etapa de verificación y servicio, se realizarán las respectivas pruebas de funcionamiento de la red de sensores utilizando diferentes niveles de desechos con los que los sensores evaluarán cuando los contenedores alcancen su capacidad máxima y de igual manera se evaluará el funcionamiento de la aplicación móvil verificando la presentación de los datos del nivel de los residuos infecciosos en los contenedores y el envío de la notificación hacia los encargados de la recolección de los mismos. También se verificará que la ruta calculada con el algoritmo de Dijkstra sea la correcta, comparando con el mapa de rutas obtenido en la etapa de análisis.

1.5. Justificación

Las sustancias contaminadas de origen médico, como las agujas usadas, los apósitos contaminados y otros objetos sanitarios, pueden albergar microorganismos peligrosos, como bacterias, virus y hongos, capaces de ocasionar enfermedades graves. Si estos residuos no se recogen de forma oportuna, se incrementa el peligro de propagación de enfermedades entre el personal de salud, los pacientes y la comunidad en su (Ecuador, 2019).

Según la Organización Mundial de la Salud (OMS), las agencias de salud y los organismos reguladores han establecido normas y regulaciones específicas que deben seguirse al gestionar desechos infecciosos. Estas regulaciones suelen incluir requisitos para la recolección y el adecuado manejo de dichos desechos. Es crucial cumplir con estas normas para garantizar la seguridad y la salud de todas las personas involucradas en el manejo de los desechos, y también para evitar posibles sanciones legales o repercusiones negativas para las instituciones y organizaciones responsables (OMS, 2022).

Una de las bases esenciales del Plan de Creación de Oportunidades del Ecuador consiste en asegurar la salud y el bienestar de la población. Es crucial realizar una recolección eficiente de desechos infecciosos con el fin de prevenir la propagación de enfermedades infecciosas y salvaguardar la salud pública. Al garantizar una recogida oportuna y adecuada de dichos desechos, se reduce al mínimo el riesgo de exposición a agentes patógenos y se protege la salud de las personas (Plan de Creación de Oportunidades, 2021).

Es por ello que considerando todos los factores influyentes con respecto al riesgo que se tiene cuando no se toma en cuenta la recolección de desechos infecciosos de manera rápida y adecuada, se realizará este proyecto en el cual se desea implementar un sistema de monitoreo de contenedores de residuos infecciosos con notificación de

recolección, indicando la mejor ruta, en los laboratorios clínicos de Ibarra, para lo cual se hará uso de plataformas en la nube y sensores que permitan realizar un cerrado automático de los contenedores en cuanto estos hayan alcanzado su capacidad de llenado máxima y seguidamente dar aviso a los recolectores de desechos indicándoles la ruta más rápida de llegada hacia el lugar, de modo que se pueda evitar la propagación de enfermedades tomando en cuenta que no se podrá realizar un llenado forzado que ocasionen derrames y a su vez su recolección de forma rápida y organizada.

2. Capítulo II: Fundamentos Teóricos

Para el desarrollo de este proyecto, es fundamental comprender los conceptos teóricos de la temática que se van a manejar, es por ello que, en el marco teórico se abordarán temas como arquitectura IoT para entender cómo interactúan diferentes dispositivos, sensores y aplicaciones. También se estudiarán las redes de sensores inalámbricos profundizando en elementos como nodos, gateways, topologías y aplicaciones. Se analizarán tecnologías inalámbricas para transmitir datos de los sensores como WiFi, incluyendo estándares 802.11. Asimismo, se investigará conceptos importantes sobre residuos infecciosos, su recolección y los diferentes métodos de rutas como parada fija, acera y contenedores. Para optimizar estas rutas de recolección, se evaluará la implementación del algoritmo de Dijkstra por su capacidad de encontrar rutas de menor costo en grafos ponderados. Finalmente, se estudiará la metodología en cascada para entender las etapas estándar de desarrollo de un sistema software, partiendo de requerimientos hasta pruebas y mantenimiento.

2.1. Internet de las Cosas

El Internet de las cosas (IoT), consiste en conectar dispositivos equipados con tecnologías integradas a una red informática para hacerlos inteligentes, de modo que, estas conexiones le permitan recopilar e intercambiar datos de forma remota, administrar dispositivos y establecer reglas de acción para los dispositivos, eliminando la necesidad de participación humana en algunas acciones y operaciones (Saltos, 2023).

2.1.1. Arquitectura IoT

La arquitectura IoT es un sistema de elementos que consta de cuatro capas, como se observa en la figura 1, en donde se especifica el proceso mediante el cual fluyen los datos desde los sensores conectados a las “cosas” a través de la red hasta llegar a un centro

de datos donde se permite la recopilación, procesamiento y análisis de estos (García, Molina, & Avarado, 2020).

Figura 1

Capas de la Arquitectura IoT



Fuente: (Chuquimarca, 2022)

- **Capa de Sensado:** Su principal función es recopilar datos ambientales utilizando sensores como temperatura, humedad, gas, IR y movimiento, y reconocer otros dispositivos del sistema IoT. Esta capa adquiere y puede controlar actuadores, preparando los datos para su transmisión. (Chuquimarca, 2022).
- **Capa de Conectividad:** Crea una conexión entre dispositivos y una plataforma o gateway mediante diversas tecnologías inalámbricas como Wi-Fi, NB-IoT, Zigbee, Bluetooth y LPWAN. Su objetivo principal es enviar y procesar datos recolectados por sensores del entorno. Además de enlazar dispositivos inteligentes, redes y servidores, esta capa puede ofrecer características como la computación en la nube. (Domínguez, 2016).
- **Capa de Análisis y Procesado:** Es esencial para procesar y analizar los datos recolectados por los dispositivos de la capa de sensado. Su función incluye la integración de información y el filtrado de datos no deseados para la capa de aplicación (Chuquimarca, 2022).

- **Capa de Aplicación:** La capa de aplicación es crucial para desarrollar y ejecutar aplicaciones que logren los objetivos del sistema y aprovechen los datos procesados. Su valor reside en ofrecer una variedad de servicios a los usuarios finales, utilizando protocolos como MQTT, CoAP, HTTP y XMPP según las necesidades del sistema IoT (Domínguez, 2016).

2.2. Redes de Sensores Inalámbricos

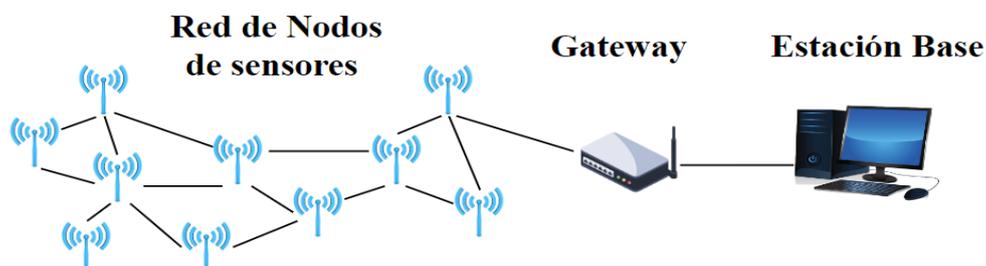
Las redes inalámbricas de sensores (WSN) son sistemas distribuidos de dispositivos autónomos que utilizan sensores para monitorear diversas condiciones ambientales (Hurtado, 2019). Estos dispositivos, llamados "motes", están equipados con microcontroladores, radios transeptores y sensores, operando, recolectando y comunicando datos localmente hacia un nodo central de coordinación (Cajas K. , 2022).

2.2.1. Elementos que Componen una Red de Sensores Inalámbricos

Una red de sensores está conformada por nodos de sensores distribuidos que tienen como objetivo la recopilación de datos de su entorno, que a su vez son enviadas hacia una estación base por medio de un nodo Gateway, para realizar el procesamiento de estos, como se observa en la figura 2 (Guamán, 2016).

Figura 2

Elementos de una red de sensores inalámbricos



Fuente: (Cajas K. , 2022)

2.2.1.1.Red de Nodos de Sensores

Los nodos sensores son dispositivos autónomos que convierten señales físicas en digitales y las transmiten de forma inalámbrica a un nodo central para su procesamiento. Cada nodo incluye unidad sensora, procesamiento, comunicación inalámbrica y alimentación (Guamán, 2016). Trabajan cooperativamente para recolectar y transmitir datos, ofreciendo flexibilidad, fácil instalación y capacidades de auto-restauración y diagnóstico para mantener la red operativa (Salazar K. , 2018).

2.2.1.2.Gateway

El nodo gateway actúa como intermediario entre los sensores y el servidor central, coordinando la comunicación y transferencia de datos recolectados para ser analizados o almacenados. Su función principal es permitir que los datos de los nodos sensores lleguen a un sistema centralizado (Martínez, 2009). Además, puede realizar tareas como filtrado, agregación de datos y enrutamiento para mejorar la eficiencia. Si bien presenta desafíos como consistencia de extremo a extremo y seguridad, el gateway es fundamental para integrar y operar eficazmente una red de sensores, garantizando comunicación fluida entre nodos y sistemas externos (Altamirano, 2018).

2.2.1.3.Estación Base

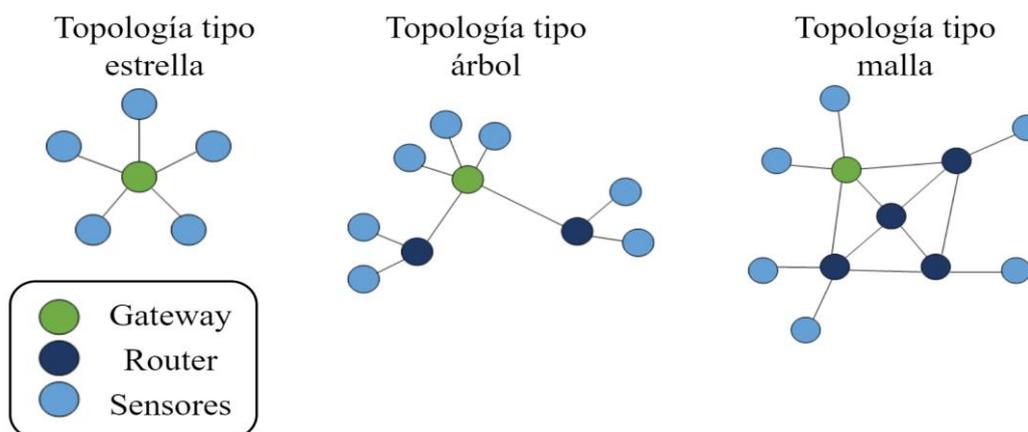
La estación base funciona como un recolector de datos, ya sea basado en un ordenador común o en un sistema embebido, que almacena y procesa los datos obtenidos por los nodos sensores. Por otro lado, también se encarga de establecer la comunicación con redes externas, como Internet, proporcionando una interfaz gráfica para que el usuario final pueda interactuar con la red de sensores inalámbricos y controlarla de forma remota (Martínez, 2009).

2.2.2. Topología de las Redes de Sensores Inalámbricos

Hay varias arquitecturas que se pueden utilizar para diseñar una red WSN, donde se debe considerar la configuración de hardware y cómo los datos se transmiten, ya que cada topología es adecuada para trabajar bajo ciertos parámetros. Entre las topologías más utilizadas se encuentran la topología tipo estrella, topología tipo árbol y topología tipo malla, presentes en la figura 3.

Figura 3

Tipos de Topologías de las Redes de Sensores



Fuente: (Tomalo, 2018)

Elaborado por: La Autora

2.2.2.1. Topología Tipo Estrella

La topología de tipo estrella consiste en nodos sensores conectados directamente al nodo central o gateway a una distancia de 30 a 100 metros, lo que permite una comunicación de baja latencia con un solo salto (Martínez, 2009). Esta configuración requiere que el nodo central esté dentro del rango de transmisión de todos los sensores. Es adecuada para redes grandes, ya que facilita la subdivisión de cargas de trabajo, pero puede haber colisiones si dos sensores utilizan el medio físico simultáneamente (Tomalo, 2018).

2.2.2.2. Topología Tipo Árbol

La topología de árbol se origina a partir de la topología en estrella, adoptando una disposición jerárquica donde algunos dispositivos se conectan directamente al nodo central, mientras que otros se conectan a dispositivos secundarios, formando una estructura en cascada (Cajas K. , 2022). Cada nodo envía datos a un nodo superior, culminando en el nodo central. Esta disposición facilita la expansión de la red y mejora la detección de errores, aunque aumenta la dependencia de los enlaces, lo que puede provocar el colapso de toda la red si alguno de ellos falla. Aunque ofrece una mayor cobertura, también implica una mayor latencia debido a su complejidad (Tomalo, 2018).

2.2.2.3. Topología Tipo Malla

La topología de malla establece conexión entre nodos de manera que se creen múltiples rutas alternativas, lo que facilita la transmisión de datos incluso si hay problemas en la red. Cada nodo puede enviar y recibir datos, actuando como routers, lo que permite una comunicación flexible y eficiente (Martínez, 2009). La estructura de la topología es similar a un árbol, con un coordinador en la parte superior y nodos intermedios y finales. Esta disposición proporciona redundancia y capacidad de recuperación ante fallos, aunque implica una inversión significativa debido a su extensión. El descubrimiento de ruta garantiza la elección de la mejor ruta disponible para cada mensaje, optimizando así la comunicación en la red (Romero, 2017).

2.2.3. Aplicaciones de las Redes de Sensores Inalámbricos

Las redes inalámbricas de sensores tienen la capacidad de ser desplegadas en una variedad de áreas gracias a características como el bajo costo de implementación y la larga duración de la batería (Tomalo, 2018). A continuación, se menciona la aplicación de WSN en las áreas más destacadas:

- **Monitorización Ambiental:** Consiste en recopilar datos del entorno como calidad del aire y temperatura. Se usa en áreas urbanas, agrícolas y nucleares para tomar decisiones informadas sobre gestión ambiental. Ofrece despliegue rápido, pero enfrenta desafíos como gestión de energía y seguridad de datos (Altamirano, 2018).
- **Cuidado de la Salud:** Utiliza dispositivos conectados para monitorear constantemente la salud de los pacientes, enviando datos biométricos a los profesionales de la salud para análisis y seguimiento, lo que permite una atención más oportuna y personalizada (Martínez, 2009).
- **Apoyo Militar:** Permiten la recopilación y transmisión de datos en tiempo real para mejorar la vigilancia, el reconocimiento y la toma de decisiones en el campo de batalla. Esto proporciona a las fuerzas militares información crítica para operaciones tácticas y estratégicas, aumentando la eficiencia y la seguridad en el terreno (Martínez, 2009).
- **Agricultura y Ganadería:** Se facilita el monitoreo remoto de condiciones ambientales y del ganado, optimizando el uso de recursos como agua y fertilizantes, previniendo enfermedades y mejorando la productividad. Esto permite a los agricultores y ganaderos tomar decisiones más informadas y eficientes para maximizar rendimientos y minimizar costos (Guamán, 2016).
- **Seguridad y Vigilancia:** Detección temprana de intrusiones, monitoreo de áreas sensibles y respuesta rápida a situaciones de riesgo. Estas redes proporcionan una cobertura amplia y continua, facilitando la protección de infraestructuras críticas y la prevención de incidentes delictivos o emergencias (Saltos, 2023).

2.3. Tecnologías de Comunicación Inalámbrica

Las tecnologías de comunicación inalámbrica son sistemas que posibilitan la transferencia de datos entre dispositivos sin la necesidad de cables físicos, utilizando señales electromagnéticas como ondas de radio o infrarrojas (Salazar J. , 2019). Ejemplos comunes incluyen Wi-Fi, Bluetooth, NFC, 4G LTE, 5G, Zigbee y RFID.

2.3.1. Clasificación de Redes Inalámbrica

Las tecnologías de comunicación inalámbrica pueden clasificarse en diferentes categorías según su alcance y propósito

2.3.1.1. Redes Inalámbricas de Área Personal (WPAN)

Las redes inalámbricas de área personal, según el estándar IEEE 802.15, son sistemas de comunicación que operan en distancias cortas, típicamente alrededor de 10 metros. A diferencia de otras redes inalámbricas, las WPANs no requieren mucha infraestructura y están diseñadas para ser eficientes en energía y de bajo costo (Salazar J. , 2019). Estas redes se implementan en una variedad de dispositivos, como teléfonos inteligentes y PDAs. Son conocidas por su bajo consumo de energía y velocidad de transmisión moderada. Las tecnologías comunes incluyen Bluetooth para dispositivos como ratones y teclados, IrDA para transferencia de datos punto a punto, ZigBee para control de procesos y UWB para enlaces de banda ancha (Talón, 2016).

2.3.1.2. Redes Inalámbricas de Área Local (WLAN)

Las redes inalámbricas de área local (WLAN) se diseñan para ofrecer acceso sin cables en áreas con un alcance típico de hasta 100 metros, comúnmente utilizadas en entornos como hogares, escuelas u oficinas. Estas redes permiten a los usuarios moverse dentro de un área cubierta y mantenerse conectados. Basadas en el estándar IEEE 802.11 y comercializadas como Wi-Fi, las WLAN han superado a otros estándares como HIPERLAN debido a su simplicidad y rápida adopción en el mercado (Manzo, 2021).

2.3.1.3. Redes Inalámbricas de Área Metropolitana (WMAN)

Las redes inalámbricas de área metropolitana (WMAN) constituyen un grupo importante en las tecnologías de comunicación sin cables. Estas redes se basan en el estándar IEEE 802.16, conocido como WiMAX (Worldwide Interoperability for Microwave Access), diseñado para proporcionar altas velocidades de transmisión de datos en áreas metropolitanas (Salazar J. , 2019). WiMAX utiliza una arquitectura punto a multipunto, lo que permite la conexión de redes LAN más pequeñas para formar una red más amplia sin necesidad de cableado costoso. A diferencia de Wi-Fi, que se enfoca en áreas pequeñas como oficinas o puntos de acceso, WiMAX ofrece cobertura a distancias mayores, operando en bandas de frecuencia de 2 GHz a 11 GHz y de 10 GHz a 66 GHz (Manzo, 2021).

2.3.1.4. Redes Inalámbricas de Área Amplia (WWAN)

Las redes inalámbricas de área amplia se caracterizan por abarcar distancias de más de 50 kilómetros y generalmente operan en frecuencias con licencia (Tuqueres, 2019). Este tipo de redes se implementan en áreas extensas como ciudades o países, utilizando sistemas de satélites o antenas distribuidas por un proveedor de servicios de Internet. Principalmente, hay dos tecnologías disponibles para estas redes: la telefonía móvil y los satélites (Talón, 2016).

2.3.2. Ventajas de las Tecnologías de Comunicación Inalámbrica

Las Tecnologías de comunicación inalámbrica, ofrecen ventajas frente a las redes cableadas. A continuación, se presenta un listado de sus beneficios generales que comparten todas las tecnologías de comunicación inalámbricas:

- **Acceso Remoto:** Posibilitan la entrada y manejo a distancia de dispositivos y sistemas, simplificando la supervisión y administración desde lugares distantes (Cajas K. , 2022).

- **Movilidad:** La capacidad de comunicarse mientras se desplaza libremente es una característica clave de todas las tecnologías inalámbricas, lo que habilita a los usuarios para emplear dispositivos y acceder a información desde cualquier punto dentro del alcance de la red (Tomalo, 2018).
- **Flexibilidad de ubicación:** La colocación flexible de dispositivos y estaciones base es posible en todas las tecnologías inalámbricas, ya que no están restringidas por cables físicos. Esto simplifica la adaptación de la red a diversos entornos y requisitos variables (Tomalo, 2018).
- **Escalabilidad:** Cada tecnología inalámbrica tiene la capacidad inherente de ser escalable, lo que implica que puede crecer y ampliarse sin dificultad mediante la adición de nuevos dispositivos o nodos a la red (Guamán, 2016).

Las redes inalámbricas son cruciales hoy en día al proporcionar conectividad entre distintos dispositivos sin la necesidad de cableado físico, siendo esto fundamentales para impulsar el Internet de las cosas (IoT), facilitando comunicación entre dispositivos inteligentes para la recopilación de datos y la automatización de procesos. Gracias a los continuos avances en velocidad, alcance y eficiencia energética, Wi-Fi se ha vuelto omnipresente tanto en hogares como espacios públicos y corporativos, destacando en el desarrollo de proyectos basados en el monitoreo, automatización y control en entornos tan diversos como ciudades inteligentes, hogares conectados, redes de sensores médicos y demás, es por ello que a continuación se muestra información más detallada sobre la tecnología de comunicación Wi-fi.

2.4.Tecnología Wi-Fi

Wi-Fi es un estándar internacional que se implementa en los niveles físico y de enlace del modelo OSI sobre un canal inalámbrico. Utiliza ondas electromagnéticas de radio en

frecuencias de alrededor de 2,4 GHz o 5 GHz para transmitir datos sin necesidad de cableado (Manzo, 2021). Aunque en sus inicios tenía un ancho de banda considerablemente menor que las redes cableadas, versiones más recientes como "n" pueden alcanzar velocidades de hasta 600 Mbps y "ac" hasta 1 Gbps, haciendo que la diferencia sea casi imperceptible en entornos profesionales. Su amplia adopción se debe a la reducción de costos, estándares consolidados y producción a gran escala (Talón, 2016).

2.4.1. Estándares que certifica la Alianza Wi-Fi

Los estándares WiFi han ido evolucionando, integrando nuevas tecnologías y bandas de frecuencia no licenciadas que permiten conexiones más rápidas y estables de dispositivos de forma inalámbrica (Mendoza, 2022). A continuación, en la tabla 1 se muestra un cuadro comparativo de los estándares más relevantes.

Tabla 1

Cuadro comparativo de los estándares IEEE 802.11 más relevante

Estándar	802.11a	802.11b	802.11g	802.11n	802.11ac
Velocidad	Hasta 54 Mbps	Hasta 11 Mbps	Hasta 54 Mbps	Hasta 600 Mbps	Hasta 6900 Mbps
Banda de Frecuencia	5 GHz	2.4 GHz	2,4 GHz	2.4/5 GHz	5 GHz
Canales	8 canales sin superposición	3 canales superpuestos	3 canales superpuestos	Hasta 4 canales sin superposición	Hasta 8 canales sin superposición
Modulación	OFDM	DSSS/CCK	OFDM	OFDM/MIMO	OFDM/MU-MIMO
Tecnologías clave	20 MHz de ancho de banda	Secuencia directa (DSSS)	Orthogonal Frequency Division Multiplexing (OFDM)	Entrada/salida múltiple (MIMO)	Multiuser MIMO (MU-MIMO)
Año de lanzamiento	1999	1999	2003	2009	2013

Nota. Esta tabla muestra los estándares más relevantes de IEEE 802.11, resaltando las características fundamentales de cada uno.

Fuente: (Mendoza, 2022) & (Castillo L. , 2023) & (Talón, 2016)

2.4.2. Características de la tecnología Wi-Fi

- Basado en el estándar 802.11 del IEEE para redes WLAN
- Opera en bandas de frecuencia no licenciadas: 2.4GHz y 5GHz
- Alta interoperabilidad entre dispositivos al usar protocolos y frecuencias globalmente estandarizados
- Velocidades de última generación de hasta 9600 Mbps (802.11ax)
- Capacidades MU-MIMO y canales de 160MHz de ancho de banda
- Seguridad robusta a nivel de enlace y cifrado (WPA2, WPA3)

2.4.3. Ventajas del uso de la Tecnología Wi-Fi

La tecnología Wi-Fi es hoy una de las tecnologías predilectas en conectividad digital por su agilidad, economía y capacidad de adaptación a diferentes requerimientos.

A continuación, se mencionan algunas de las ventajas de trabajar con esta:

- WiFi permite desplegar una red de área local de forma inalámbrica en pocos minutos, sin necesidad de cableado estructurado. Esto reduce significativamente los costos y tiempo de instalación (Mendoza, 2022).
- Es posible reubicar los puntos de acceso WiFi según requerimientos cambiantes de conectividad. También, los usuarios pueden moverse con sus dispositivos en el área de cobertura sin perder conectividad (Castillo L. , 2023).
- Comparando tecnologías cableadas e inalámbricas, WiFi tiene un menor costo por usuario y por área de cobertura. Se evitan gastos en cableado, insumos, instalación y mantenimiento asociados al cable (Tuqueres, 2019).

2.4.4. Aplicaciones de la Tecnología Wi-Fi

La tecnología Wi-Fi se emplea en diversas aplicaciones de redes de sensores inalámbricos IoT para monitorear y transmitir datos en tiempo real desde sensores

dispersos hacia una plataforma centralizada (Mendoza, 2022). A continuación, se mencionan varios ejemplos de aplicaciones de la tecnología wifi:

- En edificios inteligentes, sensores Wi-Fi monitorean la temperatura, humedad y calidad del aire, conectándose a una red central para mejorar el control climático y la eficiencia energética (Castillo L. , 2023).
- En agricultura de precisión, sensores Wi-Fi recolectan datos de suelo y clima en campos, transmitiéndolos a una estación base para optimizar procesos agrícolas como el riego y la fertilización (Talón, 2016).
- En gestión de residuos urbanos, contenedores con sensores Wi-Fi envían datos de llenado y temperatura a una plataforma centralizada, mejorando la recolección de basura y reduciendo costos (Tuqueres, 2019).

Como se menciona anteriormente, la tecnología Wi-Fi es esencial para potenciar proyectos IoT, es por ello que, se ha tomado como la tecnología de comunicación central para el desarrollo del proyecto de monitoreo de contenedores de residuos infecciosos ubicados en los laboratorios clínicos de la ciudad de Ibarra, para lo cual también se requiere realizar un estudio acerca de la recolección y manejo de los mismos.

2.5. Desechos de Residuos Infecciosos

Los principales residuos infecciosos peligrosos en los laboratorios clínicos de Ibarra incluyen muestras biológicas como sangre, suero y orina, cultivos microbiológicos, materiales contaminados durante análisis como gasas y guantes, agujas y vidriería rota. Contienen agentes patógenos que pueden causar enfermedades y requieren tratamiento previo antes de la eliminación para evitar contagios, como la esterilización en autoclave, la incineración o métodos químicos (González, 2017).

2.5.1. Clasificación de los Desechos de Residuos Infecciosos

En la tabla 2, se muestra la clasificación de los desechos de residuos infecciosos presentada según la ministra de Salud Pública y el ministro Del Ambiente los Desechos Sanitarios.

Tabla 2

Clasificación de los desechos de residuos infecciosos

Tipos de Desechos	Clasificación	Definición
Desechos Sanitarios	Desechos biológico-infecciosos	Es el material médico en contacto con sustancias corporales y agentes infecciosos, con riesgos para la salud. Incluye elementos sin filo y los de áreas de aislamiento.
	Desechos corto-punzantes	Son desechos que presentan riesgo de corte o pinchazo, como fragmentos de plástico duro, que estuvieron en contacto con sangre, fluidos corporales o cultivos infecciosos, y podrían causar accidentes infecciosos.
	Desechos anatomopatológicos	Se trata de órganos, tejidos y productos de la concepción, así como cadáveres o partes de animales infectados y fluidos corporales de procedimientos médicos. Se excluye la orina y el excremento que no provengan de áreas de aislamiento.
Desechos farmacéuticos	Desechos farmacéuticos no peligrosos	Son medicamentos caducados de bajo riesgo que se descomponen por reacciones con elementos del ambiente como el agua, oxígeno o luz.
	Desechos farmacéuticos peligrosos	Son medicamentos caducados o de baja calidad que representan un alto riesgo para la salud y el medio ambiente. Incluyen fármacos citotóxicos y materiales de administración con propiedades genotóxicas, citostáticas e inmunomoduladoras.
	Otros residuos o desechos peligrosos:	Se refieren a residuos con propiedades corrosivas, reactivas, tóxicas, inflamables o radiactivas, que representan un peligro para la salud y el medio ambiente según las leyes correspondientes.
	- Desechos radiactivos - Desechos químicos peligrosos - Desechos de dispositivos médicos con mercurio	

Nota. Esta tabla muestra la clasificación de los desechos de residuos infecciosos propuesta por la “Ordenanza sustitutiva a la ordenanza que regula la gestión de los desechos sanitarios peligrosos en el cantón Ibarra”

Fuente: (GADM-I, 2019)

2.5.2. Fases de la Gestión Integral de Desechos Infecciosos

Generación: Los desechos se producen en todas las áreas del laboratorio como muestras biológicas, medios de cultivo, material usado contaminado, y residuos de equipos. Es crucial la segregación por tipo y riesgo (GADM-I, 2019).

Almacenamiento: Los desechos se almacenan temporalmente en sitios identificados, con acceso restringido, en contenedores rígidos, impermeables, con tapa y bolsas plásticas. Se evita que el personal lo manipule directamente (GADM-I, 2019).

Transporte: Se realiza exclusivamente en carros especiales a prueba de filtraciones y Golpes, debidamente señalizados. El personal utiliza ropa, guantes y mascarilla de protección durante esta actividad (GADM-I, 2019).

Eliminación: Técnicas usadas son la incineración de alta temperatura, la autoclave por vapor, la microwavación y los procesos químicos para desinfectar o inertizar los desechos (GADM-I, 2019).

Disposición Final: Se realiza en rellenos sanitarios autorizados, con celdas exclusivas para desechos infecciosos pretratados, con registro detallado de las cantidades transportadas y procesadas diariamente (GADM-I, 2019).

2.5.3. Métodos de Recolección

Antes de que se realice la recolección externa, se debe realizar la recolección interna de los desechos infecciosos en recipientes plásticos exclusivos, con ruedas, tapa y de fácil transporte. Los desechos se deben clasificar y empaquetar adecuadamente para llevarlos desde el origen hacia las áreas de almacenamiento temporal y luego al destino final,

siguiendo los protocolos establecidos por cada laboratorio clínico (Espinoza, 2022). Cuando los contenedores de dichos desechos se encuentren llenos, se aplican los métodos de recolección; entre los principales destacan: método de parada fija, método de acera y método de contenedores, como se muestra en la figura 4.

Figura 4

Métodos de recolección de desechos infecciosos

Método de Parada Fija



Método de Contenedores



Método de Acera



Fuente: (España, 2023) y (Espinoza, 2022)

2.5.3.1. Método de Parada Fija

En este método, el vehículo recolector se estaciona en puntos fijos establecidos previamente. El personal del establecimiento de salud transporta y entrega los desechos infecciosos debidamente embalados y etiquetados directamente en dicho vehículo especial. Es importante coordinar día y hora exacta (Gabriela, 2023).

2.5.3.2. Método de Acera

Los desechos embalados se colocan en la acera exterior del establecimiento a una hora estipulada para ser recogidos manualmente por el personal del vehículo recolector.

Para este método el almacenamiento temporal debe ser en la entrada del edificio con todas las medidas de seguridad (España, 2023).

2.5.3.3. Método de Contenedores

Se utilizan contenedores móviles de alta resistencia ubicados temporalmente dentro del establecimiento de salud, los cuales una vez llenos se trasladan al camión recolector mediante montacargas o equipos automotorizados. El vehículo solo efectúa el cambio de contenedores llenos por vacíos (Rivera, 2023).

El algoritmo de Dijkstra se usa para encontrar la ruta más corta en un grafo ponderado, representando conexiones entre ubicaciones. En el caso de la recolección de desechos, este grafo puede representar calles y obstáculos desde el punto de recolección hasta los laboratorios. Al asignar pesos a las conexiones basados en factores como tiempo de viaje o seguridad, el algoritmo encuentra la ruta más eficiente para los recolectores. Esta optimización minimiza recursos y riesgos de exposición, mejorando la gestión de desechos infecciosos en entornos clínicos.

2.6. Algoritmo Dijkstra

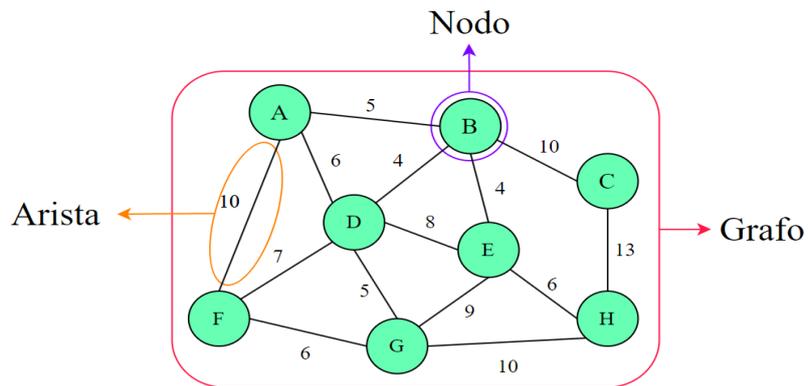
El algoritmo de Dijkstra encuentra los trayectos más cortos entre un punto de origen y todos los demás puntos en un grafo dirigido. Para su ejecución, se asume que las longitudes de las aristas son enteras y positivas, y que el grafo es dirigido, es decir, las aristas solo permiten viajar en una dirección. Su metodología radica en explorar todos los posibles caminos más cortos desde el punto inicial hacia los demás puntos del grafo. Cuando encuentra el camino más corto hacia los demás puntos desde el origen, el algoritmo termina. Sin embargo, no puede utilizarse en grafos con aristas de coste negativo, ya que, al elegir siempre el nodo con la distancia menor, podría omitir nodos que, en iteraciones posteriores, podrían reducir el costo total del camino al atravesar aristas con costos negativos (Ortega, 2019).

2.6.1. Características

- Encuentra la ruta más corta desde un nodo origen a todos los demás nodos en una red, calculando la ruta de costo mínimo entre dos puntos (Landa, 2021).
- El algoritmo requiere que la red se modele matemáticamente como un grafo dirigido, es decir que las conexiones entre nodos tienen una dirección y un valor o peso específico (Ortiz, 2020).
- Dijkstra minimiza los tiempos de cómputo al ir guardando en memoria los resultados intermedios de las rutas encontradas, mejorando tiempos con redes grandes (Daquilema, 2021).
- Evalúa una fracción de las posibles rutas en una red, aquellas que parecen más promisorias en cada iteración, por ello garantiza siempre el camino más corto disponible (Daquilema, 2021).

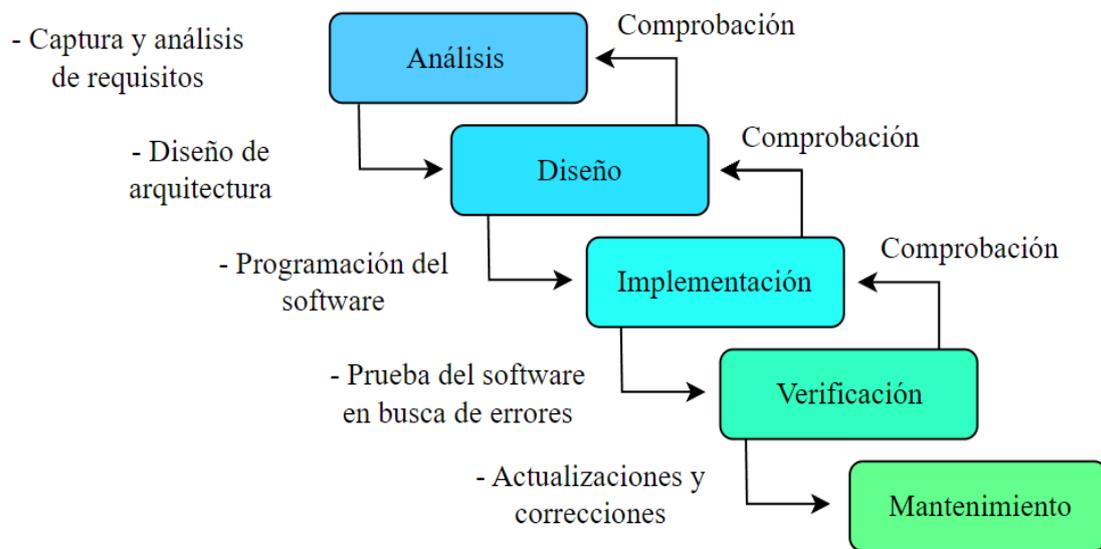
2.6.2. Grafos

Un grafo representa el mapa a través del cual se quiere encontrar la ruta de costo mínimo, donde los nodos modelan localizaciones/objetos y las aristas representan la interconexión entre ellos con un costo asociado (Daquilema, 2021). Funciona asignando distancias tentativas de los nodos y relajando esas distancias mediante la exploración de nodos adyacentes si se encuentra un camino más corto, utilizando una cola de prioridad para seleccionar siempre el siguiente nodo a expandir con la menor distancia tentativa. Al final, permite calcular la ruta de costo mínimo desde un nodo inicial a cualquier otro nodo en grafos dirigidos o no dirigidos (Landa, 2021). En la figura 5, se puede observar de forma clara, las partes de un grafo.

Figura 5*Partes del algoritmo de Dijkstra**Fuente: (Landa, 2021)**Elaborado por: La Autora*

2.7. Metodología en Cascada

El modelo de cascada, también conocido como metodología waterfall, implica dividir la fase de desarrollo de software en cinco etapas secuenciales, como se muestra en la figura 6. El proceso de desarrollo se inicia con un análisis que evalúa la viabilidad del proyecto y define los requisitos. Esta fase proporciona un pliego de condiciones y un plan financiero, seguido de una definición detallada de los requisitos. Luego, en el diseño, se formula una solución específica y se desarrolla la arquitectura del software. La implementación traduce este diseño en código, mientras que la fase de prueba verifica su cumplimiento de requisitos. Tras superar las pruebas, el software se lanza en su entorno previsto. Finalmente, en la fase de servicio, se autoriza su aplicación productiva y se inicia el mantenimiento y mejora continua del producto (Delgado, 2020).

Figura 6*Metodología en cascada*

Fuente: (Delgado, 2020)

Elaborado por: La Autora

3. Capítulo III: Diseño del Sistema

En este capítulo se centrará en la capa de diseño del modelo en cascada, que es la siguiente etapa tras analizar el marco teórico de las redes de sensores inalámbricas; para lo que se abordará el diseño y desarrollo del sistema de monitoreo de contenedores para los residuos infecciosos generados en los laboratorios clínicos de Ibarra. Antes de proceder con el diseño, se realizará un estudio previo de la situación actual en la que operan los laboratorios clínicos de la ciudad, para conocer los métodos de recolección de desechos infecciosos que utilizan, privado o público. Posteriormente, se hará un diseño general del sistema con un diagrama de bloques, listando los requisitos que debe cumplir y eligiendo la tecnología de comunicación a usar. Luego, se definirán mediante diagramas cómo están estructurados los distintos componentes del sistema, cómo se relacionan entre sí y cómo fluye la información entre ellos. Finalmente, se trabajará en los detalles específicos del prototipo, escogiendo las rutas de datos, configurando los dispositivos de la red inalámbrica, modelando la base de datos y diseñando las interfaces para el usuario final.

3.1. Descripción del Área de Estudio

En este apartado, se analizarán las rutas actuales del Gobierno Autónomo Descentralizado Municipal San Miguel de Ibarra para recoger los desechos infecciosos de los contenedores de los laboratorios clínicos, incluyendo mapeo y estado de las vías de acceso, lo que permitirá tener una línea base para la futura propuesta de ruteo optimizado. Después, se determinarán las coordenadas exactas para cada laboratorio clínico que se incorporará al sistema y se analizará el procedimiento completo que siguen las entidades antes y durante la recolección por parte de los vehículos municipales.

3.1.1. La Ciudad de Ibarra

Ibarra, conocida como "La Ciudad Blanca", fue fundada el 28 de septiembre de 1606 por el capitán Cristóbal de Troya, bajo la orden del entonces presidente de la Audiencia de Quito, Miguel de Ibarra. Tras el devastador terremoto del 16 de agosto de 1868, la ciudad quedó prácticamente en ruinas y tuvo que ser reconstruida a partir de 1872 (Ruiz, 2023).

Se encuentra ubicada geográficamente en un privilegiado entorno de la región interandina del norte de Ecuador. Está asentada en un valle al pie del volcán Imbabura, atravesado por varios ríos como el Tahuando, el Chorríco, el Ajaví y el principal río Chota, a una altura de 2.215 m.s.n.m. (GADM-I, 2023).

Ibarra es la cabecera municipal del Cantón Ibarra y capital de la Provincia de Imbabura, siendo la urbe más grande y poblada de la región. Además, ha sido designada como sede administrativa de la Región 1, que abarca las provincias de Esmeraldas, Carchi, Sucumbíos e Imbabura, debido a su estratégica ubicación geográfica (Ayala, 2022).

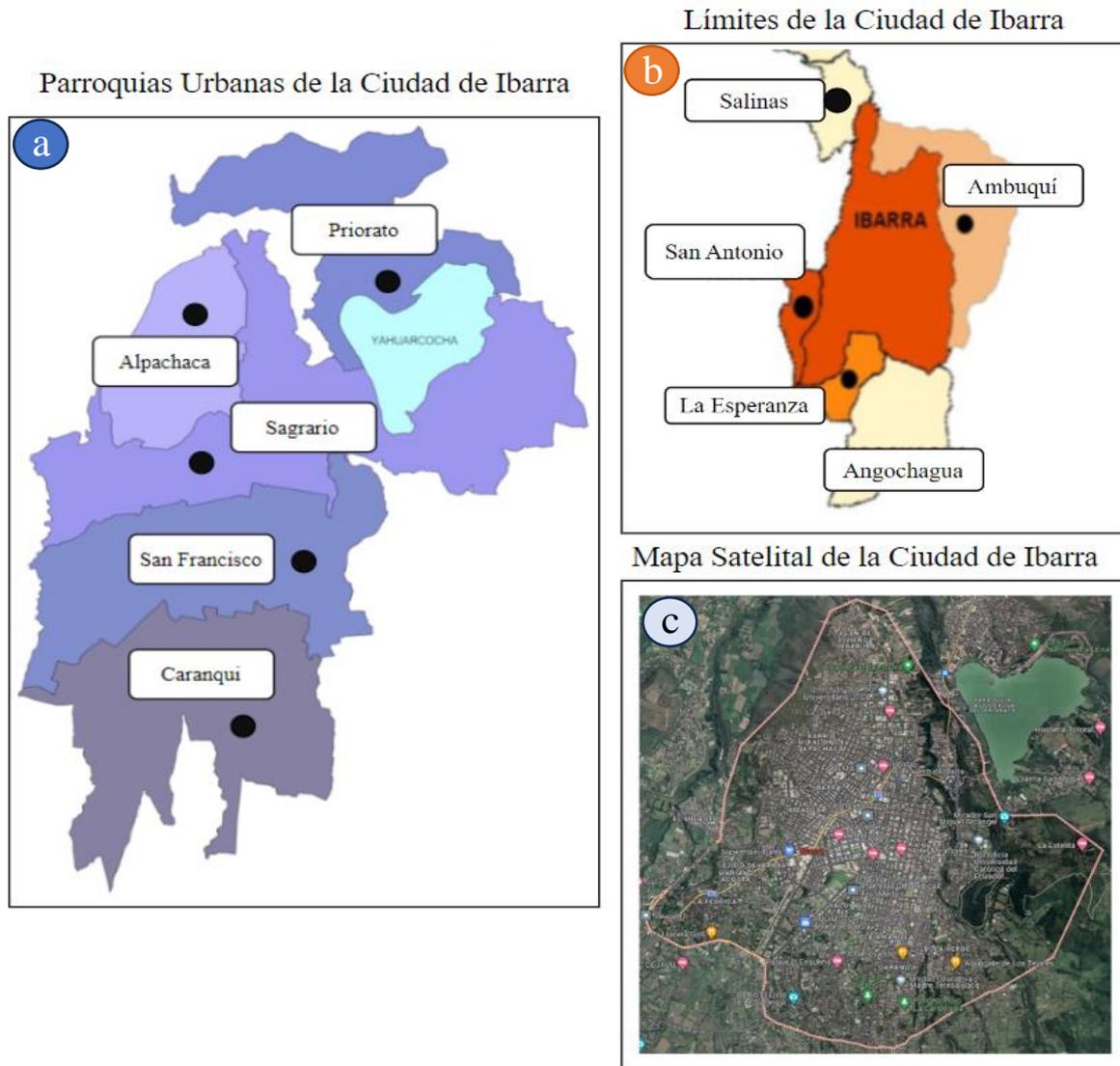
Dentro del área urbana consolidada, la ciudad de Ibarra se divide administrativa y territorialmente en 5 parroquias: San Francisco, El Sagrario, Caranqui, Alpachaca y Priorato (GADM-I, 2023).

La ciudad de Ibarra tiene sus límites definidos de la siguiente manera: al norte por las elevaciones de la Loma de Guayabillas y la Hacienda que la separan de la parroquia rural de Ambuquí, al sur el río Tahuando marca el límite con la parroquia rural de San Antonio, al este por las Lomas de Azaya que la dividen de las parroquias rurales La Carolina y Salinas, y al oeste por el río Tahuando y la quebrada Chorrera que establecen la frontera con las parroquias rurales La Esperanza y Angochagua (SENPLADES, DOCPLAYER, 2013).

En la figura 7 se presentan diferentes aspectos de la ciudad de Ibarra: la sección “a” muestra la distribución de sus parroquias urbanas; la sección “b” señala sus límites, y en la sección “c” se observa una vista del mapa satelital.

Figura 7

Ubicación geográfica de la Ciudad de Ibarra junto con sus límites y sus Parroquias



*Fuente: Secretaría Nacional de Planificación, (SENPLADES, 2021) & Google Maps
Elaborado por: La Autora*

3.1.2. Agencia de Aseguramiento de la Calidad de los Servicios de Salud y Medicina Prepagada (ACCESS-Imbabura)

El Instituto Nacional de Estadística y Censos (INEC) publicó un registro con datos estadísticos sobre los permisos de funcionamiento que se entregaron a los diferentes establecimientos de salud en Ecuador entre 2016 y 2024.

De ese registro, se analizaron los datos específicos del cantón Ibarra, provincia de Imbabura. En la figura 8, Se observa que, durante el año 2023, la Agencia de Aseguramiento de la Calidad de los Servicios de Salud y Medicina Prepagada (ACCESS) otorgó 545 permisos de funcionamiento a distintos tipos de establecimientos de salud en Ibarra (INEC, 2023).

Figura 8

Total de permisos de funcionamiento otorgado a establecimientos de salud del cantón Ibarra, de la provincia de Imbabura

zona	provincia	canton	Permisos de Funcionamiento Total
ZONA 1	IMBABURA	ANTONIO ANTE	77
ZONA 1	IMBABURA	COTACACHI	64
ZONA 1	IMBABURA	IBARRA	545
ZONA 1	IMBABURA	OTAVALO	177
ZONA 1	IMBABURA	PIMAMPIRO	20
ZONA 1	IMBABURA	SAN MIGUEL DE URQUQUI	19
Total			902

Fuente: (Instituto Nacional de Estadística y Censos (INEC) - 2023)

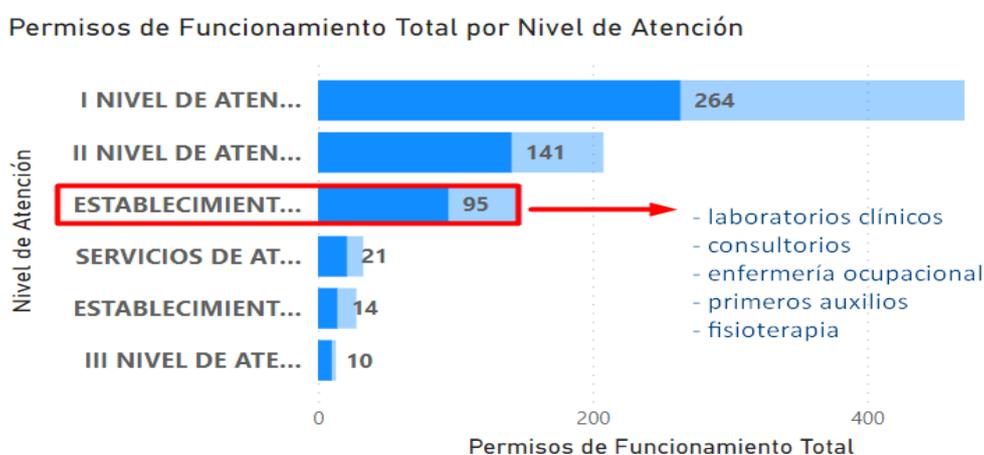
Los establecimientos de salud se clasifican en varios niveles de atención: primer nivel (puestos de salud, centros de salud y consultorios), segundo nivel (hospitales básicos y hospitales generales), establecimientos que brindan servicios de apoyo (consultorios, enfermería ocupacional, primeros auxilios, fisioterapia y laboratorios clínicos), servicios de atención móvil (ambulancias), establecimientos de apoyo indirecto (centros de

capacitación, investigación y bancos de sangre) y tercer nivel (hospitales e institutos especializados) (Cando, 2023).

De acuerdo con la tabla estadística presente por el INEC del 2023, presente en la figura 9, se puede observar que se han otorgado 95 permisos de funcionamiento, por parte del ACCESS, a los establecimientos de salud correspondientes al nivel de atención de establecimientos que brindan servicios de apoyo, destacando entre estos a los laboratorios clínicos (ACCESS, 2024).

Figura 9

Permisos de funcionamiento otorgado a establecimientos de salud del cantón Ibarra, de la provincia de Imbabura, divididos en niveles de atención



Fuente: (Instituto Nacional de Estadística y Censos (INEC) - 2023)

Para obtener información detallada sobre los laboratorios clínicos más destacados de la ciudad de Ibarra, se realizó un oficio dirigido a la delegada Provincial de ACCESS Imbabura, Dirección Zonal 1 - Ibarra, la Bioquímica Jaqueline Chiza. En respuesta, ella proporcionó datos sobre los establecimientos ubicados específicamente en las parroquias centrales de la ciudad: El Sagrario y San Francisco.

En la tabla 3, se identifica la información recibida que clasifica a los laboratorios clínicos según su nivel de complejidad: alta, mediana y baja. En ella se pueden verificar los nombres de los laboratorios correspondientes a cada categoría. Esta clasificación

permite identificar aquellos centros con capacidades más especializadas para realizar análisis clínicos complejos, así como aquellos enfocados en pruebas de rutina o menor complejidad.

Tabla 3

Listado de los Laboratorios Clínicos Ubicados en el centro de la Ciudad de Ibarra

Nombre Comercial	Tipo Establecimiento	Cantón	Parroquia
Riolab	Laboratorio de análisis clínico de alta complejidad	Ibarra	El Sagrario
Diagnostikate. ec Consultorios Médicos y Laboratorio Clínico Biomolecular	Laboratorio de análisis clínico de mediana complejidad	Ibarra	El Sagrario
Laboratorio Clínico Cedilab Milenium	Laboratorio de análisis clínico de mediana complejidad	Ibarra	El Sagrario
Laboratorio Clínico Automatizado	Laboratorio de análisis clínico de mediana complejidad	Ibarra	El Sagrario
Megalaboratorio Clínico	Laboratorio de análisis clínico de mediana complejidad	Ibarra	El Sagrario
Laboratorio Clínico Microbiolac	Laboratorio de análisis clínico de baja complejidad	Ibarra	El Sagrario
Reclab	Laboratorio de análisis clínico de baja complejidad	Ibarra	El Sagrario
Laboratorios Clínicos Dolab	Laboratorio de análisis clínico de baja complejidad	Ibarra	El Sagrario
Aryamge Lab	Laboratorio de análisis clínico de baja complejidad	Ibarra	El Sagrario
Dispensario Médico municipal	Laboratorio de análisis clínico de baja complejidad	Ibarra	El Sagrario
Laboratorio Clínico Autolab	Laboratorio de análisis clínico de alta complejidad	Ibarra	San Francisco
Laboratorio Clínico automatizado	Laboratorio de análisis clínico de baja complejidad	Ibarra	San Francisco
Laboratorio Clínico Ferti Vida	Laboratorio de análisis clínico de baja complejidad	Ibarra	San Francisco
Health Lab	Laboratorio de análisis clínico de baja complejidad	Ibarra	San Francisco

Fuente: (ACCESS Imbabura, Dirección Zonal 1 – Ibarra)

3.1.3. Protocolo de Manejo de Recolección de Desechos infecciosos

Según la “Gestión interna de residuos y desechos generados en los establecimientos de salud” establecida por el Ministerio de Salud Pública en 2019, los desechos sanitarios cuentan con una normativa específica para realizar la recolección de la manera más adecuada.

Los desechos biológicos-infecciosos deben almacenarse en recipientes de color rojo, rotulados con el símbolo de riesgo biológico. Estos recipientes solo deben llenarse hasta las tres cuartas partes de su capacidad máxima. Además, las bolsas de almacenamiento primario no deben superar los 10 kilogramos de peso por cada una.

En cuanto a los desechos corto-punzantes, la normativa establece que se deben utilizar recipientes de plástico resistente a prueba de perforaciones, evitando el contacto con las manos. Estos recipientes deben estar debidamente etiquetados y su capacidad de llenado es hasta las tres cuartas partes, momento en el cual deben cerrarse herméticamente.

La frecuencia de recolección de desechos infecciosos varía según el volumen de desechos generados y las horas de operación del establecimiento. En aquellos que trabajan 8 horas diarias, la recolección interna debe ser diaria. En los establecimientos que operan más de 8 horas diarias, la recolección debe realizarse dos veces al día. En el caso de laboratorios clínicos que generan poca cantidad de desechos, estos deben retirarse cuando los contenedores estén llenos hasta las tres cuartas partes o si hay desechos que se descomponen rápidamente. Es importante tener en cuenta que la recolección de desechos infecciosos debe llevarse a cabo en horarios de menor circulación de pacientes, evitando en lo posible el cruce con las rutas de alimentos y otras áreas sensibles del establecimiento de salud (Andramuño, 2019).

3.1.4. Métodos de Recolección de Desechos Infecciosos de la Ciudad de Ibarra

La Agencia de Aseguramiento de la Calidad de los Servicios de Salud y Medicina Prepagada (ACCESS), menciona que es de carácter obligatorio que todas las instituciones de salud deban contar con un proveedor de servicios externo privado o público calificado para la recolección externa, transporte, tratamiento y disposición final de los desechos infecciosos (Rubio, 2020).

Actualmente, en la ciudad de Ibarra, El Ministerio del Ambiente, Agua y Transición Ecológica (MAATE) siendo el ente que establece las políticas, normas y lineamientos para la adecuada gestión integral de los desechos infecciosos, registra que los establecimientos de salud como laboratorios clínicos cuentan con dos opciones para la recolección de sus desechos infecciosos: un proveedor de servicio público proporcionado por la Dirección de Gestión Ambiental/Residuos Sólidos del GADM Ibarra (GADM-I, 2019) y proveedores privados, como Incinerox y Gadere (de Veolia), reconocidos por ser empresas líderes en el manejo de residuos peligrosos y desechos infecciosos en la región (Veolia, 2024) (Incinerox, 2024). Esta dualidad en la prestación de servicios permite a los laboratorios clínicos elegir la opción más conveniente para sus necesidades, garantizando un manejo adecuado y seguro de estos residuos.

3.1.4.1. Servicio Público de Recolección de Desechos Infecciosos

Para obtener la información correspondiente al servicio público de recolección de desechos infecciosos empleado por el municipio del cantón Ibarra, se tuvo que realizar un oficio dirigido al Ingeniero Álvaro Castillo, alcalde del Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra, el cual, una vez aprobado, dio paso para realizar una reunión técnica con la Ingeniera Karla Lima Paredes, Técnico de Gestión Ambiental de la Dirección de Gestión Ambiental/Residuos Sólidos.

En la reunión realizada, la Ingeniera Karla Lima supo mencionar que: *“Anteriormente, la Dirección de Gestión Ambiental/Residuos Sólidos operaba con un horario de recolección de lunes a sábado de 07:00 a 13:30 horas Sin embargo, debido a la incorporación de nuevas instituciones médicas como laboratorios clínicos, consultorios médicos y centros de especialidades, se ha implementado un nuevo esquema de horarios divididos en dos turnos: miércoles, jueves y viernes por un lado, y jueves, viernes y sábado por otro, donde la frecuencia de recolección es cada 15 días para la totalidad de los 377 centros médicos registrados, independientemente del nivel de llenado de sus contenedores de desechos infecciosos”*. Debido a que se realiza la recolección sin importar el nivel de llenado de los contenedores, existen ocasiones en las que éstos se encuentran casi vacíos o, por el contrario, demasiado llenos. La primera situación representa una pérdida de tiempo y recursos al acudir a recolectar contenedores con muy poco volumen de desechos. Mientras que la segunda situación, con contenedores excesivamente llenos, podría significar un peligro para la salud, ya que se trata de desechos contaminantes y su almacenamiento prolongado aumenta los riesgos de exposición. La Ingeniera Karla Paredes explica que: *“No todos los laboratorios clínicos de la ciudad de Ibarra generan la misma cantidad de desechos infecciosos, ha habido casos en los que algunos laboratorios clínicos han generado menos de 1 libra de peso luego de 15 días de su anterior recolección de desechos, esto debido a que son establecimientos de nuevo funcionamiento o se encuentran alejados del centro de la ciudad”*.

Cuando el vehículo recolector llega a uno de los laboratorios clínicos para recolectar los desechos infecciosos, como se observa en la figura 10, se realiza un pesaje previo antes de cargar los desechos en el camión. La ingeniera Karla Lima menciona que: *“Los encargados del servicio pesan manualmente cada funda sellada que contiene los residuos*

infecciosos utilizando básculas portátiles, debidamente calibradas y certificadas, registrando de forma detallada el peso obtenido, la fecha, la hora y demás datos relevantes del laboratorio en cuestión. La tarifa establecida para el servicio de recolección y tratamiento de estos desechos peligrosos es de \$1,75 por cada kilogramo recolectado, por lo tanto, el peso total registrado durante el pesaje previo determinará el costo a facturar posteriormente al laboratorio clínico”.

Figura 10

Pesaje previo a la recolección de los residuos infecciosos



Fuente: (Ingeniera Karla Lima Paredes, Técnico de Gestión Ambiental de la Dirección de Gestión Ambiental/Residuos Sólidos)

El método de recolección aplicado en el cantón Ibarra sigue un orden específico determinado por la ubicación geográfica de cada centro médico, lo que permite optimizar las rutas y la eficiencia del proceso, “*Las rutas de recolección de los desechos infecciosos se realiza por zonas, es decir que se da inicio en una zona específica abarcando a todos los laboratorios clínicos ubicados dentro de esta para luego continuar con la siguiente zona y así sucesivamente*” menciona la ingeniera Karla Lima.

Actualmente, el municipio cuenta con un solo vehículo recolector especializado y acondicionado para el manejo de desechos infecciosos, cumpliendo con los requerimientos de la Norma Técnica Ecuatoriana NTE INEN 2841:2014 - Gestión

Ambiental. Ecoetiquetado Ambiental. Requisitos. La Ingeniera Karla Lima afirma que: *“Este vehículo está equipado con sistemas de refrigeración y fácil lavado para garantizar condiciones higiénicas y seguras en caso de derrames o accidentes durante el transporte de estos desechos infecciosos. Sin embargo, se han presentado inconvenientes debido a fallas técnicas presentes en este único vehículo recolector, lo que ha obligado a la Dirección de Gestión Ambiental/Residuos Sólidos a posponer el proceso de recolección en algunos horarios”*. El almacenamiento prolongado de estos desechos contaminantes representa un riesgo alto, especialmente para aquellas instituciones médicas que generan gran cantidad de residuos infecciosos. La Ingeniera Karla Paredes menciona que: *“Considerando el inconveniente presentado, el GADM Ibarra ha decidido adquirir un nuevo vehículo recolector en un futuro próximo, lo cual reforzará su capacidad, evitará retrasos y permitirá retomar el horario de recolección previamente establecido”*.

Cabe mencionar que cualquier institución médica que desea unirse al sistema de recolección de desechos infecciosos puede hacerlo realizando el procedimiento que se detalla a continuación por la ingeniera Karla Lima: *“En primer lugar, debe enviar un oficio formal dirigido al señor alcalde de la ciudad de Ibarra, solicitando su incorporación al mencionado sistema. En dicho oficio, la institución debe especificar sus datos generales, ubicación y una estimación de la cantidad de desechos infecciosos que generará mensualmente. Una vez que la solicitud sea recibida y evaluada por la Alcaldía, si cumple con los requisitos establecidos, se procederá a autorizar la unión de la institución al sistema de recolección. Posteriormente, se le notificará a la institución solicitante sobre la aprobación de su solicitud y se le brindará la información necesaria para integrarse al sistema, incluyendo los días y horarios de recolección, así como los protocolos a seguir para el correcto manejo y disposición de sus desechos infecciosos”*.

Figura 11

Información del estado actual de la recolección de desechos infecciosos de la ciudad de Ibarra “Reunión con la Ingeniera Karla Lima Paredes, Técnico de Gestión Ambiental de la Dirección de Gestión Ambiental/Residuos Sólidos”



Fuente: (Ingeniera Karla Lima Paredes, Técnico de Gestión Ambiental de la Dirección de Gestión Ambiental/Residuos Sólidos)

❖ Laboratorios Clínicos sujetos al Método de Recolección Pública

De acuerdo con la documentación proporcionada por la Ingeniera Karla Lima Paredes, Técnico de Gestión Ambiental de la Dirección de Gestión Ambiental/Residuos Sólidos del GADM Ibarra, se puede analizar la situación actual de los laboratorios clínicos tomando en cuenta la cantidad de residuos infecciosos que estos generan.

La Ingeniera Karla Lima indica que: *“En el GADM Ibarra se encuentran registradas 377 instituciones médicas que forman parte del sistema de recolección de residuos infecciosos, entre las que se encontrar centros de especialidades, hospitales, clínicas de especialidades, laboratorios clínicos, clínicas veterinarias y clínicas odontológicas”*.

El análisis se ha centrado en los laboratorios clínicos, considerando dos tipos de instituciones: las que tienen un laboratorio clínico integrado en sus instalaciones y las que trabajan solo como laboratorios clínicos independientes. Esta evaluación abarca tanto establecimientos que llevan mucho tiempo operando como los recientes inaugurados.

Tomando en cuenta los datos brindados por la Ingeniera Karla Lima, los laboratorios clínicos se pueden clasificar según la cantidad de desechos infecciosos que generan. En este sentido, se identifican tres grupos principales:

- **Los mayores generadores:** En este grupo hay 3 laboratorios clínicos, entre los que destaca el hospital San Vicente de Paul como el principal generador de residuos infecciosos por su ubicación estratégica en el centro de la ciudad.
- **Generadores considerables:** Este grupo lo forman 14 laboratorios clínicos que producen muchos desechos infecciosos. Entre ellos, se encuentran algunos centros de especialidades médicas.
- **Generadores menores:** Finalmente, hay 12 laboratorios clínicos que generan una cantidad muy pequeña de residuos infecciosos. Este grupo incluye clínicas y laboratorios que han iniciado sus operaciones recientemente o que se ubican en zonas alejadas del centro de la ciudad.

Para entender la información, en la tabla 4 se presentan los laboratorios clínicos registrados en el sistema de recolección de desechos infecciosos proporcionada por la Ingeniera Karla Lima, organizado según la cantidad de residuos infecciosos que generan. Cada grupo está resaltado con un color específico, considerando la siguiente codificación:

- **Color rojo:** Laboratorios clínicos que generan gran cantidad de desechos infecciosos.
- **Color naranja:** Laboratorios clínicos que generan una cantidad considerable de desechos infecciosos.
- **Color azul:** Laboratorios clínicos que generan poca cantidad de desechos infecciosos.

Tabla 4

Listado de los Laboratorios Clínicos de la Ciudad de Ibarra, que se encuentran registrados dentro del sistema de recolección de desechos infecciosos brindado por el GADM Ibarra

	Nombre Laboratorio Clínico	Dirección
	Laboratorio Bioozono	Av, Fray Vacas Galindo 702 y Carlos Almeida
	Laboratorio Healthlab	Av. Retorno 11-12 y Rio Blanco (ceibos)
	Hospital San Vicente de Paul	Luis torres Vargas y Gonzáles Gómez Jurado
	Biosalud Laboratorio Clinico	Sánchez y Cifuentes 6-75 y Garcia Moreno
	Centro de Especialidades fym	Av. Juan de Salinas Esquina y Obispo Mosquera
	Laboratorio Clínico Orion	Rafael Troya 2-17 y Bolivar
	Laboratorio Clínico Soluciones Diagnosticas Homelab	Av. Teodoro Gómez 13-151 y Bolivar
	Laboratorio Clininlab	Luis Vargas Torres 3-51 y Carvajal
	Laboratorio Alfa Multi Salud	Luis Vargas Torres 4-47 y Gonzalo Gómez Jurado
	Laboratorio Microbiolac	Oviedo 8-26 y Olmedo
	Centro de Especialidades Médicas Florecer	Colón 4-88 y Rocafuerte
	Centro de Especialidades	Hugo guzmán Lara y Rosa Andrade de Larea
	Centro de Especialidades Scolpire Copro	Gabriela Mistral 3-53
	Centro de Especialidades Médicas Estéticas y Salud Velife	Quito 6-42 y av. 13 de abril
	Centro Médico de Especialidades Nuestra Clínica	Sucre 13-66 y Teodoro Gómez
	Centro de Especialidades "ImbaSalud"	Obispo Mosquera 5-56 y Bolívar
	Centro de Especialidades Medicas	Cristóbal de Troya 8-189 y Fray Vacas Galindo
	Laboratorio auto lab	Sánchez y Cifuentes 9-81 y Pedro Moncayo
	Medicca Lab	Borrero 4-69 y Sucre

	Laboratorio Clínico Riolab	Flores 9-86 y Chica Narváez
	Cruz Roja Ecuatoriana Emocentro	Eugenio Espejo 9-61 y Velasco
	Laboratorio Clínico BioMedic	Galo Plaza Lasso y Judith Granda Almeida
	Clínica Médica Fértil	Cesar Morales Granda 4-76 y Salvador Dalí
	Clínica Avellaneda	Av. Mariano Acosta 17-42 y Eleodoro Ayala
	Laboratorio ITCA	el Oro y 13 de abril
	Clínica Famar	Emperatriz Mejía de Gómez Jurado 235
	Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura	Juan José Flores y Manuel de la Chica Narváez.
	Clínica Mariano Acosta	Av. Mariano acosta 11-16 y Cabezas Borja
	Laboratorio "New Lab"	Av. Pérez Guerrero 6-75 y Olmedo
	Clínica de Heridas	Ibarra/Rocafuerte y José Miguel Leoro

Fuente: (Gestión Ambiental de la Dirección de Gestión Ambiental/Residuos Sólidos del GADM Ibarra)

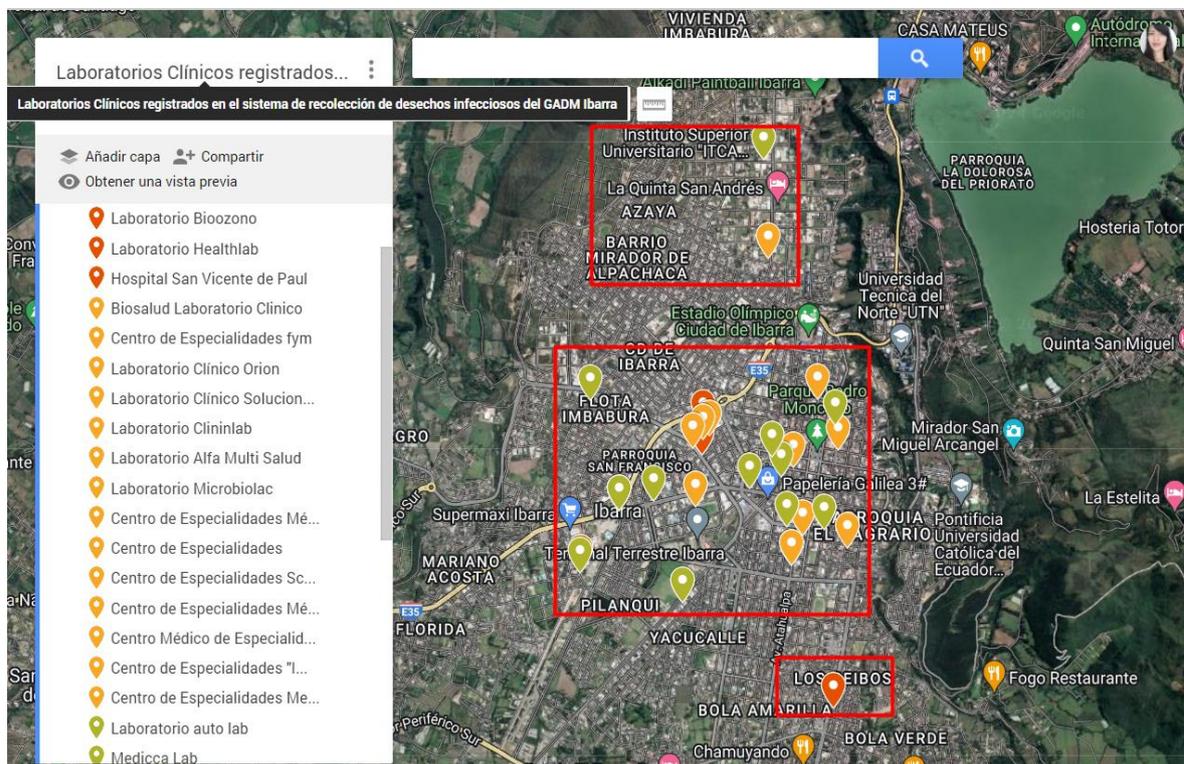
En el mapa satelital proporcionado por la aplicación "My Maps" se muestra las ubicaciones precisas de los laboratorios clínicos evaluados. Cada establecimiento se distingue con un distintivo de color específico, el cual, como se mencionaba anteriormente representa la cantidad de desechos infecciosos que generan.

Esta codificación por colores permite una visualización clara y efectiva de la situación actual de los laboratorios clínicos registrados en el sistema de recolección de desechos infecciosos del Gobierno Autónomo Descentralizado Municipal (GADM) de Ibarra.

Observando el mapa de la figura 12, se puede apreciar que la mayoría de los laboratorios clínicos que generan una gran cantidad y una cantidad considerable de desechos infecciosos se encuentran ubicados en el centro de la ciudad de Ibarra. Por otro lado, los laboratorios clínicos que generan poca cantidad de desechos infecciosos se distribuyen en diferentes zonas, tanto en el centro como en áreas más alejadas de la ciudad.

Figura 12

Ubicación de los Laboratorios Clínicos Registrados el Sitema de Recolección del Municipio de la ciudad de Ibarra



Fuente: (Sistema de Recolección de desechos infecciosos del GADM Ibarra & My Maps)

Elaborado por: La Autora

3.1.4.2. Servicio Privado de Recolección de Desechos Infecciosos

En la provincia de Imbabura existen varios proveedores privados que brindan servicios de gestión integral de los residuos peligrosos. Dentro de los principales proveedores que operan en la ciudad de Ibarra se encuentran Incinerox y Gadere (de Veolia), dos empresas multinacionales líderes en el sector ambiental. Ambas compañías cuentan con la infraestructura, el personal capacitado y los permisos necesarios otorgados por el Ministerio del Ambiente, Aguas y Transición Ecológica para realizar de manera segura y responsable la recolección, transporte, tratamiento y disposición final de los

desechos infecciosos generados por los diversos laboratorios clínicos ubicados en Ibarra y sus alrededores.

- **Incinerox**

Incinerox ofrece un servicio especializado en la incineración de desechos, comenzando con una auditoría exhaustiva de los residuos generados por cada laboratorio cliente, brindando asesoramiento sobre su separación y embalaje. Proveen contenedores retornables y resistentes al calor, y establecen rutas de recolección optimizadas para minimizar distancias y evitar tráfico (Incinerox, 2024). Los desechos son transportados en vehículos especializados a su planta de incineración, donde se destruyen a temperaturas extremas, con un monitoreo constante de emisiones para asegurar bajos impactos ambientales. Los costos se basan en el peso de los desechos y el método de incineración, con opciones de paquetes anuales, y los laboratorios deben solicitar una auditoría y cotización detallada para contratar el servicio (Incinerox, 2024).

- **Gadere**

Gadere ofrece un servicio de recolección de desechos con un enfoque en logística eficiente y tecnología de rastreo. Inician con una evaluación detallada de los desechos generados por cada laboratorio, proporcionando contenedores especiales etiquetados y clasificados. Las rutas de recolección se optimizan según la ubicación de los laboratorios y se coordinan con ellos para no interferir en sus operaciones (Veolia, 2024). El personal capacitado realiza la recolección en vehículos con GPS y asegura el cumplimiento de las regulaciones. En cuanto a los costos, Gadere ofrece tres planes: el plan básico, que incluye recolección cada 40 días sin importar el volumen de desechos; el plan estándar, que permite recolecciones más frecuentes según el volumen de desechos generado; y el plan premium, que incluye recolección personalizada y servicios adicionales como el manejo

de desechos especiales. Los laboratorios deben solicitar un análisis detallado y cotización para contratar sus servicios (Veolia, 2024).

❖ **Laboratorios Clínicos Sujetos al Método de Recolección Privado**

De acuerdo con los datos recopilados del Sistema Informático de Establecimientos Prestadores de Servicios de Salud (SACCS), proporcionados por la Bioquímica Jaqueline Chiza, delegada Provincial de ACESS Imbabura, Dirección Zonal 1 - Ibarra, se ha logrado obtener un listado más completo de los laboratorios clínicos de la ciudad de Ibarra que se encuentran sujetos al método de recolección privado de desechos infecciosos. Esta información ha sido complementada y verificada con los datos disponibles en las páginas web oficiales de ACESS Imbabura (ACESS, 2024).

En base a la información obtenida, se ha realizado una clasificación de los laboratorios clínicos en tres categorías principales, determinadas por la cantidad de desechos infecciosos que producen. La categoría roja engloba a aquellos establecimientos que generan una gran cantidad de estos residuos, lo que implica un mayor riesgo y la necesidad de implementar medidas de manejo más rigurosas. Por otro lado, la categoría naranja representa a los laboratorios que producen una cantidad considerable de desechos infecciosos, mientras que la categoría verde identifica a los establecimientos que generan una cantidad muy baja de estos residuos.

A continuación, se presenta en la tabla 5, el listado de los laboratorios clínicos sujetos al método de recolección privado, su respectiva categoría de generación de desechos infecciosos y la dirección exacta de cada uno de ellos:

Tabla 5

Listado de los Laboratorios Clínicos de la Ciudad de Ibarra, que hacen uso de gestores externos privados que brindan servicio de recolección de desechos infeccioso

	Nombre del Laboratorio Clínico	Dirección
	Zurita & Zurita Laboratorios	Av. Mariano Acosta

	DOLAB Laboratorio Clínico	Av Cristobal de Troya 1255
	Hospital de Ibarra IESS	Víctor Manuel Guzmán 1143
	Hospital Clínica Metropolitana	Parque ciudad Blanca, Galo Pl. Lasso S/N
	Sociedad Lucha Contra el Cáncer – Hospital SOLCA – Matriz	Rafael Sanchez 10 67 y, Ibarra 100150
	Laboratorio Clínico Clínica Ibarra	Jorge Davila Meza
	Laboratorio Clínico Automatizado	Juan Jose Flores 9-86, Manuel de la Chica Narvaez
	RECLAB Laboratorio Clínico Integral	Bolívar 1-29 Entre Mejía, y, Ibarra
	Laboratorio Clínico Machala	Luis Vargas Torres
	Unidad de Servicios Médicos	Av Jorge Guzmán Rueda
	Medylab	Teodoro Gómez 528 entre Sucre
	Laboratorio San Cayetano	Luis Vargas Torres 4-29
	Laboratorio Clínico Meissen	Av. Fray Vacas Galindo
	DiagnoLab	Calixto Miranda y Suarez

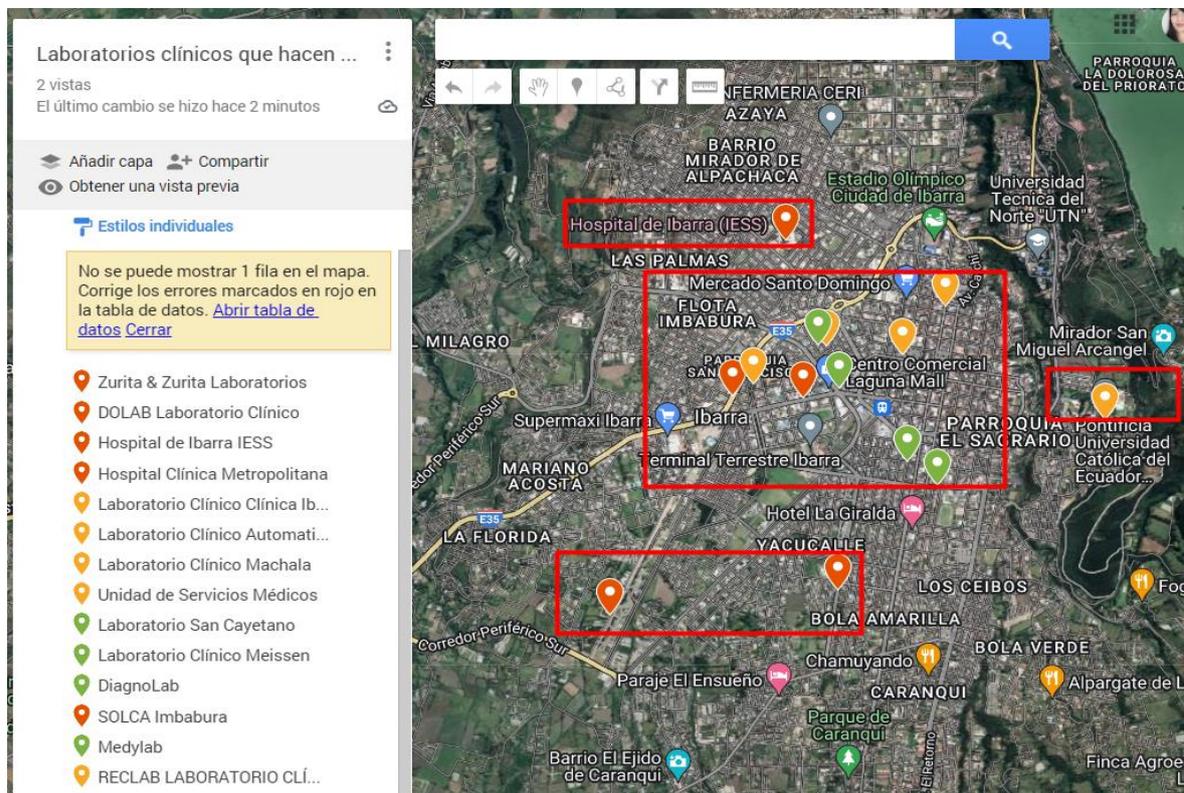
Fuente: (ACCESS, Establecimientos de Salud Acreditados/Certificados, 2024)

Para visualizar de mejor manera las ubicaciones de los laboratorios clínicos, se presenta la Figura 13, un mapa satelital de la aplicación My Maps, en el cual se puede apreciar el nombre de los laboratorios junto con sus respectivas direcciones, y un identificativo de color que representa la cantidad de desechos infecciosos que cada laboratorio clínico genera.

Analizando este mapa, se puede concluir que la gran mayoría de los laboratorios clínicos que utilizan el servicio privado de recolección de desechos infecciosos se encuentran ubicados en el centro de la ciudad, encontrándose entre ellos varios hospitales y laboratorios clínicos privados.

Figura 13

Ubicación de los Laboratorios Clínicos de la ciudad de Ibarra que utilizan el método de recolección de desechos infecciosos proporcionado por gestores privados



Fuente: (ACCESS, Establecimientos de Salud Acreditados/Certificados, 2024)

Elaborado por: La Autora

3.1.5. Métodos Interno de Recolección de Desechos Infecciosos realizado en los Laboratorios Clínicos

En los laboratorios clínicos, el manejo adecuado de desechos infecciosos es esencial debido a los riesgos para la salud y el medio ambiente. El proceso inicia con la clasificación de los diferentes tipos de residuos, como cultivos, muestras biológicas y material punzocortante (Landy, 2021). Como se puede observar en la figura 14, esta separación se realiza en el punto mismo de generación para evitar mezclas peligrosas.

Tabla 6*Etiquetas de los Desechos Sanitarios Peligrosos*

Código Etiquetas	Tipo de Desecho	Descripción
Q.86.01	Infecciosos	Materiales contaminados con agentes infecciosos, desechos biológicos, vacunas inutilizadas.
Q.86.02	Anatomo patológicos	órganos, tejidos y partes corporales extraídos mediante cirugía
Q.86.05	Cortopunzantes	Objetos cortopunzantes contaminados biológicamente.
Q.86.08	Fármacos Caducados	Fármacos caducados y fuera de especificaciones.
Q.86.09	Desechos Químicos de Laboratorio	Químicos caducados o fuera de especificaciones

Fuente: Listado Nacional de Desechos Peligrosos y Especiales & Registro Oficial N° 856.

Una vez clasificados, los desechos se envasan en recipientes específicos según su categoría: los punzocortantes, como agujas y bisturís, se colocan en contenedores rígidos a prueba de perforaciones; los desechos infecciosos no punzantes, como cultivos y muestras biológicas, se almacenan en bolsas rojas de plástico resistente; y los desechos líquidos infecciosos, como fluidos corporales, se envasan en recipientes herméticos o bolsas de seguridad. Todos los contenedores están etiquetados y codificados por colores para facilitar su identificación y manejo adecuado (MSP, 2019) (MSP, 2019).

Luego, utilizando carros de transporte exclusivos, los desechos envasados y sellados se trasladan desde los puntos de generación hasta un área de almacenamiento temporal dentro del laboratorio clínico (García F. , 2019).

Esta área de almacenamiento está debidamente acondicionada, señalizada y cuenta con acceso restringido solo para el personal autorizado y capacitado en el manejo de estos residuos. Los contenedores se organizan por tipo de desecho, y se un registro detallado de los volúmenes generados, fechas y áreas de procedencia (Rivera, 2023).

Para evitar inconvenientes por la acumulación de contenedores de desechos infecciosos totalmente llenos, debido a posibles retrasos en la recolección externa, el ACCESS ha recomendado contar con un área de almacenamiento que pueda utilizarse en casos de emergencia (Landy, 2021). En la figura 15, se observa las áreas de almacenamiento temporal de los desechos finales generados en el Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura, resaltando que se cuenta con un área de almacenamiento de reserva.

Figura 15

Áreas de almacenamiento temporal de los laboratorios clínicos



Fuente: Laboratorio Clínico – Sindicato de Choferes Profesionales de Imbabura

Finalmente, en la fecha y hora acordadas con la empresa recolectora, el vehículo especializado acude al laboratorio. Los desechos se cargan de forma segura en el camión y se realiza el pesaje correspondiente antes del retiro, siguiendo estrictos protocolos de bioseguridad en todo momento para proteger al personal, los pacientes y el medio ambiente

(Alvarado, 2020).

3.2. Requisitos del Sistema

El estándar ISO/IEC IEEE 29148 propone un enfoque estructurado para la selección de requisitos en tres niveles principales: Requisitos de Stakeholders o Usuarios, que capturan las necesidades de las partes interesadas; Requisitos del Sistema, que definen las funciones y propiedades que el sistema debe cumplir; y Requisitos de Arquitectura, que establecen la estructura, patrones de diseño y restricciones técnicas. Estos tres niveles de requisitos deben estar alineados y trazados entre sí, adaptándose al contexto específico del proyecto y documentándose formalmente para facilitar su gestión y seguimiento. En la tabla 7, se especifica el listado de los requisitos junto con sus respectivas abreviaturas.

Tabla 7

Lista de Requerimientos con sus respectivas abreviaturas

Requerimientos	Abreviatura
Requerimientos de Stakeholders	StRS
Requerimientos del Sistema	SyRS
Requerimientos de Arquitectura	ARS

Fuente: Estándar ISO/IEC IEEE 29148

3.2.1. Stakeholders

Los stakeholders son todas las personas, grupos u organizaciones que tienen algún tipo de interés directo o indirecto en las actividades del proyecto. Es fundamental identificar a todos los stakeholders y comprender sus funciones, necesidades y expectativas, ya que sus requisitos influirán directamente en el diseño y desarrollo exitoso del proyecto. La Tabla 8 enumera y describe los distintos stakeholders involucrados en este proyecto en particular.

Tabla 8*Lista de Requerimientos con sus respectivas abreviaturas*

STAKEHOLDERS			
Número	Involucrados	Clasificación Directo/Indirecto	Función
1	Tnlg. Edwin Vásquez Mera	Indirecto	Secretario General del Sindicato Provincial de Choferes Profesionales de Imbabura
3	Ing. Karla Lima	Indirecto	Técnico de Gestión Ambiental de la Dirección de Gestión Ambiental/Residuos Sólidos del GADM Ibarra.
4	MSc. Jaime Michilena	Directo	Director de Trabajo de Titulación
5	MSc. Luis Suárez	Directo	Asesor de Trabajo de Titulación
6	Melannie Calero	Directo	Autor de Trabajo de Titulación

*Elaborado por: La Autora***3.2.2. Requerimientos de Stakeholders**

Los requerimientos de los stakeholders, son las necesidades, expectativas y restricciones que tienen las diferentes partes interesadas involucradas en el proyecto, definiendo lo que el producto final debe hacer para satisfacerlas. Estos requerimientos capturan aspectos relevantes desde la perspectiva de los distintos grupos de stakeholders, tales como funcionalidades operativas, usabilidad, accesibilidad y cualquier otro aspecto relevante, con el fin de asegurar que el producto final cumpla con los objetivos operacionales y las necesidades de los usuarios finales. En la tabla 9, se observan los requerimientos de Stakeholders, dividido en requerimientos operacionales y requerimientos de usuario.

Tabla 9*Requerimientos de Stakeholders*

StRS					
N°	REQUERIMIENTOS	PRIORIDAD			
		Alta	Media	Baja	
Requerimientos Operacionales					
StRS1	El dispositivo debe mantener una conexión inalámbrica mediante wi-fi para el envío de datos a la plataforma.	x			
StRS2	El dispositivo debe tener una capacidad de funcionamiento de al menos 8 horas diarias.	x			
StRS3	El dispositivo debe contar con un sensor ultrasónico que permita medir el nivel de llenado de los contenedores.	x			
StRS4	El dispositivo debe contar con un microcontrolador para la captura y envío de datos del nivel de los contenedores.	x			
StRS5	El dispositivo debe contar con baterías para alimentación del sensor ultrasónico y microcontrolador.		x		
Requerimientos de Usuarios					
StRS6	La aplicación móvil contará con el inicio de sesión para diferentes tipos de usuarios.	x			
StRS7	Los usuarios visualizarán el listado de laboratorios clínicos con un identificativo de color (rojo, naranja y verde) según el nivel de llenado de sus contenedores.		x		
StRS8	El usuario visualizará en la aplicación móvil, los niveles de llenado de los contenedores, especificando a qué laboratorios clínicos pertenecen.		x		
StRS9	La aplicación móvil incluirá un mapa de rutas realizado con el algoritmo de Dijkstra para encontrar la ruta más corta hacia los laboratorios clínicos que necesiten recolección de desechos	x			
StRS10	Cada tipo de usuario (recolector, laboratorio clínico o administrador) podrá enviar notificaciones según la necesidad que tengan.	x			
StRS11	El usuario administrador podrá verificar los informes de los contenedores de cada laboratorio clínico	x			
StRS12	El usuario administrador podrá registrar nuevos usuarios, editar y eliminar.			x	

Elaborado por: La Autora

Los requerimientos vistos en la tabla 9, se plantearon en base a las necesidades propuestas en la entrevista realizada a la ingeniera Karla Lima, Técnico de Gestión Ambiental de la Dirección de Gestión Ambiental/Residuos Sólidos del GADM Ibarra, y a los técnicos responsables de los laboratorios clínicos, donde se especificó que el sistema debe contar con la visualización del nivel de llenado de los contenedores de residuos infecciosos, la visualización de rutas y el envío de notificaciones.

3.2.3. Requerimientos del Sistema

Los requerimientos del sistema establecen las características y especificaciones necesarias para que el sistema cumpla con sus objetivos y funcione correctamente. Abarcan interfaces, funcionalidades, rendimiento, restricciones físicas y demás parámetros que guiarán su diseño e implementación según el sistema que se va a implementar.

Tabla 10

Requerimientos del Sistema

SyRS				
N°	REQUERIMIENTOS	PRIORIDAD		
		Alta	Media	Baja
Requerimientos de Interfaz				
SyRS1	Acceso al IDE Arduino para la programación de la placa microcontroladora.	x		
SyRS2	Establecimiento de conectividad con la red Wi-fi	x		
SyRS3	Almacenamiento del nivel de llenado de los contenedores de residuos infecciosos en la base de datos.	x		
SyRS4	Aplicación del algoritmo de Dijkstra para el cálculo de la ruta más corta hacia los laboratorios clínicos.		x	
SyRS5	La interfaz debe mostrar los contenedores con colores representativos del nivel de llenado (rojo, naranja, verde) para facilitar la visualización rápida.		x	
Requerimiento de Uso				
SyRS6	Instalación de dispositivos de monitoreo en los contenedores de residuos infecciosos.		x	
SyRS7	Recolección de datos sobre el estado de los contenedores (nivel de llenado).	x		

SyRS8	El dispositivo debe estar conectado a las baterías para alimentar los elementos electrónicos.	x	
Requerimiento de Performance			
SyRS9	El dispositivo debe estar ubicado en la tapa de los contenedores para medir el nivel de llenado.		x
SyRS10	Lectura continua del nivel de llenado de los contenedores.	x	
SyRS11	Los datos recolectados deben ser almacenados en la base de datos para su visualización posterior.	x	
SyRS12	Determinación de la ruta más corta hacia los laboratorios para la recolección.	x	
SyRS13	El sistema de monitoreo debe ser capaz de operar durante al menos 8 horas continuas con la batería del dispositivo sin necesidad de recarga.		
Requerimiento de Modo/Estado			
SyRS14	El sistema debe estar encendido y funcionando durante el día para la recolección de datos.	x	
SyRS15	El dispositivo debe transmitir datos periódicamente mediante Wi-Fi para conocer el estado del contenedor.	x	
Requerimientos Físicos			
SyRS16	El sistema de monitoreo debe estar instalado en las tapas de los contenedores de residuos.	x	
SyRS17	El sistema debe ser fácilmente desmontable para que los recolectores puedan vaciar el contenedor sin dañar los elementos electrónicos.		x
SyRS18	El sistema de monitoreo no debe ocupar mucho espacio dentro del contenedor.		x
SyRS19	Los elementos electrónicos deben estar protegidos dentro de una carcasa hermética.		x

Elaborado por: La Autora

3.2.4. Requerimientos de la Arquitectura

Los requerimientos de la arquitectura especifican las características y componentes necesarios de hardware, software y sus interacciones para el adecuado funcionamiento del sistema. Incluyen requisitos lógicos de estructura y flujos de datos, requisitos de diseño arquitectónico, especificaciones de hardware como dispositivos, procesamiento y almacenamiento, requerimientos de software como plataformas, así como requisitos eléctricos de alimentación y consumo energético. En la tabla 11, se muestra una tabla más detallada de los requerimientos de la arquitectura.

Tabla 11*Requerimientos de la arquitectura*

ARS					
N°	REQUERIMIENTOS	PRIORIDAD			
		Alta	Media	Baja	
Requerimientos Lógicos					
ARS1	La placa microcontroladora debe manejar pines digitales y analógicos para conectar y controlar los sensores.	x			
ARS2	Programación de microcontrolador mediante el IDE Arduino.	x			
ARS3	Implementación de una base de datos para almacenar y procesar los datos de nivel de llenado recolectados.	x			
ARS4	Cálculo de rutas mediante el algoritmo de Dijkstra para determinar la ruta óptima de recolección.	x			
Requerimientos de la APP					
ARS5	Interfaz intuitiva y fácil de usar, con diseño claro y accesible.	x			
ARS6	Mensajes de confirmación para el correcto inicio de sesión.		x		
ARS7	Mensajes de confirmación para el correcto envío de notificaciones.		x		
ARS8	Colores diferenciados (rojo, verde, amarillo) para facilitar la comprensión.		x		
ARS9	Manuales de usuario claros y fáciles de entender.	x			
ARS10	La aplicación debe ser compatible con dispositivos móviles comunes.	x			
Requerimientos de Software					
ARS11	Uso de librerías y plugins compatibles con el microcontrolador y sensores utilizados en el sistema.	x			
ARS12	Almacenamiento de datos en la base de datos en la nube para garantizar accesibilidad y seguridad.	x			
ARS13	El sistema de monitoreo debe recopilar datos de nivel de llenado en tiempo real y mostrar esta información en la aplicación móvil.	x			
ARS14	Los datos obtenidos con los sensores serán almacenados en una base de datos para su posterior análisis.	x			
ARS15	La aplicación móvil debe mostrar de manera interactiva las rutas óptimas para la recolección de desechos en un mapa.	x			

Requerimientos de Hardware		
ARS16	Sensor ultrasónico para el cálculo del nivel de llenado de los contenedores de residuos infecciosos.	x
ARS17	Placa microcontroladora capaz de recolectar los datos de los sensores y conectarse a la red Wi-Fi.	x
ARS18	Baterías recargables para alimentar los sensores y el microcontrolador.	x

Elaborado por: La Autora

3.3. Componentes del Sistema

La parte de componentes del sistema ayuda a identificar los materiales y componentes más comunes que se utilizan para construir sistemas de monitoreo. De modo que, luego de realizar un análisis teniendo en cuenta factores importantes como el costo, la facilidad de conseguirlos, la funcionalidad y compatibilidad con otros componentes, su rendimiento y su confiabilidad, se pueda seleccionar los materiales y componentes más adecuados para el sistema. A continuación, se muestran los componentes seleccionados, considerando cada etapa del desarrollo del sistema.

3.3.1. Etapa de Sensado

En la etapa de sensado, se analizan los sensores disponibles evaluando su rango de medición, precisión y consumo energético para seleccionar los más adecuados. También se estudian los módulos microcontroladores ESP32, que permiten establecer conexiones inalámbricas mediante WiFi. Se explora cómo configurar y programar estos módulos para conectarse a un nodo gateway, que actúa como punto de acceso para recolectar y procesar los datos de los sensores, facilitando la comunicación en tiempo real y el monitoreo remoto del sistema.

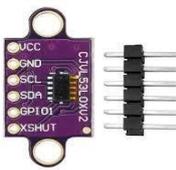
❖ Análisis de Sensores para Medición de Nivel de Contenedores

Es importante analizar las características de los sensores para medir el nivel de llenado de los contenedores de desechos infecciosos. Se consideran tres métodos: medición de peso (báscula), sensores de nivel (detectan presencia de material en distintas alturas) y

medición de distancia (compara la distancia del sensor a la superficie de los desechos con la altura del contenedor). Para este proyecto, se seleccionó el método de medición de distancia, que ofrece alta precisión sin contacto directo con los desechos. La tabla 12 presenta un cuadro comparativo de los sensores más utilizados, destacando opciones como el HC-SR04 (sensor ultrasónico con rango de 2 cm a 4 m), el VL53L0X (sensor de tiempo de vuelo con láser, hasta 2 m), el LiDAR Lite v3 (sensor LiDAR de bajo costo, hasta 40 m), la cámara estéreo (visión estereoscópica para cálculo de distancias) y el radar de onda continua (sensor de ondas electromagnéticas).

Tabla 12

Tabla comparativa de Sensores Medidores de Distancia

Nombre Sensor	HC-SR04	VL53L0X	LiDAR Lite v3	Cámara estéreo	Radar de Onda Continua
Figura					
Tipo	Ultrasónico	Láser ToF	Lidar	Visión artificial	Ondas de radio
Alimentación	5V	2.6V - 3.5V	4.5V - 5.5V	5V	5V
Corriente	15mA	20mA	105mA	500mA	150mA
Frecuencia	40kHz	-	905nm	-	24GHz - 77GHz
Tamaño	45x20x15 mm	4.8x2.9x1.1 mm	20x48x57 mm	Varía según cámara	42x29x13 mm
Rango de Medición	2cm - 4m	1m - 2m	Hasta 40m	Hasta 50m	Hasta 300m
Precisión	±3mm	±3%	±2.5cm	±1cm	±1cm
Costo	Muy bajo (\$2-\$5)	Moderado (\$5-\$10)	Moderado (\$100-\$150)	Moderado (\$50-\$300)	Alto (\$500-\$1000)
Ventajas	<ul style="list-style-type: none"> - Bajo costo - Fácil de usar - Buena precisión a corto alcance 	<ul style="list-style-type: none"> - Alta precisión - Pequeño tamaño - Buen rendimiento en condiciones de poca luz 	<ul style="list-style-type: none"> - Buen rango y precisión - Funciona bien en interiores y exteriores 	<ul style="list-style-type: none"> - Alta precisión - Detección de obstáculos y mapeo 3D - Funciona bien en superficies irregulares 	<ul style="list-style-type: none"> - Muy alto rango y precisión - Funciona bien en condiciones adversas - Preciso en superficies irregulares
Desventajas	<ul style="list-style-type: none"> - Rango limitado - No funciona bien en superficies irregulares 	<ul style="list-style-type: none"> - Rango de medición limitado 	<ul style="list-style-type: none"> - Consumo de energía más alto que sensores ultrasónicos - Precio más alto 	<ul style="list-style-type: none"> - Alto consumo de energía - Procesamiento de imágenes complejo - Sensible a condiciones de iluminación 	<ul style="list-style-type: none"> - Alto costo - Interferencias potenciales con otras señales de radio

absorbente s - No funciona bien en presencia de ruido ultrasónico o	que sensores ultrasónicos	- Complejidad de configuración
---	---------------------------------	--------------------------------------

Nota. En esta tabla se ha recopilado la información más importante sobre los sensores de medición de distancia, más populares del mercado.

Fuente: (Cajas K. , 2022) & (Tuqueres, 2019) & (Hurtado, 2019)

- **Sensor Ultrasónico HC-SR04**

El sensor ultrasónico HC-SR04 se perfila como la opción más adecuada para el desarrollo de nuestro sistema de cálculo del nivel de llenado de contenedores de desechos infecciosos. A diferencia de otros sensores propuestos, como el VL53L0X, el LiDAR Lite v3, la cámara estéreo y el radar de onda continua, el HC-SR04 ofrece una solución económica, compacta y fácil de integrar, lo cual es fundamental en un proyecto de este tipo. Además, su rango de medición de 2 cm a 4 metros es suficiente para la mayoría de los contenedores utilizados en entornos médicos, y su principio de funcionamiento basado en la emisión y recepción de pulsos ultrasónicos lo hace ideal para operar en ambientes con poca iluminación o presencia de obstáculos. Si bien otros sensores pueden ofrecer una mayor precisión o un rango de medición más amplio, el HC-SR04 representa un equilibrio óptimo entre costo, rendimiento y facilidad de implementación para nuestras necesidades específicas.

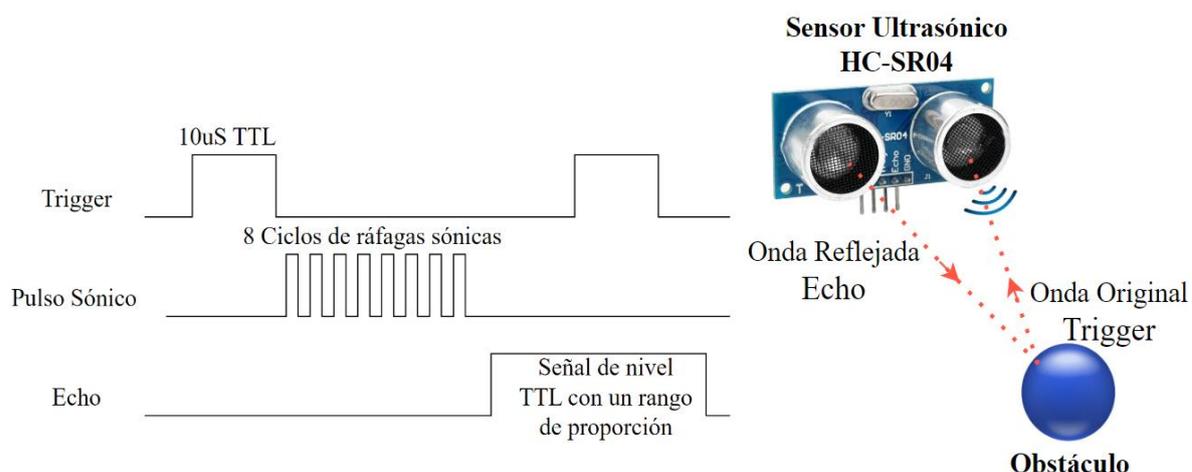
- **Funcionamiento del Sensor Ultrasónico HC-SR04**

El sensor ultrasónico HC-SR04 emite pulsos de ondas ultrasónicas y mide el tiempo que tarda el eco en reflejarse desde los objetos cercanos. Utiliza un transmisor piezoeléctrico para generar las ondas y un receptor para captar el eco. A partir del tiempo

transcurrido y la velocidad del sonido en el aire, calcula la distancia al objeto. Este proceso se repite continuamente, proporcionando mediciones precisas y en tiempo real, lo que lo hace ideal para aplicaciones de detección de obstáculos, medición de niveles y sistemas de proximidad (Mayorga, 2019).

Figura 16

Funcionamiento del Sensor Ultrasónico HC-SR04



Fuente: (Mayorga, 2019)
Elaborado por: La Autora

❖ **Análisis de Módulos Microcontroladores**

Es esencial analizar los módulos microcontroladores disponibles, ya que son clave para conectar los nodos de sensores con la red WiFi. La elección adecuada garantizará una comunicación inalámbrica confiable y eficiente entre los sensores y el sistema de monitoreo del nivel de llenado de los contenedores de desechos infecciosos. Los microcontroladores procesan los datos de los sensores y gestionan su transmisión. Se deben evaluar características como potencia de procesamiento, memoria, compatibilidad con protocolos inalámbricos, consumo energético, facilidad de programación, costo,

tamaño y robustez. A continuación, se presenta una tabla comparativa (tabla 13) con las principales características de los módulos microcontroladores.

Tabla 13

Tabla comparativa de Módulos Microcontroladores

Módulo	ESP32	ESP8266	Raspberry Pi Pico W	Particle Argon	Adafruit Feather Huzzah ESP8266	Wio Terminal
Figura						
Microcontrolador	Tensilica Xtensa LX6 de 32 bits	Tensilica L106 de 32 bits	RP2040 de doble núcleo Arm Cortex-M0+	STM32L476 Arm Cortex-M4	ESP8266	ATSAMD51P19 Arm Cortex-M4F
Frecuencia	240MHz	80-160MHz	133MHz	80MHz	80-160MHz	120MHz
Memoria RAM	520KB	80KB	264KB	128KB	80KB	256KB
Memoria Flash	4MB	512KB	2MB	1MB	512KB	512KB
Voltaje	2.7-3.6V	3.0-3.6V	1.8-5.5V	3.6-5.5V	3.0-3.6V	3.3-5V
Consumo de Corriente	80-260mA	60-170mA	100-200mA	120-220mA	60-170mA	120-200mA
Velocidad de Reloj	240MHz	80-160MHz	133MHz	80MHz	80-160MHz	120MHz
Pines Digitales	34	17	26	27	17	14
Pines Analógicos	16	1	4	12	1	8
WiFi	Sí	Sí	Sí	Sí	Sí	Sí
Bluetooth	Sí	No	No	No	No	Sí

Costo Aproximado	\$5-\$10	\$2-\$5	\$6-\$10	\$20-\$30	\$15-\$20	\$30-\$40
-------------------------	----------	---------	----------	-----------	-----------	-----------

Nota. En esta tabla se resaltan las fortalezas y debilidades de cada opción de manera visual y condensada, facilitando así la selección del módulo microcontrolador más adecuado para el proyecto.

Fuente: (Castillo L. , 2023) & (Gabriela, 2023) & (Cajas K. , 2022)

- **Módulo Microcontrolador ESP32**

El módulo microcontrolador ESP32 se perfila como la opción más adecuada para desarrollar el proyecto de monitoreo del nivel de llenado de contenedores de desechos infecciosos, ya que combina de manera excepcional características clave como su potente procesador de doble núcleo y amplia capacidad de memoria RAM y flash, lo que lo convierte en una opción capaz de manejar tareas complejas de procesamiento de datos y comunicación inalámbrica de manera eficiente; su compatibilidad nativa con WiFi, es importante para la conexión inalámbrica confiable y eficiente de los nodos de sensores con la red WiFi; su versatilidad con abundantes pines de entrada/salida que facilitan la integración con diversos sensores, incluyendo el sensor ultrasónico seleccionado; su bajo consumo energético, ideal para aplicaciones con restricciones de energía; su facilidad de programación, bibliotecas, herramientas de desarrollo y su costo bajo en comparación con otras alternativas como ESP8266, Raspberry Pi Pico W, Particle Argon, Adafruit Feather HUZZAH ESP8266 y Wio Terminal.

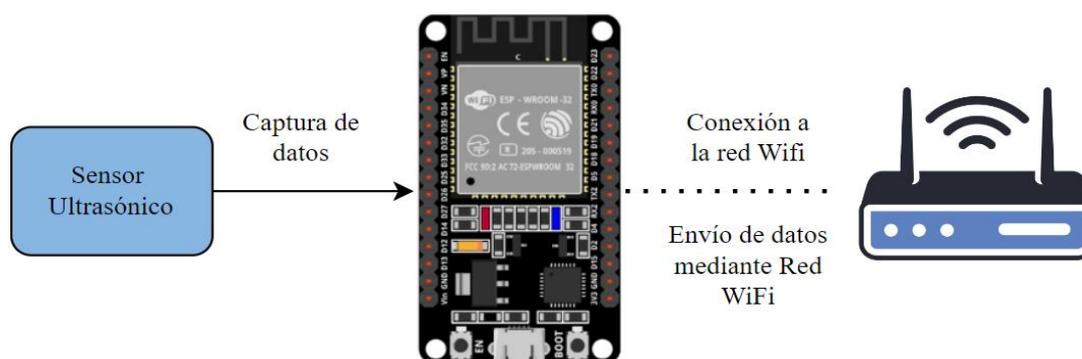
- **Funcionamiento del Módulo ESP32**

Como se observar en la figura 17, cuando el ESP32 se conecta al sensor ultrasónico HC-SR04, utiliza sus pines GPIO para activar el disparo del sensor y medir el tiempo que tarda en recibir el eco reflejado, calculando así la distancia a un objeto mediante la velocidad del sonido; luego, aprovechando su potente procesador de doble núcleo y su módulo WiFi integrado, puede procesar los datos de distancia obtenidos por

el sensor, encapsularlos en paquetes y transmitirlos de forma inalámbrica a través de su interfaz WiFi a un servidor remoto, una aplicación móvil o cualquier otro dispositivo conectado a la red, lo que lo convierte en una solución completa y eficiente para proyectos de monitoreo de distancia y presencia en el ámbito del Internet de las Cosas (Altamirano, 2018).

Figura 17

Funcionamiento del Módulo Microcontrolador ESP32



Fuente: (Altamirano, 2018)
Elaborado por: Calero, 2024

3.3.2. Etapa de Monitoreo

En este proyecto de monitoreo de los contenedores ubicados en los laboratorios clínicos, el ESP32 se encargará de conectarse a la red WiFi utilizando los routers proporcionados por los proveedores de internet presentes en las instalaciones. Como se observa en la figura 18, estos routers actuarán como nodos gateway o puntos de acceso para que el microcontrolador pueda establecer una conexión inalámbrica segura (Chávez, 2020).

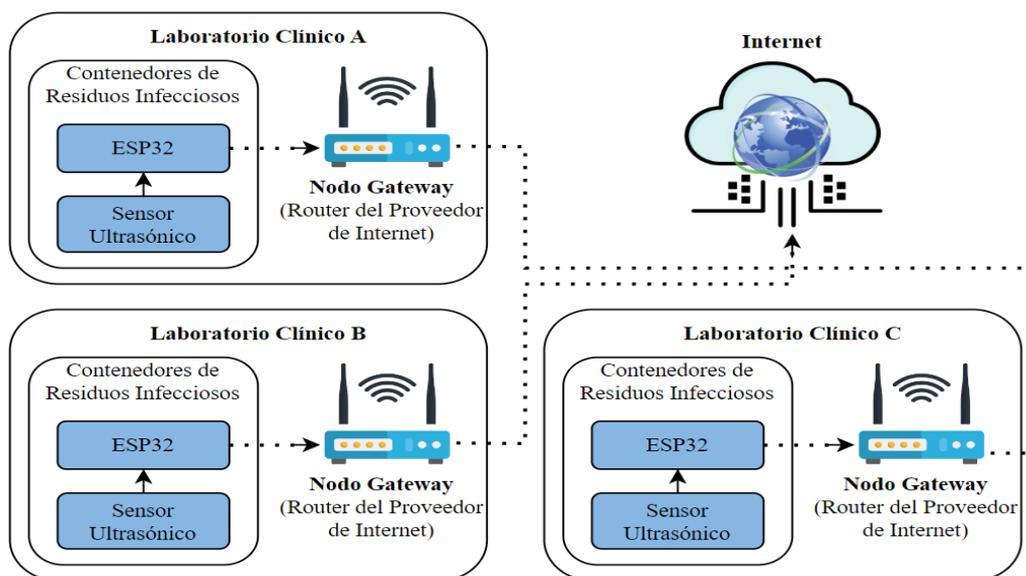
El proceso comenzará con el escaneo de las redes WiFi-disponibles en el área por parte del ESP32. Una vez identificados los routers configurados previamente con sus credenciales de acceso (SSID y contraseña), el microcontrolador procederá a autenticarse y enlazarse a la red inalámbrica mediante los protocolos y estándares correspondientes.

Después de establecer la conexión WiFi, el ESP32 podrá recopilar los datos del sensor ultrasónico HC-SR04, el cual se encargará de medir distancias mediante la emisión y recepción de pulsos ultrasónicos. Estos datos de distancia serán procesados adecuadamente por el microcontrolador (Hurtado, 2019).

A continuación, el ESP32 aprovechará su interfaz WiFi y la conexión inalámbrica establecida para transmitir los datos recopilados a través de la red. Para ello, utilizará protocolos y librerías específicas que permitan el envío de información hacia el servidor central de base de datos donde se almacenará la información y se proyectará en una aplicación móvil. De esta manera, los datos de monitoreo podrán llegar a su destino de forma inalámbrica, sin necesidad de tender cableado en las instalaciones. Esto brinda mayor flexibilidad en la ubicación de los dispositivos de medición dentro de los contenedores de residuos infecciosos de los laboratorios clínicos (Mendoza, 2022).

Figura 18

Funcionamiento del Sistema de Monitoreo



Fuente: (Mendoza, 2022) & (Hurtado, 2019)

Elaborado por: La Autora

3.3.3. Etapa de Almacenamiento de Datos

En la etapa de almacenamiento, los datos recopilados por los sensores ultrasónicos de los contenedores de residuos infecciosos de los laboratorios clínicos se almacenarán estructurada y organizadamente. Estos datos incluirán las mediciones del nivel de llenado realizados por los sensores, los porcentajes de llenado calculados en función del tamaño de los contenedores y cualquier otra información relevante que pueda ser útil para el posterior análisis y visualización.

Una vez que el ESP32 haya enviado los datos a través de la red WiFi, estos serán recibidos y almacenados en un servidor en la nube. Se evaluarán diferentes opciones de bases de datos, tanto relacionales como no relacionales, para determinar cuál se ajusta mejor a las necesidades específicas del sistema en cuanto a rendimiento, escalabilidad, capacidad de almacenamiento y facilidad de integración con el resto de los componentes.

Algunas de las bases de datos que se analizarán en la tabla 14 son MySQL, PostgreSQL, MongoDB, Cassandra y Firebase, entre otras. Se considerarán factores como la estructura y volumen de datos a almacenar, las operaciones de lectura y escritura requeridas, la capacidad de procesamiento de consultas complejas y la facilidad de implementación y mantenimiento.

La información almacenada en la base de datos seleccionada servirá como fuente principal para la visualización y análisis posterior de los datos en una aplicación móvil. Esta aplicación podrá acceder a los datos mediante APIs o servicios web, permitiendo a los usuarios monitorear en tiempo real el estado de llenado de los contenedores, generar informes y tomar decisiones informadas sobre la gestión y recolección de los residuos infecciosos.

Tabla 14

Tabla comparativa de Bases de Datos

Nombre	MongoDB	Firestore Database	PostgreSQL	MySQL	Cassandra
Figura					
Tipo	NoSQL	NoSQL	SQL	SQL	NoSQL
Modelo de Datos	Documentos (BSON)	Documentos (JSON)	Relacional	Relacional	Columnar
Estructura y Volumen de Datos	No relacional (documentos), ideal para datos semi-estructurados y escalabilidad horizontal	No relacional (documentos), enfocada en desarrollo de aplicaciones en tiempo real	Relacional, soporta grandes volúmenes de datos estructurados	Relacional, soporta grandes volúmenes de datos estructurados	No relacional (columnar), diseñada para alta escalabilidad y disponibilidad
Operación de Lectura y Escritura	Rápida para lectura y escritura de documentos JSON	Rápida para lectura y escritura de datos en tiempo real	Rápida para lectura y escritura de datos relacionales	Rápida para lectura y escritura de datos relacionales	Excelente para lectura y escritura de grandes volúmenes de datos
Facilidad de Implementación y Mantenimiento	Fácil de implementar y escalar, pero puede requerir más esfuerzo de mantenimiento que bases de datos relacionales	Fácil de implementar y mantener, integración nativa con servicios de Google	Implementación y mantenimiento más complejos que MySQL, pero con mayor flexibilidad y características	Fácil de implementar y mantener, amplia documentación y comunidad	Implementación y mantenimiento más complejos, requiere conocimientos específicos de arquitectura distribuida
Rendimiento	Alto	Alto	Alto	Alto	Muy alto

Seguridad	Autenticación, autorización, auditoría y cifrado	Reglas de seguridad basadas en expresiones	Autenticación, autorización, auditoría y cifrado	Autenticación, autorización y cifrado	Autenticación, autorización y cifrado
Costo	Gratuito para uso personal, planes de pago por uso	Gratuito hasta cierto límite, luego pago por uso	Completamente gratuito	Completamente gratuito	Completamente gratuito

Nota. En esta tabla se ha recopilado la información más relevante acerca de las bases de datos más utilizadas, para poder analizar la base de datos que mejor se ajuste a las necesidades del sistema.

Fuente: (Domínguez, 2016) & (Chuquimarca, 2022) & (Castillo L. , 2023)

- **Firestore Realtime Database**

Una vez que se ha analizado rigurosamente las características de las bases de datos presentes en la tabla anterior, Firestore Realtime Database ha sido seleccionada debido a su capacidad de proporcionar actualizaciones en tiempo real de los datos capturados por los sensores de los contenedores, lo que permite una gestión eficiente del proceso de recolección; su alta escalabilidad para manejar una gran cantidad de contenedores ubicados en cada laboratorio clínico; su compatibilidad con el acceso en línea y fuera de línea, útil para ubicaciones remotas; su fácil integración con aplicaciones móviles, facilitando el monitoreo y la emisión de notificaciones hacia los encargados de la recolección de desechos infecciosos; y sus robustas reglas de seguridad y acceso para garantizar la privacidad y protección de los datos, convirtiéndola en la opción más adecuada en comparación con otras bases de datos.

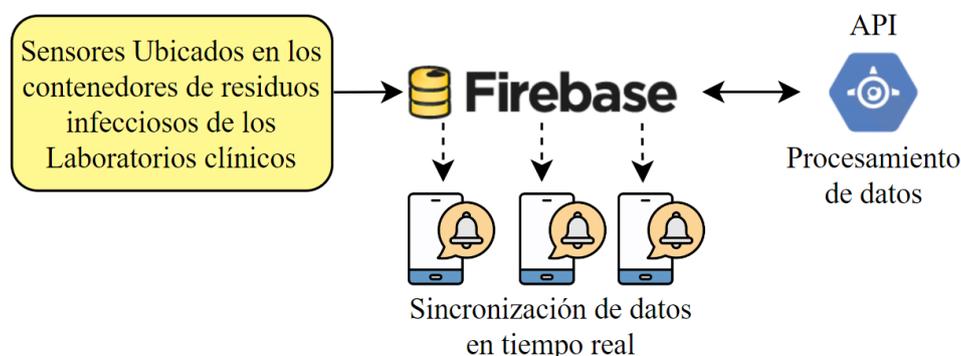
- **Funcionamiento de Firestore Realtime Database**

Firestore Realtime Database es una base de datos NoSQL alojada en la nube, que almacena y sincroniza datos en formato JSON en tiempo real. Utiliza una arquitectura de árbol anidado para organizar los datos, donde cada nodo representa un valor o un objeto

JSON. Cuando se realizan cambios en los datos, ya sea mediante escrituras, actualizaciones o eliminaciones, estos cambios se propagan automáticamente a todos los dispositivos y aplicaciones conectados a la base de datos, garantizando una sincronización instantánea (Baltazar, 2020). Como se observa en la figura 19, Firebase Realtime Database ofrece una API flexible que permite acceder y manipular los datos desde diferentes plataformas, como aplicaciones web, móviles o de servidor. La seguridad es un aspecto clave, con reglas de acceso granulares que permiten controlar quién puede leer y escribir datos en la base de datos. También admite el acceso fuera de línea, almacenando los datos localmente y sincronizándolos cuando se restablece la conexión (Vargas, 2020).

Figura 19

Funcionamiento de Firebase Realtime Database



Fuente: (Baltazar, 2020)

Elaborado por: Calero, 2024

3.3.4. Etapa de Visualización

Con el objetivo de desarrollar la aplicación móvil que permita presentar datos capturados por los sensores y registrados en la base de datos Firebase, se ha realizado un análisis exhaustivo de los diversos softwares disponibles en el mercado para el desarrollo de aplicaciones móviles, comparando detalladamente las características y capacidades de cada uno de ellos, con el fin de seleccionar aquel que mejor se adapte a las condiciones

específicas del sistema y ofrezca una integración óptima con Firebase, la base de datos elegida para este proyecto.

Para el desarrollo de aplicaciones móviles con Firebase, existen varias opciones de IDEs y herramientas. Android Studio y Xcode son las opciones nativas oficiales para Android e iOS, respectivamente, y ofrecen una integración completa con todos los servicios de Firebase (Oliver, 2016). React Native y Flutter permiten desarrollar aplicaciones nativas multiplataforma con JavaScript/React o Dart, y soportan la mayoría de los servicios de Firebase a través de paquetes oficiales (Castillo A. , 2020). Ionic, por otro lado, es un framework para crear aplicaciones híbridas basadas en tecnologías web y permite integrar varios servicios de Firebase (Saavedra González, 2018). Todas estas herramientas pueden aprovechar las herramientas en línea de Firebase, como la Consola y el Emulador, para configurar y probar los servicios. La elección dependerá de factores como el enfoque nativo o híbrido, el rendimiento requerido, la curva de aprendizaje, el soporte multiplataforma y las preferencias de lenguaje de programación del equipo de desarrollo.

A continuación, se muestra un cuadro comparativo (tabla 15) entre los softwares que sirven para el desarrollo de aplicaciones móviles:

Tabla 15

Tabla comparativa de Softwares para el desarrollo de Aplicaciones Móviles

Criterio	Android Studio	Xcode	React Native	Flutter	Ionic
Plataforma objetivo	Android	iOS	Android, iOS	Android, iOS	Android, iOS, Web
Lenguaje de programación	Java, Kotlin	Swift, Objective-C	JavaScript (React)	Dart	TypeScript, JavaScript
Integración de Firebase	Nativa (librerías oficiales)	Nativa (librerías oficiales)	Paquete react-native-firebase	Paquetes oficiales	Paquete @ionic/angular-firebase

Servicios de Firebase soportados	Todos	Todos	La mayoría	La mayoría	Varios
Enfoque de desarrollo	Nativo	Nativo	Nativo	Nativo	Híbrido
Rendimiento	Alto	Alto	Alto	Alto	Moderado
Curva de aprendizaje	Moderada	Moderada	Moderada	Moderada	Baja
Multiplataforma	No	No	Sí	Sí	Sí
Herramientas en línea	Consola de Firebase, Emulador de Firebase				

Nota: el cuadro comparativo proporciona una visión general de las principales características y diferencias entre estas herramientas populares para el desarrollo de aplicaciones móviles que se integran con Firebase, lo que facilita la selección de la opción más adecuada según los requisitos y preferencias del proyecto.

Fuente: (Castillo A. , 2020) & (Saavedra González, 2018) & (Oliver, 2016)

- **Flutter**

Después de una evaluación exhaustiva entre diversas plataformas de desarrollo como Android Studio, Xcode, React Native, Flutter e Ionic, se ha optado por Flutter como la opción más adecuada para este proyecto. Esto se debe a su integración fluida con Firebase, facilitada por ser un framework de código abierto de Google (Miriam, 2021). Flutter permite una construcción eficiente de aplicaciones nativas para iOS y Android a partir de una sola base de código, garantizando un rendimiento óptimo gracias a su motor de renderizado Skia. Su capacidad de hot reload agiliza el ciclo de desarrollo al permitir la visualización en tiempo real de los cambios en el código. Además, cuenta con un sólido respaldo por parte de Google y una comunidad activa que aseguran actualizaciones frecuentes y una amplia gama de recursos disponibles (Castillo A. , 2020).

- **Funcionamiento de Flutter para el desarrollo de aplicaciones móviles**

Flutter es un marco de trabajo de código abierto desarrollado por Google para crear aplicaciones nativas para móviles, web y escritorio. En el caso del desarrollo de aplicaciones móviles para Android, el código fuente escrito en Dart se compila a través del compilador de Android en código nativo ARM o ARM64, dependiendo de la arquitectura del dispositivo objetivo (Ávila, 2022). Este código nativo se empaqueta junto con los recursos de la aplicación, como imágenes y fuentes, dentro de un archivo APK (Android Package Kit). Flutter utiliza su propio motor de renderizado, llamado Skia, para dibujar todos los widgets de la interfaz de usuario directamente en la GPU del dispositivo Android, brindando un alto rendimiento y una experiencia fluida (Miriam, 2021). El motor de Flutter también maneja la entrada táctil, los gestos y el acceso a servicios y sensores del dispositivo Android a través de plugins específicos de la plataforma. Además, Flutter ofrece widgets de Material Design listos para usar, así como la posibilidad de crear widgets personalizados, lo que facilita el desarrollo de interfaces de usuario nativas y atractivas para dispositivos Android (Castillo A. , 2020).

3.4. Diseño del Sistema

En esta sección, abordaremos el diseño del sistema, donde se presentarán el Diagrama de Bloques que brindará una representación visual de los diferentes bloques funcionales y su interconexión, los Requerimientos del Sistema detallando las especificaciones técnicas y características deseadas, la Selección de Tecnologías justificando la elección de las herramientas utilizadas, la Arquitectura describiendo la estructura general y patrones de diseño empleados, y los Diagramas de Interconexión que ilustrarán las conexiones físicas y lógicas entre los componentes del sistema, proporcionando una comprensión sólida del diseño para su posterior implementación.

3.4.1. Diagrama de Bloques

A continuación, en la figura 20, se muestra el diagrama de bloques, en donde se observa como las capas interactúan entre sí para formar un sistema integrado que permite el monitoreo en tiempo real de los contenedores de residuos infecciosos, la generación de notificaciones de recolección y la optimización de las rutas de recolección en los laboratorios clínicos de la ciudad de Ibarra. Para el diseño del diagrama de bloques, se ha tomado como referencia la arquitectura IoT, de modo que a continuación se describe cada una de ellas.

Bloque de Sensado: En este bloque, el sensor ultrasónico se conecta a un microcontrolador, que facilita la conexión a internet para la transmisión de datos. Tanto el sensor como el microcontrolador están alimentados por una batería de 5V, que asegura su funcionamiento autónomo. El microcontrolador recopila los datos del sensor ultrasónico y los transmite al siguiente bloque del sistema.

Bloque de Conectividad: Este bloque incluye dispositivos como routers y gateways que sirven para establecer una conexión WiFi y permitir la comunicación entre los módulos microcontroladores y la capa de análisis y procesado. Los datos recopilados por los sensores ultrasónicos y procesados por los microcontroladores se transmiten a través de este bloque utilizando el protocolo de comunicación inalámbrica WiFi.

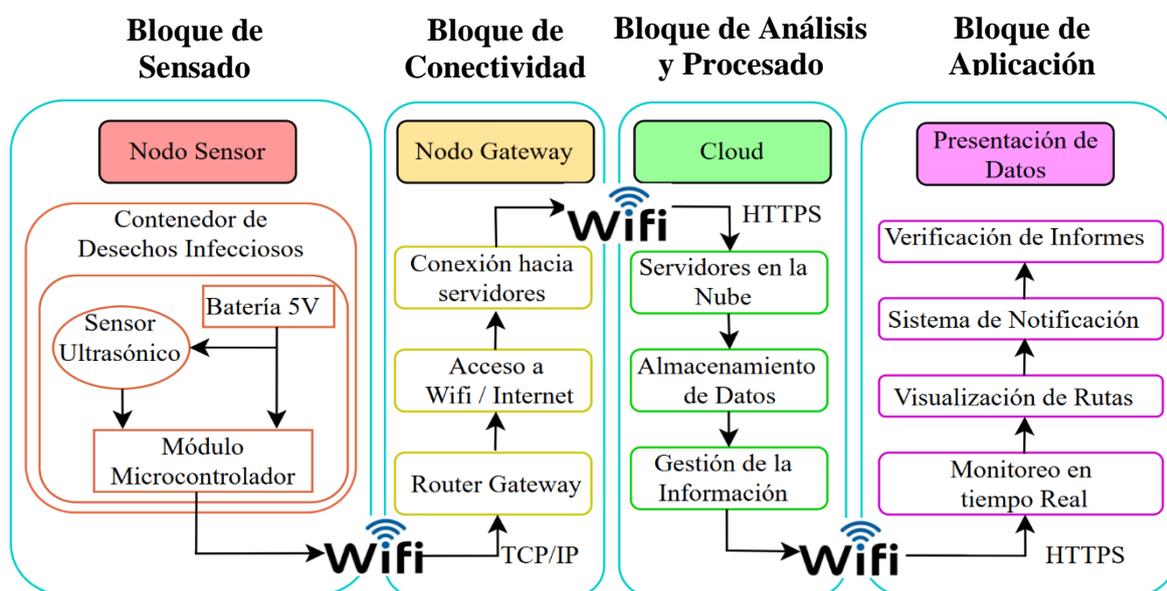
Bloque de Análisis y Procesado: En este bloque se encuentra una base de datos donde se almacenan los datos recibidos de la capa de conectividad. Además, contiene módulos de procesamiento que analizan estos datos para determinar cuándo un contenedor necesita ser recolectado, teniendo en cuenta factores como el nivel de llenado, la capacidad del contenedor y las políticas de recolección establecidas.

Bloque de Aplicación: Este bloque consta de una aplicación móvil que presenta al usuario final los datos relevantes obtenidos por el bloque de sensado. Dependiendo del

tipo de usuario que haya iniciado sesión (recolector, laboratorio clínico o administrador), podrá realizar diferentes actividades, como la verificación del nivel de llenado de los contenedores, el envío de notificaciones, la verificación de la ruta más corta hacia el laboratorio o la consulta de informes con los datos recopilados.

Figura 20

Diagrama de Bloques del Sistema



Elaborado por: La Autora

3.4.2. Selección de la Tecnología de Comunicación

Dado que el proyecto se desarrollará en el centro de la ciudad, se optó por una tecnología inalámbrica que facilite la implementación y operación del sistema. Para la elección de la tecnología más adecuada, se consideraron factores clave como el alcance, el consumo energético, la velocidad de transmisión y la seguridad.

Cada tecnología tiene ventajas específicas dependiendo de la aplicación y el entorno. Por ejemplo, Wi-Fi es ideal para áreas con alta densidad de dispositivos y necesidades de alta velocidad, pero su alto consumo energético puede ser un inconveniente para proyectos IoT de larga duración (Muñoz, 2019). Por otro lado, LoRa y Sigfox son opciones de bajo consumo energético y largo alcance, lo que las hace aptas

para aplicaciones IoT en áreas rurales o para dispositivos que deben funcionar durante largos períodos sin necesidad de recarga. Bluetooth y ZigBee son opciones viables para comunicaciones de corto alcance con bajo consumo, especialmente en entornos con dispositivos cercanos (Sosa, 2020).

La tabla comparativa (Tabla 16) proporciona un resumen de las características técnicas de cada tecnología, lo que facilita la visualización de sus ventajas y limitaciones para este proyecto.

Tabla 16

Tabla comparativa de las Tecnologías de Comunicación Inalámbrica

Tecnología	Wi-Fi	LoRa	Bluetooth	ZigBee	Sigfox
Estándar	IEEE 802.11	Patentado por Semtech	IEEE 802.15.1	IEEE 802.15.4	Patentado por Sigfox
Frecuencia	2.4 GHz, 5 GHz	Sub-GHz ISM	2.4 GHz	2.4 GHz	Sub-GHz ISM (868/902 MHz)
Rango	45m (indoor), 90m (outdoor)	5-15 km (urbano), 20+ km (rural)	10-100m	10-100m	30-50 km (urbano), 50+ km (rural)
Velocidad	11-6900 Mbps	0.3-50 Kbps	1-24 Mbps	20-250 Kbps	100 bps (uplink), 600 bps (downlink)
Topología red	Punto a multipunto	Punto a multipunto	Punto a punto	Malla, estrella, árbol	Estrella
Seguridad	WEP, WPA, WPA2	AES-128	E0 stream cipher	AES-128	Secuencia aleatoria (ECC)
Principales usos	WLAN, IoT	LPWAN, IoT	Periféricos, IoT	Automatización, IoT	IoT (smart cities, agricultura)

Nota. En este cuadro comparativo, se consideraron aspectos fundamentales como el rango de cobertura, la tasa de transferencia de datos, el consumo energético, las capacidades de seguridad, la facilidad de implementación y los costos asociados a cada tecnología.

Fuente: (Sosa, 2020) & (Chávez, 2020) & (Cajas C. , 2023) & (Muñoz, 2019)

- **Tecnología de Comunicación Wi-Fi**

Después de analizar detenidamente cada tecnología en el cuadro comparativo, considerando los factores mencionados y las necesidades específicas del proyecto, se determinó que Wi-Fi es la opción más adecuada porque permite la conexión inalámbrica, lo cual es fundamental en este caso ya que los laboratorios clínicos se encuentran distribuidos en toda la ciudad de Ibarra. Al no requerir cables, Wi-Fi facilita enormemente la instalación de los sensores en cada contenedor. Además, WiFi tiene un rango de alcance adecuado para cubrir el área de los laboratorios y sus alrededores, permitiendo que los datos de los sensores se transmitan sin problemas desde cualquier contenedor al sistema central de monitoreo. Otra ventaja clave es que, esta tecnología es compatible con la mayoría de los dispositivos modernos como teléfonos móviles, tabletas y computadoras. Esto facilita el envío de notificaciones y la visualización de rutas de recolección en dispositivos portátiles para el personal encargado.

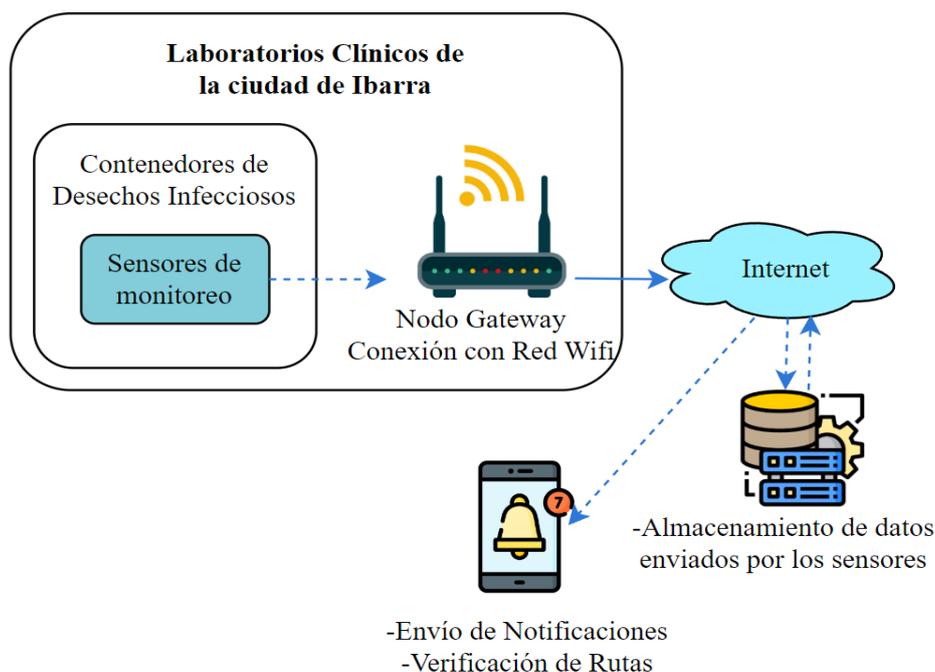
- **Funcionamiento de la Tecnología de Comunicación Wi-Fi**

WiFi es una tecnología de red inalámbrica que utiliza ondas de radio para transmitir datos.

Funciona mediante un punto de acceso (router WiFi), que emite una señal de radio cubriendo un área determinada, permitiendo que dispositivos como laptops, teléfonos o tabletas se conecten (Cajas K. , 2022). Estos dispositivos intercambian datos con el punto de acceso usando protocolos WiFi, el cual está vinculado a una red cableada (como Internet) para enrutar el tráfico como se observa en la figura 21. WiFi usa técnicas de codificación, modulación de señales y acceso múltiple para permitir que varios dispositivos compartan el mismo canal de manera eficiente (Salazar K. , 2018).

Figura 21

Tecnología de Comunicación Wifi



Fuente: (Cajas C. , 2023)

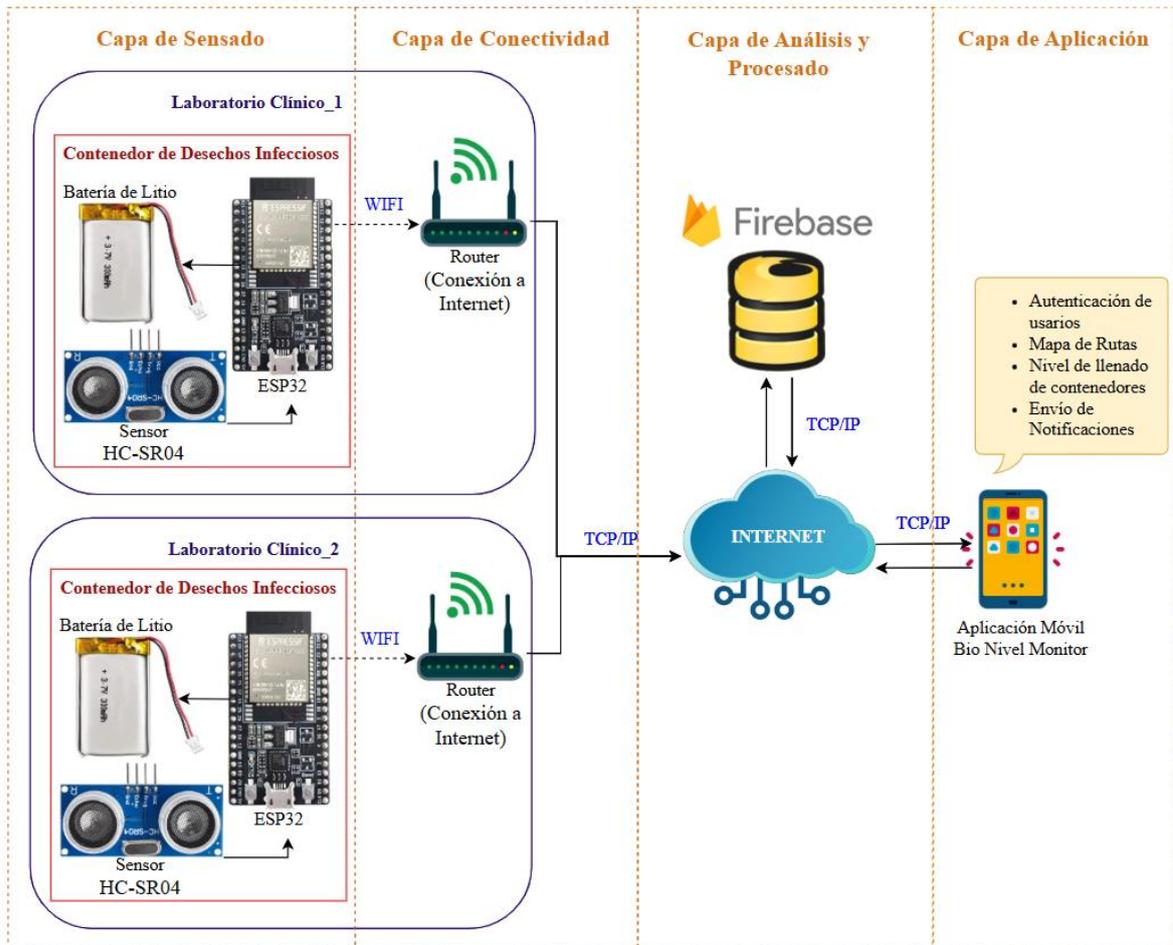
Elaborado por: La Autora

3.4.3. Arquitectura del Sistema

En la figura 22, se muestra la arquitectura del sistema, la cual se encuentra representada entre las capas de la arquitectura IoT. En la capa de sensado, se cuenta con el nodo sensor (ubicado en la tapa de los contenedores) conformado por un sensor HC-SR04 y una placa ESP32, alimentadas por una batería de litio, los cuales miden el nivel de llenado. El ESP32 se conecta a la red WiFi del laboratorio correspondiente, transmitiendo los datos a través de la capa de conectividad a routers con acceso a Internet. En la capa de análisis y procesado, los datos se envían a la base de datos Firebase, la cual almacena y procesa la información en tiempo real. Finalmente, en la capa de aplicación, la aplicación móvil permite a los usuarios autenticarse, visualizar rutas, monitorear el llenado de los contenedores y recibir notificaciones, facilitando la gestión eficiente de los desechos desde cualquier dispositivo móvil.

Figura 22

Arquitectura del Sistema



Elaborado por: La Autora

3.4.4. Diagramas de Interconexión

Una vez realizado el análisis de los requerimientos del sistema, en donde se seleccionó los componentes de hardware y software, se procede a establecer el diagrama de interconexión del nodo sensor, en donde se observa cada componente con su respectiva conexión.

3.4.4.1. Nodo Sensor

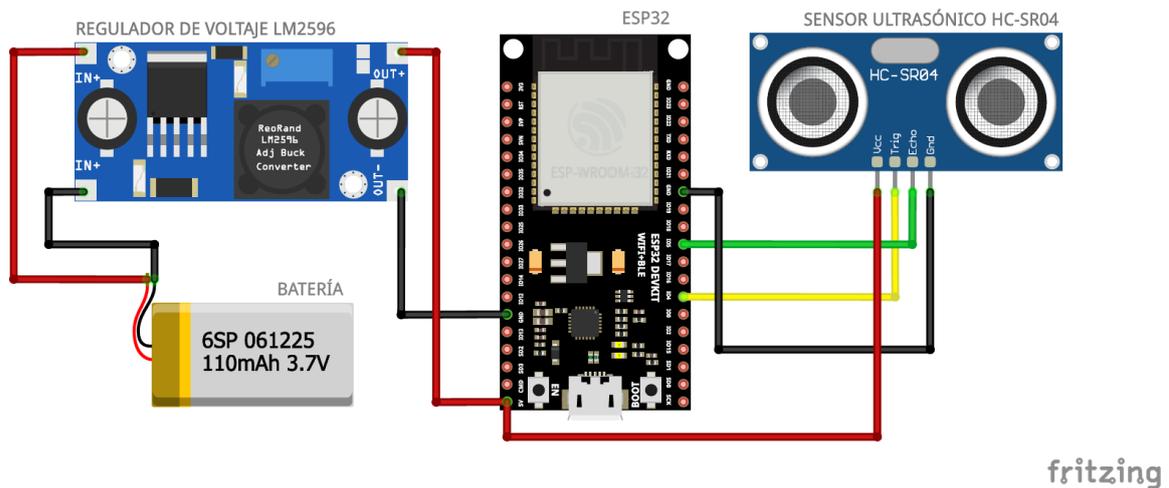
El nodo sensor, elemento que realizará la función de medición, toma y envío de datos, estará compuesto por diferentes elementos electrónicos, los cuales permitirán la correcta

funcionalidad para el sistema, en este caso se visualiza los componentes respectivos y su respectivo diagrama de conexión.

A continuación, en la figura 23 se observa las respectivas conexiones realizadas entre los diferentes elementos:

Figura 23

Diagrama de interconexión del nodo sensor



Elaborado por: La Autora

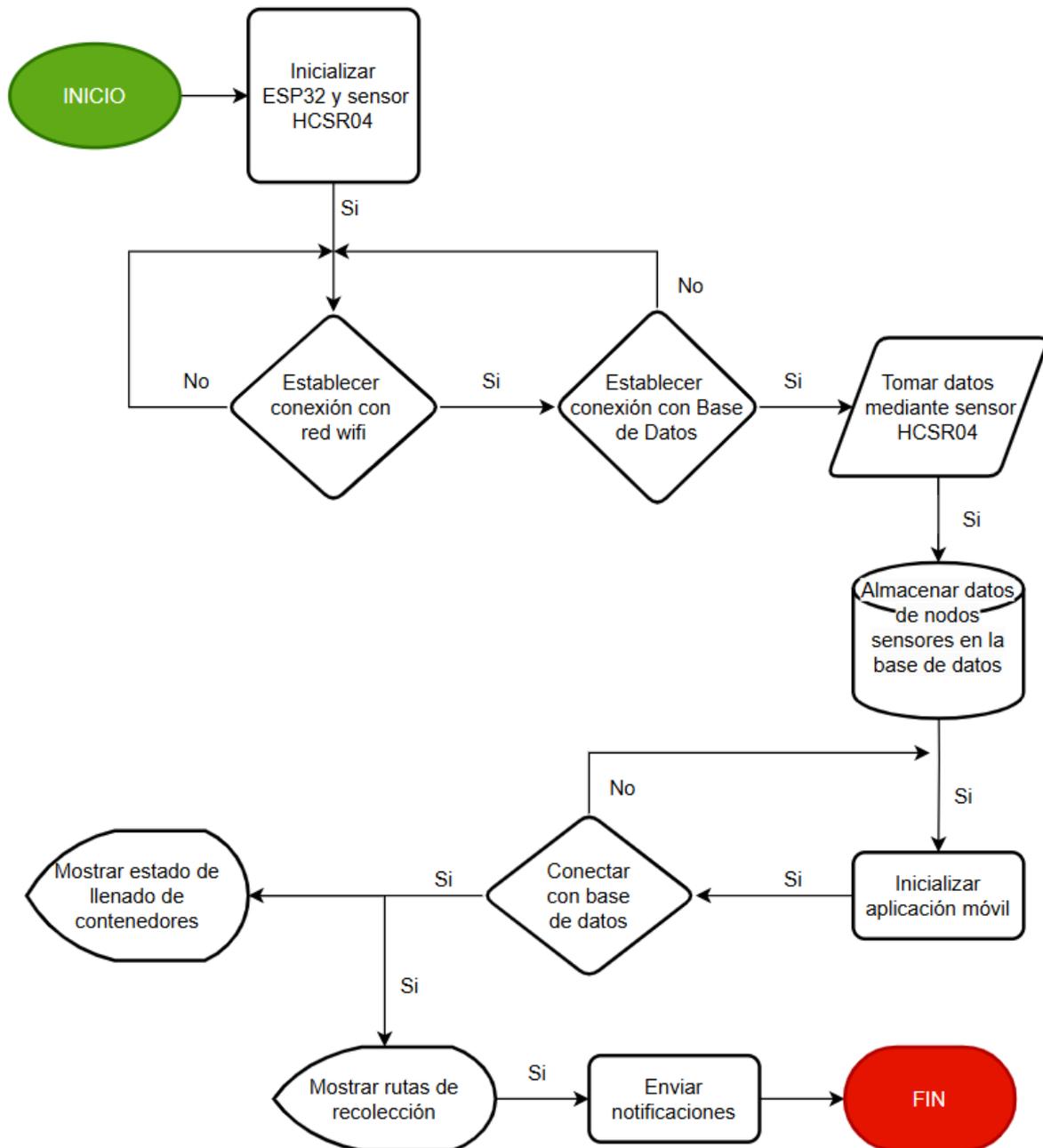
- El pin TRIGGER del sensor HC-SR04 se conecta al pin IO4 del ESP32.
- El pin ECHO del sensor HC-SR04 se conecta al pin IO5 del ESP32.
- El pin VCC del sensor HC-SR04 se conecta al pin 5V del ESP32.
- El pin GND se del sensor HC-SR04 se conecta al pin GND del ESP32.
- El cable positivo de la BATERÍA se conecta al pin IN+ del módulo LM2596.
- El cable negativo de la BATERÍA se conecta al pin IN- del módulo LM2596.
- El pin OUT+ se conecta al pin 5V del ESP32.
- El pin OUT- se conecta al pin GND del ESP32.

3.5. Desarrollo del Software

A continuación, se evidencia el desarrollo del sistema, en el cual se evidencian las configuraciones tanto de hardware como de software realizadas en cada componente, para cumplir con la funcionalidad planteada de cada elemento que conforma el proyecto.

3.5.1. Diagrama de Flujo del Sistema

El diagrama de flujo mostrado en la figura 24 describe el proceso general de monitoreo de contenedores de desechos infecciosos ubicados en los Laboratorios Clínicos de la ciudad de Ibarra. Se comienza con la inicialización del ESP32 y el sensor HC-SR04. Luego, se verifica si se ha establecido la conexión con la red WiFi; si no, se vuelve a intentar. Una vez conectados a la red, se establece la conexión con la base de datos y se procede a la toma de datos mediante el sensor HC-SR04, almacenándolos en la base de datos. Paralelamente, se inicializa la aplicación móvil que también se conecta con la base de datos para acceder a la información. La aplicación muestra el estado de llenado de los contenedores, rutas de recolección y envía notificaciones a los usuarios.

Figura 24*Diagrama de Flujo del Sistema**Elaborado por: La Autora*

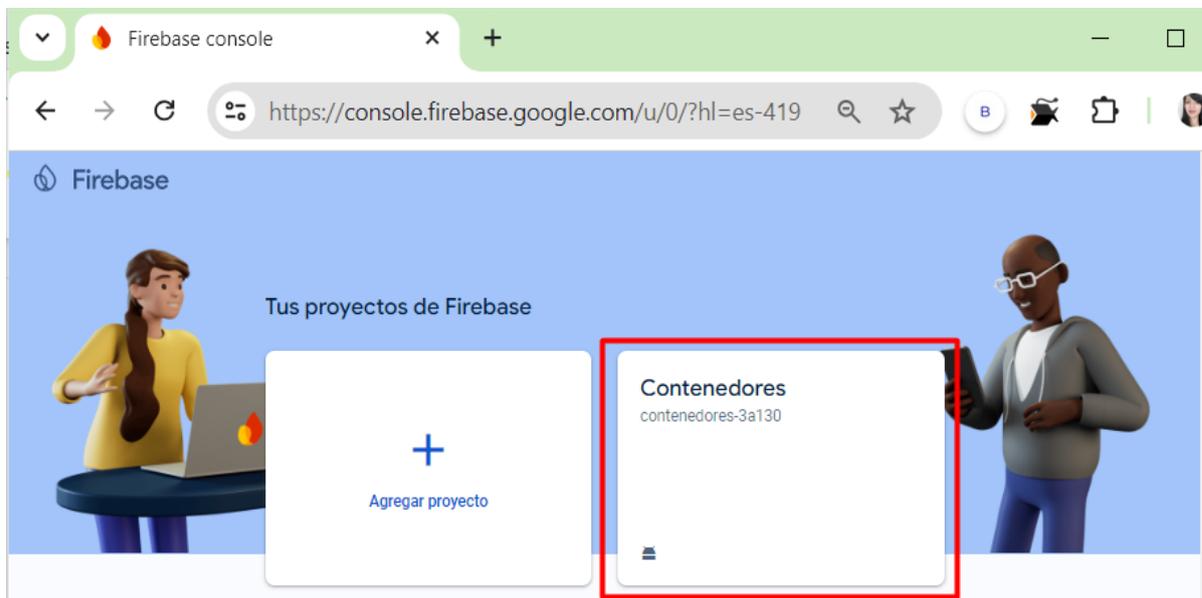
3.5.2. Configuración de la Base de Datos

Los datos recopilados por los nodos sensores y la información de los usuarios autenticados en la aplicación móvil se almacenan en Firebase, una plataforma de base de datos en la nube. Para implementar esto, se crea un nuevo proyecto accediendo a la

consola de Firebase ingresando al siguiente link: <https://console.firebase.google.com>. La base de datos resultante se ha registrado con el nombre "Contenedores", como se muestra en la figura 25.

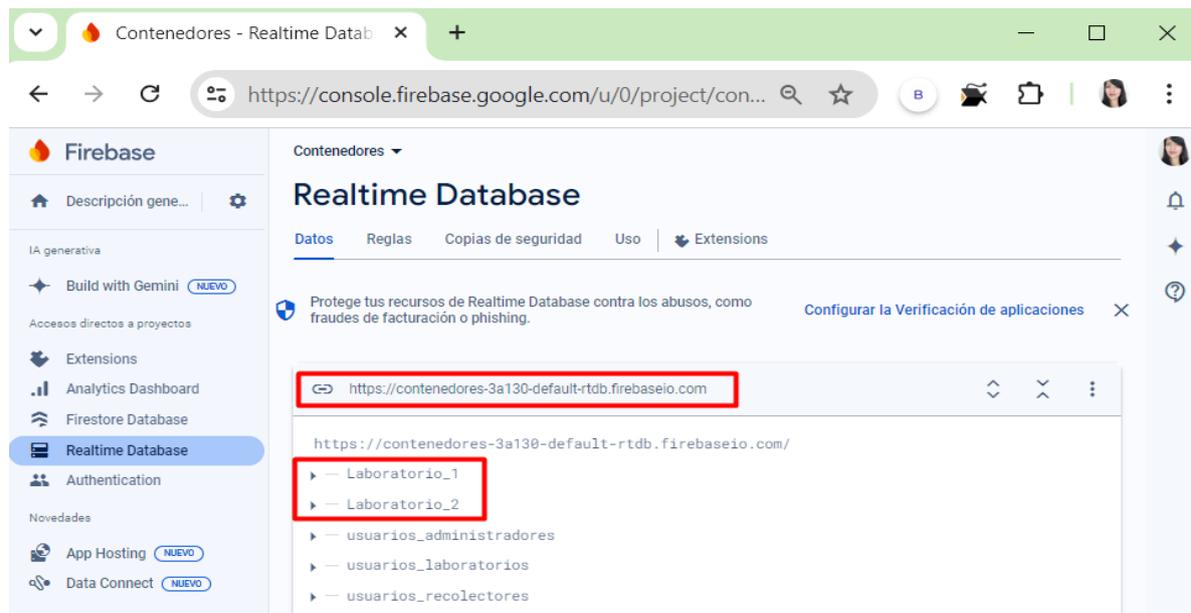
Figura 25

Creación de nuevo Proyecto



Elaborado por: La Autora

Después de crear exitosamente el nuevo proyecto, se accede a la página principal de Firebase. Allí, se selecciona la opción "Realtime Database", como se ilustra en la figura 26. Para establecer la conexión entre la base de datos y el código programado en el ESP32, es necesario obtener dos elementos clave: el host y el código de autenticación. Estos datos se pueden encontrar en la sección de configuraciones de la base de datos. Esta información es esencial para que el ESP32 pueda comunicarse correctamente con Firebase y almacenar o recuperar datos.

Figura 26*Creación de Base de Datos*

Elaborado por: La Autora

En la figura 27 se muestra la configuración del SDK (Software Development Kit) de Firebase para una aplicación Android llamada "Sistema de Monitoreo - Desechos Infecciosos". Se está preparando la conexión entre Firebase y la aplicación móvil. Los elementos clave que se observan son: la opción para descargar el archivo "google-services.json", que contiene las credenciales y configuraciones necesarias para que la app se conecte a Firebase, y las huellas digitales del certificado SHA-1, que son esenciales para la autenticación y seguridad de la conexión. Estos pasos son cruciales para establecer una comunicación segura entre la aplicación móvil y los servicios de Firebase, permitiendo funcionalidades como la autenticación de usuarios y el acceso a la base de datos en tiempo real.

Figura 27

Configuración del SDK

Tus apps

Agregar app

Apps para Android

Sistema de Monitoreo - Desechos ...
com.ContenedoresInfecciosos.Desechos

Configuración del SDK

¿Necesitas volver a configurar los SDK de Firebase en tu app? Revisa las instrucciones de configuración del SDK o descarga el archivo de configuración con las claves y los identificadores de tu app.

[Ver las instrucciones del SDK](#) [google-services.json](#)

ID de la app ⓘ
1:790433505996:android:dd4a103cba20d4e2e01869

Sobrenombre de la app
Sistema de Monitoreo - Desechos Infecciosos ✎

Nombre del paquete
com.ContenedoresInfecciosos.Desechos

Huellas digitales del certificado SHA ⓘ	Tipo ⓘ
88:62:50:d8:af:e4:e5:41:31:ff:19:58:04:63:d3:2b:d9:43:db:3b	SHA-1

[Agregar huella digital](#)

Elaborado por: La Autora

Las librerías que se muestran en la figura 28 permiten conectar y configurar Firebase en la aplicación de Flutter. La primera, `firebase_options.dart`, contiene las opciones de configuración específicas para vincular la app con el proyecto en Firebase, y se genera automáticamente durante la configuración inicial de Firebase; la segunda, `firebase_core.dart`, es fundamental para iniciar los servicios de Firebase en la aplicación, permitiendo que todas las funcionalidades de Firebase se integren correctamente; y finalmente, `firebase_database.dart` proporciona acceso a la base de datos en tiempo real de Firebase, lo que permite leer y escribir datos de manera sincronizada.

Figura 28*Bibliotecas de conexión con Firebase*

```
import 'package:bio_nivel_monitor/firebase_options.dart';
import 'package:firebase_core/firebase_core.dart';
import 'package:firebase_database/firebase_database.dart';
```

Elaborado por: La Autora

El código presente en la figura 29, es crucial para iniciar una aplicación de Flutter con Firebase. La primera línea, `WidgetsFlutterBinding.ensureInitialized()`, garantiza que Flutter esté completamente listo antes de cualquier operación. Luego, `Firebase.initializeApp()` conecta la app a Firebase usando las configuraciones específicas de `DefaultFirebaseOptions.currentPlatform`, necesarias para la autenticación. Finalmente, `runApp(const MyApp())` lanza la aplicación principal, ya conectada a Firebase.

Figura 29*Conexión con Firebase (Base de Datos)*

```
Run | Debug | Profile
void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp(
    options: DefaultFirebaseOptions.currentPlatform,
  );
  runApp(app: const MyApp());
}
```

*Elaborado por: La Autora***3.5.3. Configuración del Nodo Sensor**

El nodo sensor tiene como objetivo medir el nivel de llenado de los contenedores y enviar estos datos a una base de datos para su almacenamiento. Para programar estas funcionalidades, se utiliza el entorno de desarrollo integrado (IDE) de Arduino. En este IDE se desarrolla el código que permite al nodo sensor ejecutar las funciones de medición y transmisión de datos de manera eficiente.

3.5.3.1. Programación ESP32

La programación del nodo sensor comienza con la configuración del microcontrolador ESP32. El primer paso es incluir las librerías necesarias para su funcionamiento (figura 30). Se utiliza la librería WiFi, que permite utilizar la capacidad de conexión inalámbrica del dispositivo, y la librería Firebase, que facilita el envío de datos a la base de datos Firebase previamente configurada. Estas librerías son fundamentales para que el nodo sensor pueda conectarse a la red y transmitir los datos recolectados.

Adicionalmente, se observa cómo se inicializa las variables con el nombre de la red y la contraseña para que el dispositivo se conecte a internet, estas variables serán posteriormente utilizadas en las funciones del código de programación.

Figura 30

Importación de Librerías Wifi y credenciales de conexión

```
NodoSensor.ino
1 //Inclusión de librerías
2 #include <WiFi.h> // Conexión Wi-Fi
3 #include <FirebaseESP32.h> // Interacción con Firebase
4 #include <Preferences.h> // Almacenamiento no volátil
5 #include <time.h> // Manejo de fechas y horas
6
7 // Credenciales de conexión a la red Wi-Fi
8 const char* ssid = "Melannie"; //Nombre red Wi-Fi
9 const char* password = "12345678"; //Contraseña red Wi-Fi
```

Elaborado por: La Autora

El siguiente paso en la programación es la declaración de las variables de autenticación necesarias para establecer una conexión segura con el proyecto creado en Firebase, como se observa en la figura 31. Estas variables incluyen la URL del proyecto y la clave de autenticación. Además, se definen las variables de configuración para la conexión con Firebase, que abarcan tanto los parámetros de autenticación como los de configuración general. Estos elementos son esenciales para garantizar que el nodo sensor

pueda comunicarse de manera segura con la base de datos Firebase, donde se almacenarán los datos recolectados por el sensor.

Figura 31

Conexión con Base de Datos

```
11 //Datos de conexión a Firebase: URL de la base de datos y Token de autenticación
12 #define FIREBASE_HOST "https://contenedores-3a130-default-rtdb.firebaseio.com"
13 #define FIREBASE_AUTH "HehHAgRp9o0bbpjyG05hNheujLjmSj21cYBg5Kq7"
14
15 FirebaseData fbdo; // Objeto para manejar datos de Firebase
16 FirebaseAuth auth; // Objeto para la autenticación en Firebase
17 FirebaseConfig config; // Objeto de configuración de Firebase
```

Elaborado por: La Autora

Se configuran los pines trigPin y echoPin para el sensor ultrasónico, junto con la variable tamanoContenedor, que almacena el tamaño del contenedor solo la primera vez que se utiliza el nodo sensor. El tamaño medido se guarda en la memoria no volátil utilizando el objeto Preferences, que permite conservar los datos incluso si el dispositivo se apaga o reinicia. También se incluyen variables para controlar las lecturas del sensor: “ultimaLectura” almacena el tiempo de la última lectura, mientras que “intervalo” define el tiempo mínimo necesario para establecer la conexión con el Wi-Fi y la base de datos. Además, se declaran constantes para identificar el laboratorio y el contenedor en la base de datos. Finalmente, se define la estructura DatosTemporales, que permite almacenar de forma temporal datos como la marca de tiempo, la distancia medida y el porcentaje de llenado del contenedor, como se observa en la figura 32.

Figura 32*Configuración de Sensor Ultrasónico HC-SR04*

```

19 // Pines del sensor ultrasónico
20 const int trigPin = 4; // Pin de disparo (Trigger)
21 const int echoPin = 5; // Pin de recepción (Echo)
22
23 // La primera vez que se utiliza el nodo sensor se mide el tamaño del contenedor
24 float tamanoContenedor = 0.0;
25
26 Preferences preferences;
27
28 unsigned long ultimaLectura = 0; // Tiempo de la última lectura
29 const unsigned long intervalo = 5000; // Intervalo de 5 segundos entre lecturas
30
31 const String nombreLaboratorio = "Laboratorio_1"; // Nombre del laboratorio
32 const String nombreContenedor = "Contenedor_1"; // Nombre del contenedor
33
34 // Variables para almacenamiento temporal
35 struct DatosTemporales {
36     String marcaTiempo;
37     float distancia;
38     int porcentaje;
39 };
--

```

Elaborado por: La Autora

En la figura 33, se visualiza la función encargada de establecer la conexión con la red Wi-Fi. En esta función, se inicia el proceso de conexión utilizando el SSID y la contraseña configurados previamente. La conexión intenta establecerse hasta un máximo de 20 veces, verificando cada intento en intervalos de 500 milisegundos. Si la conexión es exitosa, se imprime un mensaje confirmando la conexión; de lo contrario, se notifica que no se pudo conectar a la red Wi-Fi.

Figura 33

Conexión con la red Wi-Fi

```

43 //Conexión con red Wi-Fi
44 void conectarWiFi() {
45     WiFi.begin(ssid, password);//Inicialización de red Wi-Fi
46     int intentos = 0;
47     while (WiFi.status() != WL_CONNECTED && intentos < 20) { //Se intenta conectar hasta 20 veces
48         delay(500);
49         Serial.println("Conectando a WiFi...");
50         intentos++;
51     }
52
53     if (WiFi.status() == WL_CONNECTED) {
54         Serial.println("Conectado a WiFi");//Mensaje de conexión exitosa
55     } else {
56         Serial.println("No se pudo conectar a WiFi");//Mensaje de conexión inválida
57     }
58 }

```

Elaborado por: La Autora

En la figura 34, se visualiza la configuración de la conexión con Firebase y la sincronización con el servidor NTP para obtener la hora. Primero, se configura Firebase asignando el host y el token de autenticación necesarios para establecer la comunicación. La función `Firebase.begin` inicializa la conexión y se verifica su estado mediante `Firebase.ready()`, imprimiendo un mensaje si la conexión es exitosa o notificando un error en caso contrario. Posteriormente, se configura el servidor NTP (Network Time Protocol) para sincronizar la hora del dispositivo. Esto se realiza utilizando la función `configTime`, especificando el huso horario y el servidor NTP. Un bucle se ejecuta hasta que se obtiene la hora correcta, mostrando mensajes en cada intento de sincronización y confirmando cuando la hora ha sido sincronizada exitosamente.

Figura 34

Conexión con la base de datos Firebase y servidor NTP

```
71 // Configuración de Firebase
72 config.host = FIREBASE_HOST;
73 config.signer.tokens.legacy_token = FIREBASE_AUTH;
74 Firebase.begin(&config, &auth);
75
76 if (Firebase.ready()) {
77   Serial.println("Conexión con Firebase activa.");
78 } else {
79   Serial.println("Error en la conexión con Firebase.");
80 }
81
82 // Configuración del servidor NTP para obtener la hora
83 configTime(-5 * 3600, 0, "pool.ntp.org");
84 while (!time(nullptr)) {
85   delay(100);
86   Serial.println("Sincronizando con el servidor NTP...");
87 }
88 Serial.println("Hora sincronizada con NTP");
```

Elaborado por: La Autora

Se inicializa la memoria no volátil, que permite guardar datos importantes que permanecen disponibles incluso si el dispositivo se reinicia. Luego, se verifica si el tamaño del contenedor ya está almacenado. Si no lo está, el sensor HC-SR04 mide el tamaño del contenedor por primera vez, un proceso que tarda aproximadamente 5 minutos. Una vez medido, si el resultado es válido, se guarda en la memoria para futuros usos; de lo contrario, se imprime un mensaje de error. Si el tamaño ya estaba almacenado, se recupera directamente de la memoria y se muestra en la consola el tamaño total del contenedor en centímetros, como se observa en la figura 35.

Figura 35*Cálculo del tamaño del contenedor*

```

93     // Inicialización de memoria no volátil
94     preferences.begin("contenedor", false);
95
96     // Recuperar tamaño del contenedor o medirlo si no está almacenado
97     tamanoContenedor = preferences.getFloat("tamano", 0.0);
98     if (tamanoContenedor == 0.0) { // Si no está guardado, se mide por primera vez
99         Serial.println("Midiendo tamaño del contenedor por primera vez...");
100        Serial.println("Esto tomará 5 minutos. Por favor espere...");
101        tamanoContenedor = medirTamanoContenedor();
102        if (tamanoContenedor > 0) { // Si la medición es válida, se almacena
103            preferences.putFloat("tamano", tamanoContenedor);
104            Serial.println("Tamaño del contenedor almacenado en memoria.");
105        } else {
106            Serial.println("Error: No se pudo medir el tamaño del contenedor.");
107        }
108    } else {
109        Serial.println("Tamaño del contenedor recuperado de la memoria.");
110    }
111    // Muestra el tamaño total del contenedor
112    Serial.print("Tamaño total del contenedor (cm): ");
113    Serial.println(tamanoContenedor);
114 }

```

Elaborado por: La Autora

En la figura 36, se muestra la función encargada de calcular el promedio del nivel de llenado del contenedor. Se realizan 15 mediciones consecutivas, obteniendo la distancia bruta desde el sensor. Si alguna medición es inválida (valor no positivo), se ignora y se continúa con la siguiente. Para cada medición válida, se calcula el nivel de llenado como la diferencia entre el tamaño total del contenedor y la distancia medida. Solo se consideran los valores positivos para calcular el valor promedio de la distancia la cual será enviada a la base de datos. Entre cada medición, se espera un intervalo de 4 segundos. Al finalizar, si hubo mediciones válidas, se calcula y retorna el promedio. En caso contrario, se retorna un valor de -1 indicando que no hubo datos válidos, y se muestra un mensaje de error en la consola.

Figura 36*Medición de nivel de llenado con Sensor Ultrasónico HC-SR04*

```

219 float calcularDistanciaPromedio() {
220     float sumaDistancias = 0.0;
221     int medicionesValidas = 0;
222     Serial.println("Tomando 15 mediciones (calculando nivel de llenado):");
223     for (int i = 0; i < 15; i++) {
224         // Leer distancia bruta desde el sensor
225         float distanciaMedida = leerDistanciaBruta();
226         if (distanciaMedida <= 0) {
227             // Si la medición no es válida, saltarla
228             Serial.print("Medición ");
229             Serial.print(i + 1);
230             Serial.println(": Valor inválido (ignorado).");
231             continue;
232         }
233         // Calcular nivel de llenado
234         float nivelllenado = tamanoContenedor - distanciaMedida;
235         Serial.print("Medición ");
236         Serial.print(i + 1);
237         Serial.print(": Nivel de llenado calculado: ");
238         Serial.println(nivelllenado);
239         // Considerar solo valores positivos para el promedio
240         if (nivelllenado > 0) {
241             sumaDistancias += nivelllenado;
242             medicionesValidas++;
243         }
244         delay(1000); // Esperar 4 segundos entre mediciones
245     }
246     // Calcular promedio si hay mediciones válidas
247     if (medicionesValidas > 0) {
248         float promedio = sumaDistancias / medicionesValidas;
249         Serial.print("Promedio de nivel de llenado: ");
250         Serial.println(promedio);
251         return promedio;
252     } else {
253         Serial.println("No hay datos positivos para calcular el promedio.");
254         return -1; // Retorna -1 si no hubo datos válidos
255     }
256 }
---
```

Elaborado por: La Autora

En la figura 37, se muestra la función que calcula el porcentaje de llenado del contenedor. Primero, verifica si el tamaño del contenedor es válido; si no lo es, retorna 0 y muestra un mensaje de error. Luego, divide el tamaño total del contenedor en segmentos del 10% y determina a cuál corresponde el nivel medido usando un bucle. Finalmente, ajusta el porcentaje para que esté entre 0 y 100 y lo retorna.

Figura 37

Cálculo del porcentaje de llenado con Sensor Ultrasónico HC-SR04

```

259 int calcularPorcentajeLlenado(float promedioLlenado) {
260     if (tamanoContenedor <= 0) { // Verifica si el tamaño del contenedor es válido
261         Serial.println("Error: Tamaño del contenedor no válido.");
262         return 0; // Retorna 0 si el tamaño no es válido
263     }
264     // Calcular el tamaño de cada segmento del 10%
265     float segmento = tamanoContenedor / 10.0;
266
267     // Determinar el porcentaje de llenado basado en el promedio
268     int porcentajeLlenado = 0;
269     for (int i = 1; i <= 10; i++) {
270         if (promedioLlenado <= segmento * i) {
271             porcentajeLlenado = i * 10; // Asigna el porcentaje correspondiente
272             break; // Detiene el bucle al encontrar el porcentaje
273         }
274     }
275     // Asegurar que el porcentaje esté dentro del rango 0-100
276     porcentajeLlenado = max(0, min(100, porcentajeLlenado));
277     return porcentajeLlenado; // Retorna el porcentaje de llenado
278 }

```

Elaborado por: La Autora

La función se encarga de guardar datos temporalmente en el ESP32. Primero, obtiene la hora actual y la convierte a un formato legible (YYYY-MM-DD HH:MM:SS). Luego, crea un registro con la marca de tiempo, la distancia promedio medida y el porcentaje de llenado calculado. Este registro se agrega a un buffer de datos que almacena la información mientras se espera su envío a Firebase. Finalmente, imprime un mensaje para confirmar que los datos han sido almacenados como se observa en la figura 38.

Figura 38

Datos guardados temporalmente

```

144 void guardarDatosTemporalmente(float distanciaPromedio, int porcentajeLlenado) {
145     time_t now = time(nullptr); // Obtiene el tiempo actual
146     struct tm* timeinfo = localtime(&now); // Convierte el tiempo a formato local
147     char buffer[26];
148     strftime(buffer, sizeof(buffer), "%Y-%m-%d %H:%M:%S", timeinfo);
149     String marcaTiempo = String(buffer); // Convierte el formato a una cadena de texto
150     // Crea un registro con la marca de tiempo, distancia y porcentaje
151     DatosTemporales dato = {marcaTiempo, distanciaPromedio, porcentajeLlenado};
152     bufferDatos.push_back(dato); // Agrega el registro al buffer de datos
153     Serial.println("Datos almacenados temporalmente en el ESP32."); // Mensaje de confirmación
154 }

```

Elaborado por: La Autora

En la figura 39 se observa la función encargada de enviar datos almacenados en el buffer a Firebase. Primero, verifica si hay conexión Wi-Fi y si Firebase está listo. Si ambas condiciones se cumplen, recorre el buffer enviando cada registro de distancia y porcentaje al nodo correspondiente en Firebase, eliminando del buffer los datos que se envían con éxito. En caso de error durante el envío, muestra un mensaje indicando la razón del fallo y detiene el proceso. Si no hay conexión Wi-Fi, imprime un mensaje informando que los datos permanecerán en el ESP32 hasta que se restablezca la conexión.

Figura 39

Envío de datos a Firebase

```

156 void enviarDatosFirebase() {
157     if (WiFi.status() == WL_CONNECTED && Firebase.ready()) { // Verifica conexión Wi-Fi y Firebase
158         for (auto it = bufferDatos.begin(); it != bufferDatos.end(); ) { // Recorre los datos en el buffer
159             String nodoLaboratorio = nombreLaboratorio; // Nodo de Firebase para el laboratorio
160             String nodoContenedor = nombreContenedor; // Nodo de Firebase para el contenedor
161
162             // Envía distancia y porcentaje a Firebase
163             if (Firebase.setFloat(fbdo, "/" + nodoLaboratorio + "/" + nodoContenedor + "/distancias/" + it->marcaTiempo, it->distancia) &&
164                 Firebase.setInt(fbdo, "/" + nodoLaboratorio + "/" + nodoContenedor + "/porcentajes/" + it->marcaTiempo, it->porcentaje)) {
165                 Serial.println("Datos enviados a Firebase: " + it->marcaTiempo); // Confirmación de envío exitoso
166                 it = bufferDatos.erase(it); // Elimina el dato enviado del buffer
167             } else {
168                 Serial.println("Error al enviar datos: " + fbdo.errorReason()); // Muestra error si falla el envío
169                 break; // Sale del bucle en caso de error
170             }
171         }
172     } else {
173         Serial.println("No hay conexión a WiFi. Datos permanecerán en el ESP32."); // Mensaje si no hay conexión con Wi-Fi
174     }
175 }

```

Elaborado por: La Autora

3.5.3.2. Diagrama de flujo del Nodo Sensor

El diagrama de flujo del Nodo Sensor de la figura 40 explica cómo funciona el sistema de monitoreo utilizando el ESP32. El proceso comienza con la inicialización del ESP32, seguida por el intento de conexión a la red Wi-Fi. Si la conexión es exitosa, se establece la conexión con la base de datos.

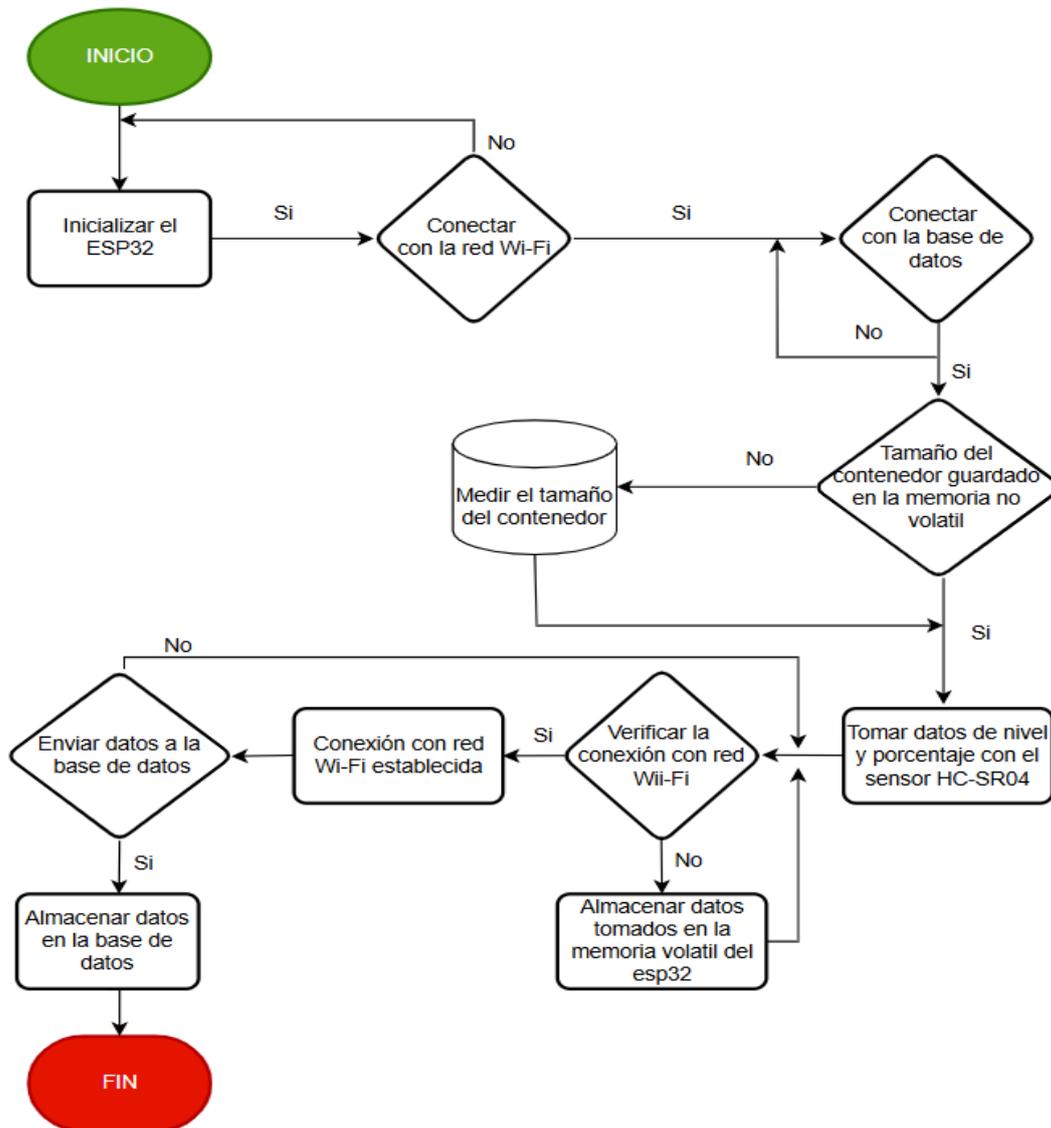
A continuación, se verifica si el tamaño del contenedor está almacenado en la memoria del ESP32. Si no está guardado, mide el tamaño del contenedor y lo almacena.

Una vez confirmado el tamaño, el sistema toma lecturas del nivel de llenado y el porcentaje utilizando el sensor ultrasónico HC-SR04.

Antes de enviar los datos a la base de datos, el sistema verifica nuevamente la conexión Wi-Fi. Si la conexión está activa, envía los datos a la base de datos y, si el envío es exitoso, los datos se almacenan y el proceso finaliza. En caso de que no haya conexión Wi-Fi, los datos se guardan temporalmente en la memoria del ESP32 hasta que la conexión sea restablecida, asegurando que no se pierda información. Si hay fallos en algún paso, el sistema reintenta las operaciones hasta completarlas correctamente.

Figura 40

Diagrama de Flujo del Nodo Sensor



Elaborado por: La Autora

3.5.4. Configuración de la Aplicación Móvil con Flutter

3.5.4.1. Autenticación de Usuarios

En el desarrollo de la aplicación móvil utilizando Flutter, se implementa un sistema de autenticación de usuarios para controlar el acceso. En la figura 41, se define una función específica que maneja este proceso, tomando como parámetros principales el correo electrónico y la contraseña ingresados por el usuario. Esta función verifica las

credenciales en la base de datos de usuarios, previamente validados por el administrador. Esto asegura que solo los usuarios autenticados puedan acceder a las funcionalidades de la aplicación, mejorando así la seguridad y el control de acceso.

Figura 41

Función de Inicio de Sesión

```
Future<void> login() async {
  try {
    final String email = emailController.text.trim();
    final String password = passwordController.text.trim();
    // Buscar el usuario en la base de datos
    final DataSnapshot snapshot = await _databaseRef.child(path: 'usuarios_recolectores').get();
    bool userFound = false;
    if (snapshot.exists) {
      for (var DataSnapshot child in snapshot.children) {
        final Map<dynamic, dynamic> data = child.value as Map<dynamic, dynamic>;
        if (data['correo'] == email && data['password'] == password) {
          userFound = true;
          break;
        }
      }
    }
    if (userFound) {
      Navigator.push<dynamic>(
        context: context,
        route: MaterialPageRoute<dynamic>(
          builder: (BuildContext context) => const RecolectorDashboardPage(), // MaterialPageRoute
        );
    } else {
      // Si el usuario no se encuentra, limpiar los campos
      emailController.clear();
      passwordController.clear();
      ScaffoldMessenger.of(context: context).showSnackBar(
        snackBar: const SnackBar(content: Text(data: "Error en el inicio de sesión")),
      );
    }
  } catch (e) {
    ScaffoldMessenger.of(context: context).showSnackBar(
      snackBar: const SnackBar(
        content: Text(data: "Error en la conexión con la base de datos")), // SnackBar
    );
  }
}
```

Elaborado por: La Autora

3.5.4.2. Niveles de Llenado de los Contenedores de Desechos Infecciosos

Una parte esencial de la aplicación móvil es la visualización de datos. Para lograr esto, se accede a la información almacenada en la base de datos Firebase, que contiene

los registros capturados por el nodo sensor. Específicamente, la aplicación consulta los niveles de llenado de cada contenedor en Firebase. Estos datos se recuperan en tiempo real y se presentan de manera clara y comprensible en la interfaz de la aplicación móvil, permitiendo a los usuarios monitorear fácilmente el estado actual de los contenedores de desechos infecciosos. Esta funcionalidad proporciona una visión actualizada y precisa del sistema de monitoreo, facilitando la toma de decisiones basada en datos en tiempo real. En la figura 42, se observa el código que permite cumplir con la obtención de datos de los contenedores.

Figura 42

Obtención de datos de los Contenedores (Llamado a base de datos Firebase)

```
Stream<Map<String, String>> obtenerUltimosDatos(  
    String laboratorio, String contenedor) {  
    final DatabaseReference ref =  
        FirebaseDatabase.getInstance().ref().child(path: laboratorio).child(path: contenedor);
```

Elaborado por: La Autora

Para visualizar las mediciones realizadas por los sensores, la aplicación extrae el último valor enviado a la base de datos para cada contenedor (uno y dos). Se implementa una lógica de comparación para verificar que los valores consultados en la base de datos no sean cero, como se observa en la figura 43. Si se detecta un valor válido (distinto de cero), este se muestra en la aplicación, representando el nivel actual de llenado del contenedor. En caso de que el sensor no esté realizando mediciones o se detecte un valor de cero, la aplicación establece los valores en cero para indicar la ausencia de datos válidos. Este proceso asegura que la información mostrada sea precisa y actualizada, reflejando el estado real de los contenedores o la falta de mediciones cuando corresponda.

Figura 43

Obtención del último dato guardado (Porcentajes y distancias)

```

return ref.onValue.map<Map<String, String>>(convert: (DatabaseEvent event) {
    String ultimaDistancia = "0";
    String ultimoPorcentaje = "0";

    final DataSnapshot distanciasData = event.snapshot.child(path: 'distancias');
    if (distanciasData.exists && distanciasData.value is Map<dynamic, dynamic>) {
        final Map<String, dynamic> distanciasMap =
            Map<String, dynamic>.from(other: distanciasData.value as Map<dynamic, dynamic>);
        final List<String> ultimaClaveDistancia = distanciasMap.keys.toList()..sort();
        ultimaDistancia =
            distanciasMap[ultimaClaveDistancia.last]?.toString() ?? "0";
    }

    final DataSnapshot porcentajesData = event.snapshot.child(path: 'porcentajes');
    if (porcentajesData.exists && porcentajesData.value is Map<dynamic, dynamic>) {
        final Map<String, dynamic> porcentajesMap =
            Map<String, dynamic>.from(other: porcentajesData.value as Map<dynamic, dynamic>);
        final List<String> ultimaClavePorcentaje = porcentajesMap.keys.toList()..sort();
        ultimoPorcentaje =
            porcentajesMap[ultimaClavePorcentaje.last]?.toString() ?? "0";
    }

    return <String, String>{
        'distancia': ultimaDistancia,
        'porcentaje': ultimoPorcentaje,
    };
});
}

```

Elaborado por: La Autora

3.5.4.3. Informes de Laboratorios

En la pantalla del administrador, se ha incorporado la funcionalidad que permite visualizar informes detallados de las mediciones de los laboratorios, como se observa en la figura 44. Esta característica muestra los valores de las medidas obtenidas por los sensores en una tabla. Además, se incluye el porcentaje de llenado del contenedor y el tiempo en que se realizó cada medición. Esta herramienta facilita un seguimiento preciso y en tiempo real de las actividades del laboratorio.

Figura 44*Conexión con Base de datos*

```

stream: _database.child(path: "Laboratorio_2/$contenedor/porcentajes").onValue,
builder: (BuildContext context, AsyncSnapshot<DatabaseEvent> snapshot) {
  if (snapshot.hasData && snapshot.data!.snapshot.value != null) {
    // Procesa los datos obtenidos de Firebase
  } else if (snapshot.hasError) {
    return const Text(data: "Error al cargar los datos");
  } else {
    return const Center(child: CircularProgressIndicator());
  }
},
) // StreamBuilder

```

Elaborado por: La Autora

En la figura 45, se muestra el código que permite presentar los datos de un contenedor en una tabla. Primero, filtra las entradas de datos para una fecha específica y extrae la hora, el nivel de distancia y el porcentaje de llenado. Luego, ordena estos datos por hora para mostrarlos de manera cronológica. Finalmente, crea una tabla con tres columnas: una para la hora, otra para el nivel (distancia) y otra para el porcentaje de llenado, mostrando cada registro en una fila separada.

Figura 29*Presentación de todos los datos guardados en Base de datos*

```

List<Map<String, dynamic>> details = porcentajes.entries
  .where(test: (MapEntry<dynamic, dynamic> entry) => entry.key.toString().contains(other: date))
  .map<Map<String, dynamic>>(toElement: (MapEntry<dynamic, dynamic> entry) => {
    "hora": entry.key.toString().split(pattern: ' ')[1],
    "nivel": distancias[entry.key] ??
      "-", // Obtener la distancia correspondiente o "-"
    "porcentaje": entry.value,
  })
  .toList()
  ..sort(compare: (Map<String, dynamic> a, Map<String, dynamic> b) =>
    a["hora"].compareTo(b["hora"])); // Ordenar por hora

return DataTable(
  columns: const <DataColumn>[
    DataColumn(label: Text(data: "Hora")),
    DataColumn(label: Text(data: "Nivel")),
    DataColumn(label: Text(data: "Porcentaje")),
  ],
  rows: details.map<DataRow>(toElement: (Map<String, dynamic> entry) {
    return DataRow(
      cells: <DataCell>[
        DataCell(child: Text(data: entry["hora"])),
        DataCell(child: Text(data: entry["nivel"]
          .toString()), // Mostrar el nivel (distancia) // Text // DataCell
        DataCell(child: Text(data: "${entry["porcentaje"]}%"

```

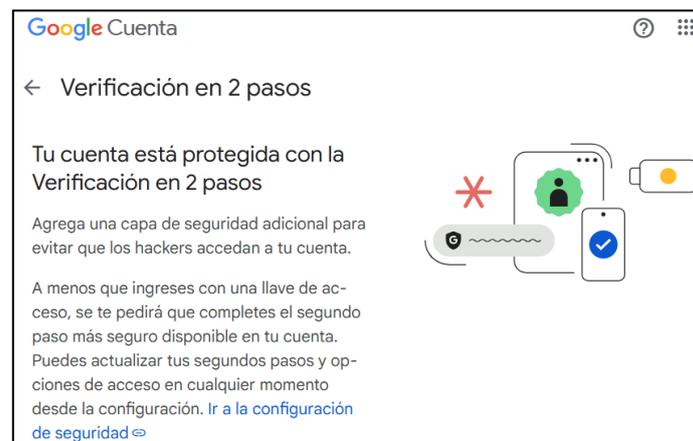
Elaborado por: La Autora

3.5.4.4. Sistema de Notificaciones

Una función clave de la aplicación móvil es el sistema de notificaciones, que se realiza a través de correos electrónicos. Para esto, se configuran cuentas de Gmail, ya que son compatibles con aplicaciones de terceros y permiten gestionar las notificaciones de forma eficiente. Para usar la cuenta en la aplicación, es necesario configurarla activando la verificación en dos pasos, lo que garantiza la autenticidad y seguridad del acceso, como se muestra en la figura 46.

Figura 30

Verificación en dos pasos

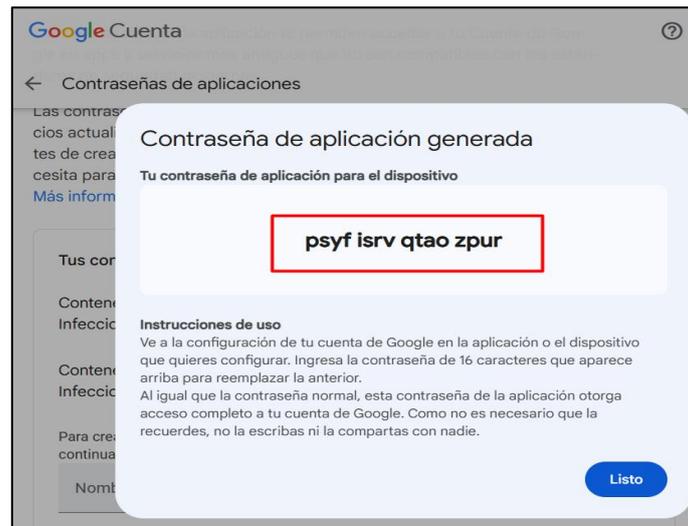


Elaborado por: La Autora

El siguiente paso es crear una contraseña de aplicación, que permitirá utilizar la cuenta de correo para enviar notificaciones desde la aplicación móvil, como se muestra en la figura 47. Una vez generada la contraseña, es importante copiarla y guardarla en un lugar seguro, ya que será necesaria para su uso posterior.

Figura 31

Contraseñas de Aplicaciones



Elaborado por: La Autora

La contraseña obtenida en el caso anterior debe colocarse dentro del código de la aplicación móvil, como se observa en la figura 48, esto en el lugar en donde se ejecuta la función de envío de notificaciones. Aquí se observa la importación de librerías `stmplib` y `email.mime` para poder utilizar el servicio de correo dentro de la aplicación, de igual forma se especifica el destinatario, el correo remitente y la contraseña de acceso para el envío de correo desde el remitente hacia el destinatario.

Figura 32

Conexión de Correo electrónico con Código para Aplicación Móvil

```
import 'package:mailer/mailer.dart';
import 'package:mailer/smtp_server/gmail.dart';

Future<void> enviarCorreo(
  String asunto, String mensaje, List<String> destinatarios) async {
  final smtpServer = gmail('melyacalero99@gmail.com',
    'psyf isrv qtao zpur'); // contraseña de aplicaciones de Google
```

Elaborado por: La Autora

En la figura 49, se observa la función para realizar la conexión con el servidor de correo, en este caso el servicio de Gmail, esto se realiza para poder utilizar el correo electrónico como medio para el envío de notificaciones.

Figura 33

Envío de correo electrónico (Notificación)

```
try {
  await send(email, smtpServer);
  print('Correo enviado con éxito');
} on MailerException catch (e) {
  print('Error al enviar el correo: $e');
  for (var p in e.problems) {
    print('Problema: ${p.code}: ${p.msg}');
  }
}
```

Elaborado por: La Autora

En la figura 50, se define una función que envía una notificación por correo a un grupo específico de usuarios. Primero, obtiene una lista de correos y verifica si el grupo solicitado es, en este caso, usuarios_laboratorio1. Si el grupo existe y tiene correos asociados, envía el correo con el asunto y mensaje especificados. Luego, muestra una confirmación en pantalla con un mensaje verde de éxito. Si el grupo no existe o no tiene correos, muestra un mensaje de error en pantalla con un color rojo.

Figura 34*Mensajes de Notificaciones*

```

Future<void> enviarNotificacion(
    String asunto, String mensaje, String grupo) async {
  try {
    final correos =
      await obtenerCorreos();

    // Verificar si el grupo es usuarios_laboratorio1
    if (grupo == "usuarios_laboratorio1") {
      // Enviar notificación a usuarios_laboratorio1
      if (correos.containsKey(grupo) && correos[grupo]!.isNotEmpty) {
        await enviarCorreo(asunto, mensaje, correos[grupo]!);

        ScaffoldMessenger.of(context).showSnackBar(
          SnackBar(
            content:
              Text("Notificación enviada a usuarios_laboratorio1: $asunto"),
            backgroundColor: Colors.green,
            duration: const Duration(seconds: 1),
          ), // SnackBar
        );
        print("Correo enviado a grupo $grupo");
      } else {
        ScaffoldMessenger.of(context).showSnackBar(
          SnackBar(
            content:
              Text("No se encontró el grupo $grupo o no tiene correos."),
            backgroundColor: Colors.red,
            duration: const Duration(seconds: 1),
          ), // SnackBar
        );
      }
    }
  }
}

```

Elaborado por: La Autora

3.5.4.5. Cálculo de Rutas mediante el Algoritmo de Dijkstra

Para verificar la ruta más corta desde un punto de inicio hasta un destino, se aplica el algoritmo de Dijkstra. Primero, se realiza el cálculo manualmente, paso a paso, para identificar la ruta óptima entre los nodos seleccionados. Luego, se implementa el mismo cálculo en código y se comparan ambos resultados. Esto permite confirmar que el algoritmo en código genera la misma ruta más corta que la obtenida manualmente, validando la precisión de la implementación.

- **Aplicación del Algoritmo de Dijkstra de forma manual**

Para calcular la ruta más corta mediante el algoritmo de Dijkstra, se ha utilizado un grafo con nodos dirigidos que representan las diferentes ubicaciones y caminos entre ellas. En el grafo presente en la figura 51, se han marcado con diferentes colores las posibles rutas, donde se han especificado las direcciones de los tramos tomando en cuenta como nodos de origen el Parqueadero recolector (nodo A), el laboratorio clínico ORION (nodo g) y el laboratorio clínico Micro Biolac (nodo k), y como nodo destino al Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura (nodo Z). Cada conexión entre nodos muestra la longitud de cada tramo, conocidas como "aristas ponderadas".

Figura 35

Grafo con nodos dirigidos



Elaborado por: La Autora

En este caso se realizará el cálculo de la ruta más corta desde el “Parqueadero Recolector” (nodo A) hasta el “Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura” (nodo Z) usando el algoritmo de Dijkstra, primero se asigna un valor inicial de cero al nodo de origen (nodo A) y se establece una distancia infinita para el resto de los nodos. A continuación, se examinan las aristas ponderadas desde el

Nodo	Paso17	Paso18	Paso19	Paso20	Paso21	Paso22	Paso23	Paso24	Paso25
R	(1327,S)	x	x	x	x	x	x	x	x
L	(1368,K)	(1368,K)	x	x	x	x	x	x	x
K	x	x	x	x	x	x	x	x	x
Q	∞	(1540,R)	(1540,R)	x	x	x	x	x	x
M	∞	∞	(1621,L)	(1621,L)	x	x	x	x	x
O	∞	∞	∞	(1828,Q)	(1828,Q)	(1828,Q)	x	x	x
N	∞	∞	∞	∞	(1729,M)	x	x	x	x
P	∞	∞	∞	∞	∞	∞	(1933,O)	x	x
Y	∞	∞	∞	∞	∞	∞	∞	(2137,P)	x
Z	∞	∞	∞	∞	∞	∞	∞	∞	(2170,Y)

Nota. La tabla 17 representa los pasos del algoritmo de Dijkstra, con nodos en amarillo seleccionados en cada iteración y valores en verde indicando el costo mínimo acumulado y su predecesor.

Fuente. La Autora

▪ Aplicación del Algoritmo de Dijkstra mediante código

Antes de calcular la ruta más corta con el algoritmo de Dijkstra, es esencial definir los nodos y sus ubicaciones. En la figura 52, cada nodo representa un punto específico en el mapa, definido por coordenadas de latitud y longitud (LatLng). Esto incluye puntos clave como el "Parqueadero Recolectores" y varios laboratorios. Al establecer estos nodos, el algoritmo puede entender el mapa y calcular la ruta más corta entre el origen y el destino.

Figura 36

Nodos y sus ubicaciones

```
final Map<String, LatLng> locations = {
    'Parqueadero Recolectores': LatLng(0.3599963, -78.13006),
    'NodoB': LatLng(0.3603531, -78.13055),
    'NodoC': LatLng(0.3611282, -78.12959),
    'NodoD': LatLng(0.3619785, -78.12773),
    'NodoE': LatLng(0.3607742, -78.12602),
    'NodoF': LatLng(0.3605140, -78.12588),
    'NodoG': LatLng(0.3598488, -78.12684),
    'NodoH': LatLng(0.3611121, -78.12514),
    'NodoI': LatLng(0.3589651, -78.12120),
```

Elaborado por: La Autora

También es importante que las conexiones entre los nodos están etiquetadas con la distancia en metros, como se muestra en la figura 53. Estas distancias entre nodos son esenciales para que el algoritmo de Dijkstra pueda calcular la ruta más corta desde un

punto de origen hasta un destino, evaluando las conexiones disponibles y eligiendo el camino con menor costo en función de las distancias definidas en el grafo.

Figura 37

Definición de distancias entre nodos (aristas).

```
final Map<String, Map<String, double>> graph = {
  'Parqueadero Recolectores': {'NodoB': 71},
  'NodoB': {'Parqueadero Recolectores': 71, 'NodoC': 137, 'NodoX': 452},
  'NodoC': {'NodoB': 137, 'NodoD': 226},
  'NodoD': {'NodoC': 226, 'NodoE': 136},
  'NodoE': {'NodoD': 136, 'NodoF': 32, 'NodoG': 142},
  'NodoF': {'NodoE': 32, 'NodoH': 105},
  'NodoG': {'NodoE': 142, 'NodoJ': 508},
  'NodoH': {'NodoF': 105, 'NodoI': 493},
  'NodoI': {'NodoH': 493, 'NodoL': 302},
}
```

Elaborado por: La Autora

A continuación, ya se puede dar inicio con la aplicación del algoritmo de Dijkstra. Primero, se inicializan las distancias de todos los nodos desde el nodo de inicio (startNode), asignando una distancia de 0 al nodo inicial y una distancia infinita a los demás. En la figura 54, se observa el código para la inicialización de las distancias de los nodos.

Figura 38

Inicialización de las distancias de todos los nodos

```
void findShortestPath() {
  if (startNode == null || endNode == null) return;

  String start = startNode!;
  String end = endNode!;
  Map<String, double> distances = {};
  Map<String, String?> previous = {};

  for (var node in graph.keys) {
    distances[node] = double.infinity;
    previous[node] = null;
  }

  distances[start] = 0.0;
  List<String> unvisited = graph.keys.toList();
}
```

Elaborado por: La Autora

Luego, mientras haya nodos no visitados, se selecciona el nodo con la menor distancia conocida, llamado `currentNode`. Para cada vecino de `currentNode`, el código calcula una distancia provisional sumando la distancia actual de `currentNode` y el peso de la conexión (arista) hacia el vecino, como se muestra en la figura 55. Si esta distancia es menor que la registrada, se actualiza la distancia y se guarda `currentNode` como el nodo anterior en la ruta.

Figura 39

Selección de nodo con menor distancia conocida

```

while (unvisited.isNotEmpty) {
    String currentNode =
        unvisited.reduce((a, b) => (distances[a]!) < (distances[b]!) ? a : b);

    unvisited.remove(currentNode);

    if (currentNode == end) break;

    if (graph[currentNode] != null) {
        for (var neighbor in graph[currentNode]!.keys) {
            distances.putIfAbsent(neighbor, () => double.infinity);
            previous.putIfAbsent(neighbor, () => null);

            double distance =
                distances[currentNode]! + graph[currentNode]![neighbor]!;

            if (distance < distances[neighbor]!) {
                distances[neighbor] = distance;
                previous[neighbor] = currentNode;
            }
        }
    }
}

```

Elaborado por: La Autora

Cuando se encuentra el nodo de destino (`endNode`), se termina el cálculo y se reconstruye la ruta más corta, rastreando desde el nodo de destino hacia el inicio mediante los nodos anteriores guardados (figura 56). Finalmente, esta ruta se guarda y se muestra en el mapa ajustando la vista para que sea visible.

Figura 40*Reconstrucción de la ruta más corta*

```

List<String> path = [];
String? currentNode = end;
while (currentNode != null) {
    path.insert(0, currentNode);
    currentNode = previous[currentNode];
}

setState(() {
    shortestPath = path.map((node) => locations[node!]).toList();
    _fitMapToPath();
});

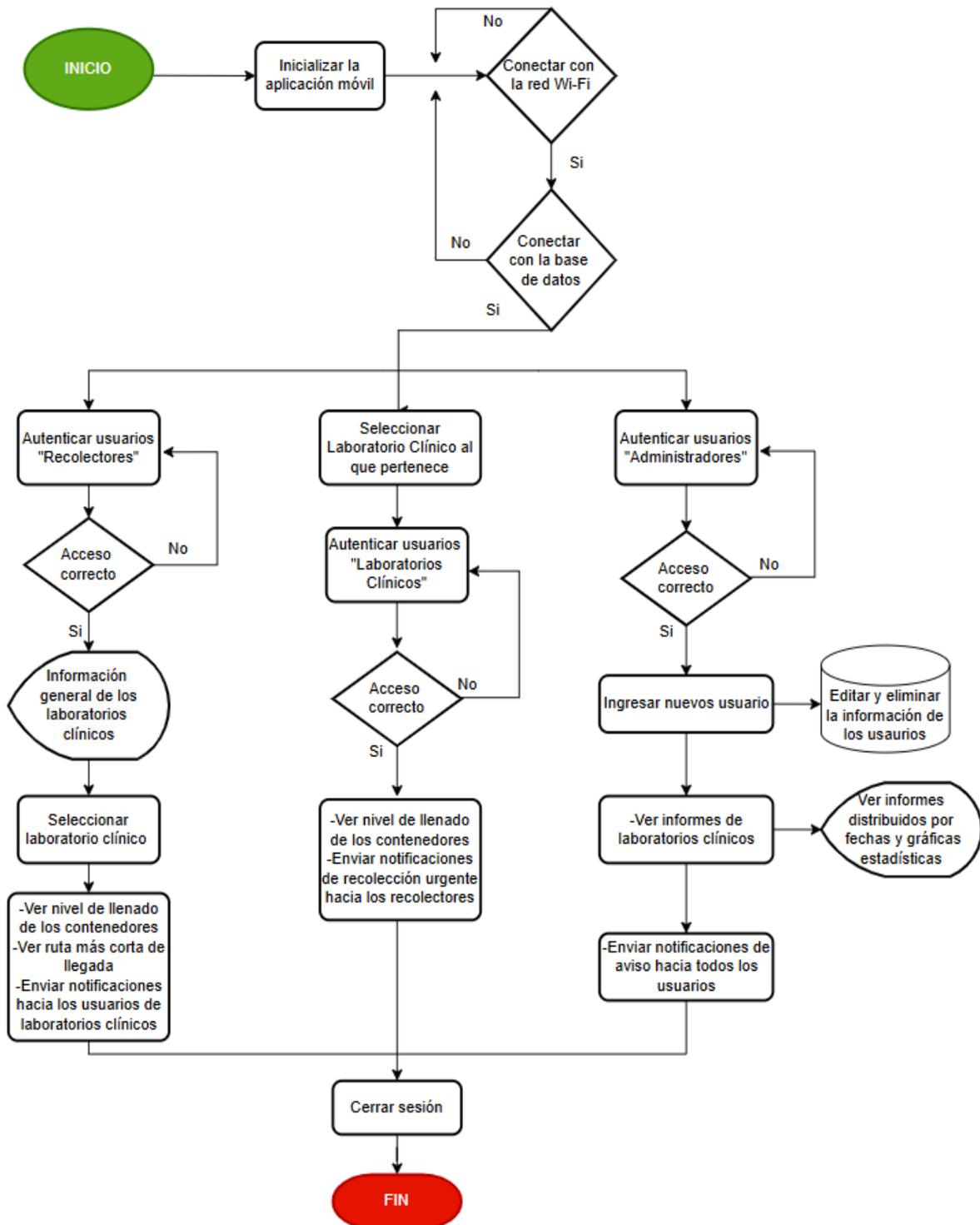
```

*Elaborado por: La Autora***3.5.4.6. Diagrama de Flujo de la Aplicación Móvil**

El diagrama de flujo presente en la figura 57 muestra el proceso de uso de la aplicación móvil para gestionar el sistema de monitoreo de desechos infecciosos. Inicia con la inicialización de la aplicación móvil y la conexión a la red Wi-Fi y la base de datos. Los usuarios se autentican según su rol: recolectores, laboratorios clínicos o administradores. Si la autenticación es exitosa, los recolectores acceden a información general de los laboratorios, seleccionan uno y verifican el nivel de llenado de los contenedores, además de consultar la ruta más corta para la recolección, y enviar notificaciones hacia los usuarios de los laboratorios. Los usuarios de laboratorios clínicos verifican el nivel de llenado de los contenedores y envían notificaciones hacia los recolectores si es necesaria una recolección urgente. Los administradores gestionan la información de los usuarios, crean nuevos usuarios y generan informes de los laboratorios clínicos, además de enviar notificaciones de aviso hacia todos los usuarios, en caso de ser necesario. Una vez completadas las tareas, los usuarios cierran sesión, finalizando el proceso.

Figura 41

Diagrama de Flujo de la Aplicación Móvil



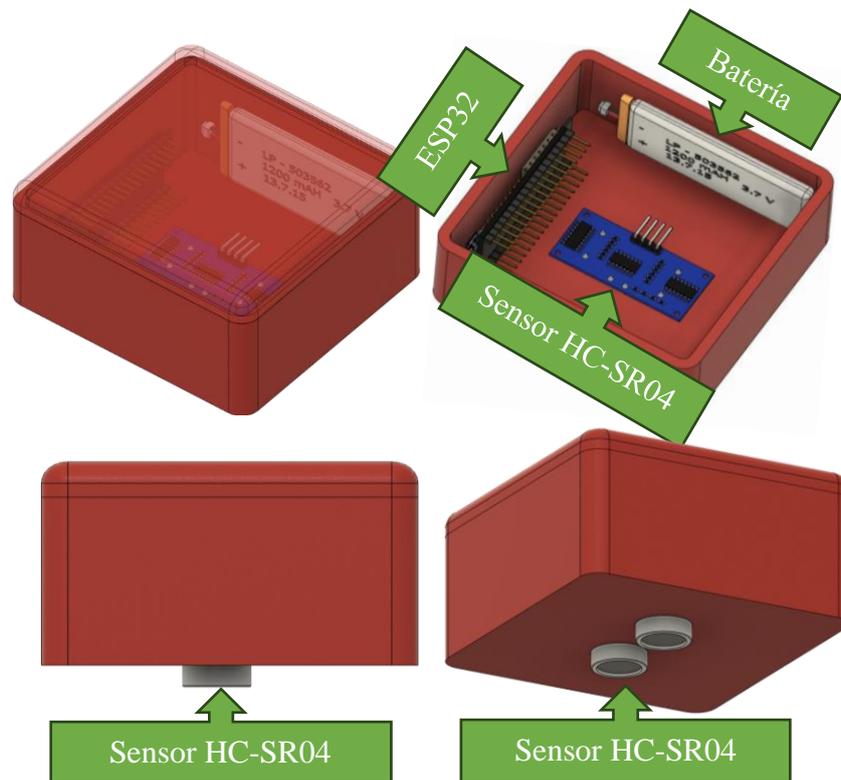
Elaborado por: La Autora

3.6. Diseño y Montaje de la Carcasa para el Nodo Sensor

En esta sección se describirá el diseño y el montaje del nodo sensor, ensamblando los componentes electrónicos previamente seleccionados y organizándolos de manera óptima para asegurar un funcionamiento eficiente. El diseño final busca una disposición física adecuada y un consumo energético optimizado, permitiendo que el nodo capte y transmita datos de forma fiable en el entorno de monitoreo.

3.6.1. Distribución de Componentes en el Nodo Sensor

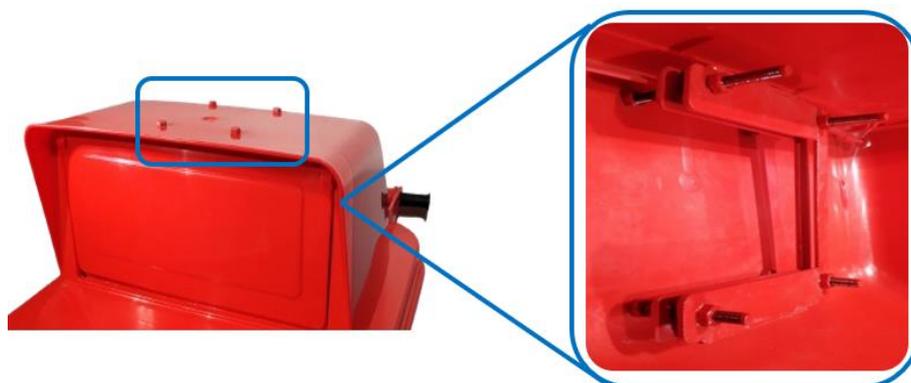
El nodo sensor cuenta con una estructura compacta y bien organizada, en la que cada componente se ha ubicado estratégicamente para maximizar el espacio y asegurar su funcionamiento. En el interior de la caja, la placa ESP 32 se coloca en una posición central para facilitar las conexiones eléctricas con los demás elementos. El sensor ultrasónico HC-SR04 está ubicado en la parte inferior de la caja, con sus emisores y receptores apuntando hacia el exterior, permitiendo una detección precisa de distancia. La batería se sitúa en un compartimento superior, manteniéndola alejada del sensor para evitar interferencias y asegurar un fácil acceso en caso de reemplazo. Este diseño garantiza tanto la funcionalidad del nodo como su durabilidad en entornos de monitoreo. En la figura 58, se puede apreciar de mejor manera la distribución de los componentes del nodo sensor.

Figura 42*Distribución de componentes del nodo sensor**Elaborado por: La Autora***3.6.2. Ubicación del Nodo Sensor en el Contenedor**

En la figura 59 se muestra el sistema de soporte para el Dispositivo Medidor de Nivel dentro de un contenedor. A la izquierda, se observan los tornillos en la tapa del contenedor que aseguran la plataforma donde se ubicará el dispositivo. A la derecha, se ve una vista interna de la tapa, en la que se aprecia una plataforma metálica diseñada para sostener de forma segura el medidor de nivel. Esta estructura garantiza la estabilidad y alineación del dispositivo, permitiendo su correcto funcionamiento en la medición del nivel dentro del contenedor.

Figura 43

Plataforma metálica que sujetará al nodo sensor

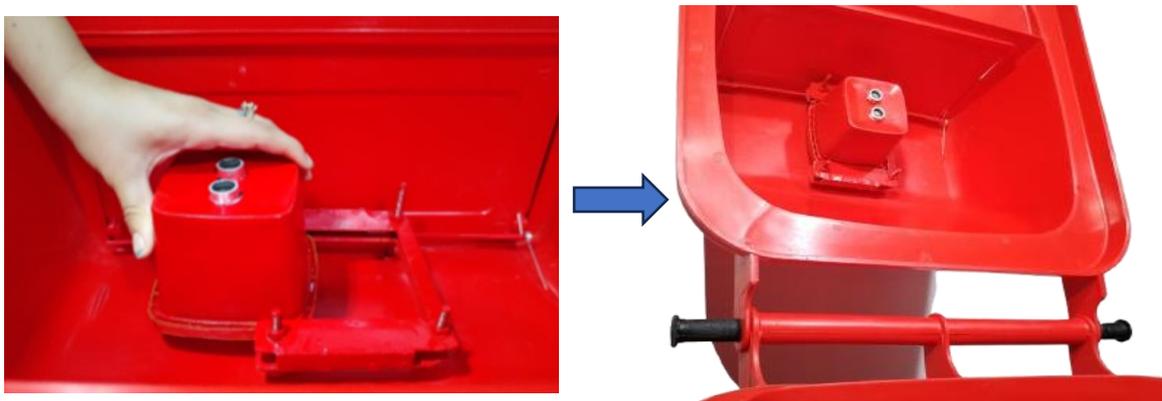


Elaborado por: La Autora

El dispositivo debe colocarse en la parte interna de la tapa del contenedor, encajándolo cuidadosamente sobre la plataforma metálica de soporte, como se observa en la figura 60. Esta posición permite que el dispositivo quede firme y orientado adecuadamente para su funcionamiento.

Figura 44

Colocación del nodo sensor



Elaborado por: La Autora

3.7. Funcionamiento de la Aplicación Móvil

A continuación, se presenta el funcionamiento de la aplicación móvil, denominada "Bio Nivel Monitor" debido a su enfoque en el monitoreo y gestión eficiente del nivel de residuos infecciosos. Donde se destacan sus principales características y funcionalidades,

permitiendo comprender cómo opera la aplicación, así como la interacción del usuario con sus diferentes módulos y opciones.

3.7.1. Página Principal de Aplicación Móvil

Al iniciar la aplicación "Sistema de Monitoreo de los Contenedores de Desechos Infecciosos de Laboratorios Clínicos de Ibarra", se muestra una pantalla de bienvenida con el símbolo de riesgo biológico, destacando la importancia de los desechos gestionados. El usuario puede elegir su tipo de acceso entre "Recolectores", "Laboratorios Clínicos" o "Administrador", como se observa en la figura 61, permitiendo una navegación específica según su rol en el sistema.

Figura 45

Pantalla principal de la aplicación móvil



Elaborado por: La Autora

3.7.2. Usuarios Recolectores de Desechos Infecciosos

Para autenticar a los recolectores, se requiere ingresar correo y contraseña, luego dar clic en "Iniciar sesión". Si las credenciales son correctas, se accede a la página de

visualización para recolectores de desechos infecciosos (figura 62). Tras ingresar correctamente, los recolectores acceden a una página con información general sobre el porcentaje de llenado de los contenedores en cada laboratorio clínico. Para facilitar la comprensión, se utiliza un código de colores: rojo para laboratorios con contenedores llenos al 70% o más, naranja para niveles entre 50% y 60%, y verde para niveles inferiores al 50%. Se recomienda al recolector dar prioridad a los laboratorios clínicos en color rojo, ya que indican que sus contenedores están próximos a su capacidad máxima.

Figura 46

Página de inicio de sesión de usuarios recolectores e ingreso a página principal



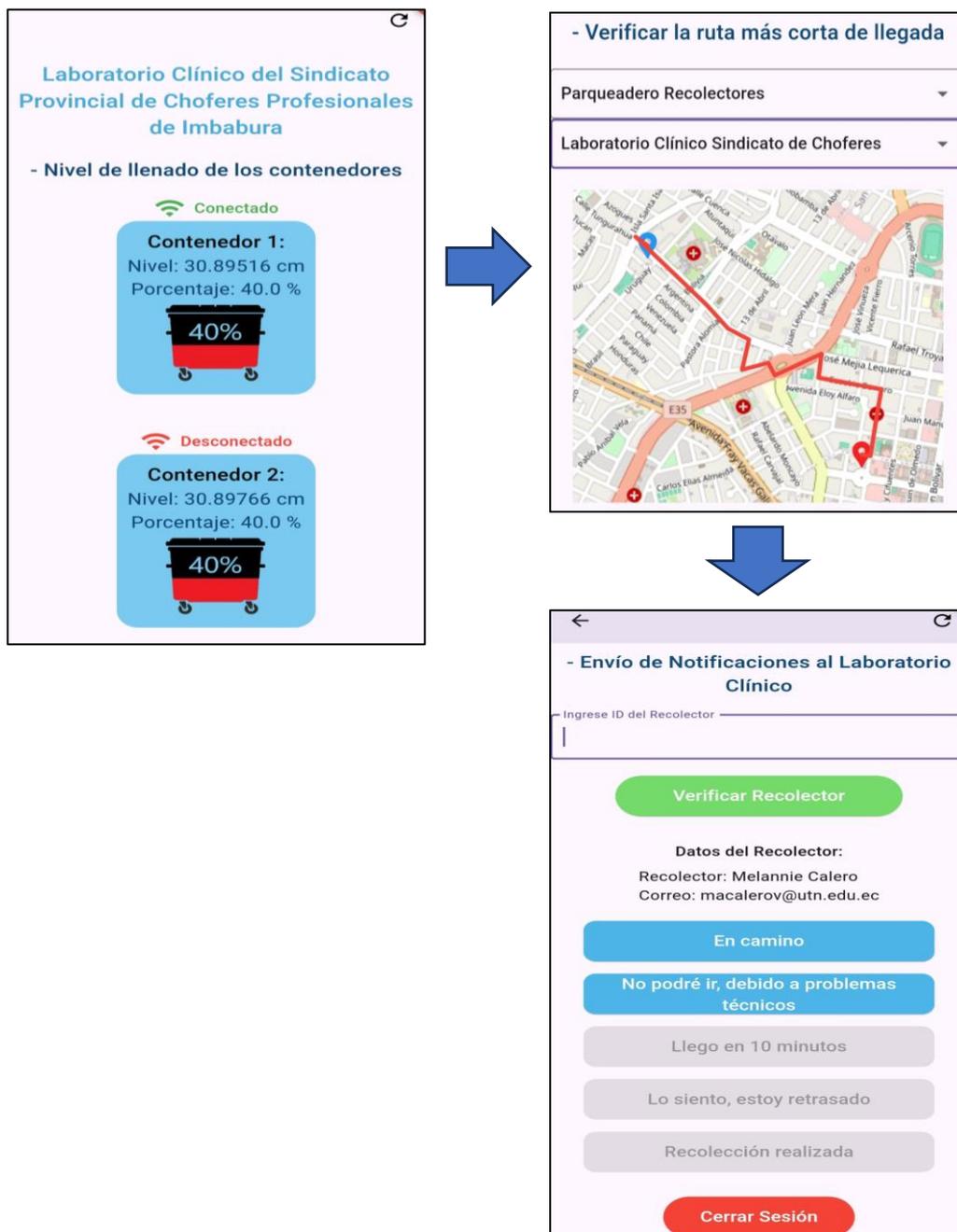
Elaborado por: La Autora

Al seleccionar uno de los laboratorios clínicos, se presenta una interfaz dividida en tres secciones para facilitar su uso (figura 63). La primera pantalla (superior izquierda) muestra el nivel de llenado de los contenedores del laboratorio clínico, incluyendo el estado de conexión, el nivel en centímetros y el porcentaje de llenado. La segunda pantalla (superior derecha) permite verificar la ruta más corta desde un punto de origen hasta el

laboratorio clínico seleccionado, visualizada mediante un mapa. La tercera pantalla (inferior derecha) permite enviar notificaciones al laboratorio clínico, incluyendo opciones como informar el estado del recolector ("En camino", "No podré ir debido a problemas técnicos", entre otras), además de mostrar información del recolector asociado.

Figura 47

Página de laboratorio clínico



Elaborado por: La Autora

3.7.3. Usuarios de Laboratorios Clínicos

En la figura 64, el proceso de inicio de sesión de los usuarios de laboratorio clínico comienza con una pantalla en la que se presenta un listado de laboratorios clínicos para que el usuario seleccione al que pertenece. Una vez elegido el laboratorio, se muestra una nueva pantalla de autenticación donde el usuario debe ingresar su correo y contraseña para acceder al sistema. Este flujo asegura que cada usuario inicie sesión en el laboratorio correspondiente, proporcionando una interfaz clara y personalizada para cada institución.

Figura 48

Página de inicio de sesión de usuarios de laboratorios clínicos



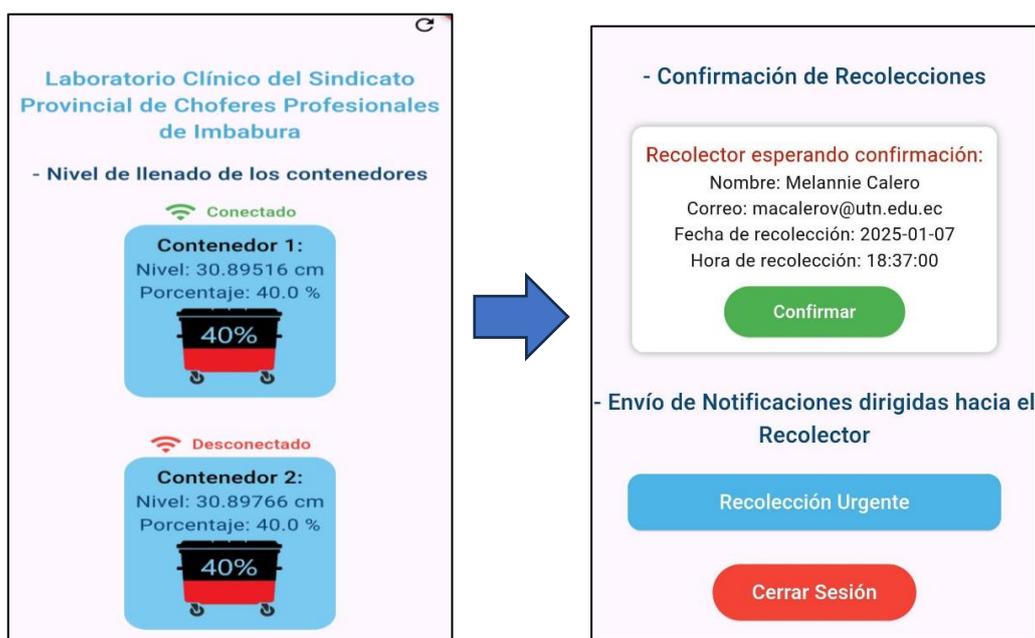
Elaborado por: La Autora

En la figura 65, se observa la pantalla que se muestra al iniciar sesión como usuario de laboratorio clínico, dividida en dos partes para una mejor comprensión. En la primera parte, se muestra el nivel de llenado de los contenedores del laboratorio clínico, indicando el estado de conexión, el nivel en centímetros y el porcentaje de llenado de cada contenedor. En la segunda parte, se presenta la sección de confirmación de

recoleciones, donde se muestra información del recolector (nombre, correo, fecha y hora programada de recolección) y un botón para confirmar la recolección. Además, se incluye un botón de "Recolección Urgente" para enviar una notificación inmediata al recolector en casos especiales que requieran atención prioritaria.

Figura 49

Pantalla principal de usuarios de laboratorio clínico



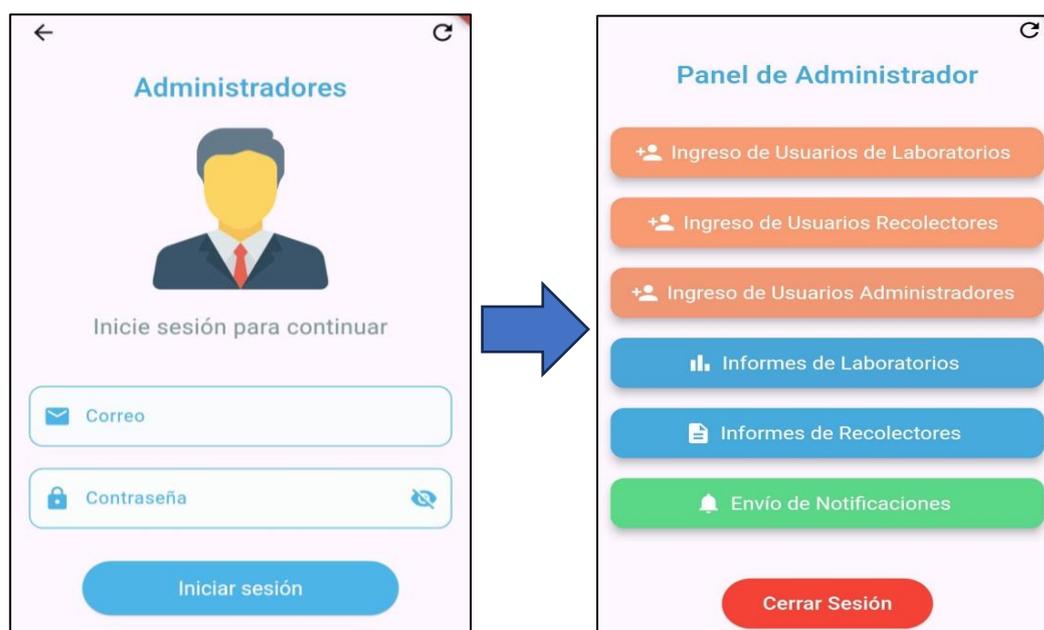
Elaborado por: La Autora

3.7.4. Usuarios de Administración de Aplicación Móvil

La figura 66 muestra dos pantallas correspondientes al módulo de administradores. La primera pantalla (izquierda) es la interfaz de inicio de sesión, donde los administradores deben ingresar su correo y contraseña para acceder al sistema. La segunda pantalla (derecha) es el panel principal del administrador, que ofrece opciones como el ingreso de usuarios de laboratorios, recolectores y administradores, así como acceso a informes de laboratorios, informes de recolectores y la función de envío de notificaciones. También incluye un botón para cerrar sesión.

Figura 50

Inicio de sesión como usuario administrador



Elaborado por: La Autora

3.7.4.1. Registro de Usuarios de Laboratorio, Recolectores, Administradores

El proceso de registro de usuarios se realiza para laboratorios clínicos, recolectores y administradores, pero en el caso específico de los usuarios de laboratorios clínicos, se requiere primero seleccionar el laboratorio al que pertenecerá dicho usuario, como se observa en la figura 67. Una vez realizado este paso, el resto del procedimiento es igual para los tres tipos de usuarios.

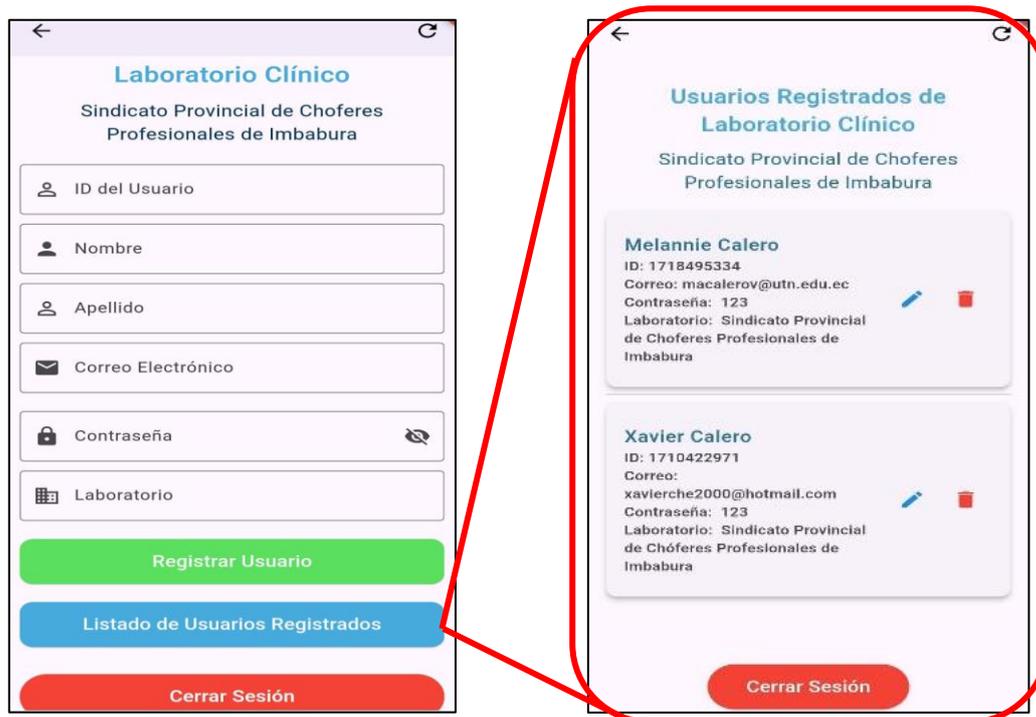
Figura 51*Registro de usuarios de laboratorio clínico*

Elaborado por: La Autora

Las imágenes muestran el proceso general de gestión de usuarios en laboratorios clínicos. En la primera pantalla (izquierda), se presenta un formulario para registrar nuevos usuarios ingresando información básica como ID, nombre, apellido, correo electrónico, contraseña y el laboratorio al que pertenecen. Posteriormente, al seleccionar la opción de "Listado de usuarios registrados", se accede a una pantalla (derecha) que muestra los usuarios ya registrados con sus datos principales y opciones para editar o eliminar sus registros, facilitando la administración de usuarios, como se observa en la figura 68.

Figura 52

Registro de usuarios y funciones de edición y eliminación de información registrada



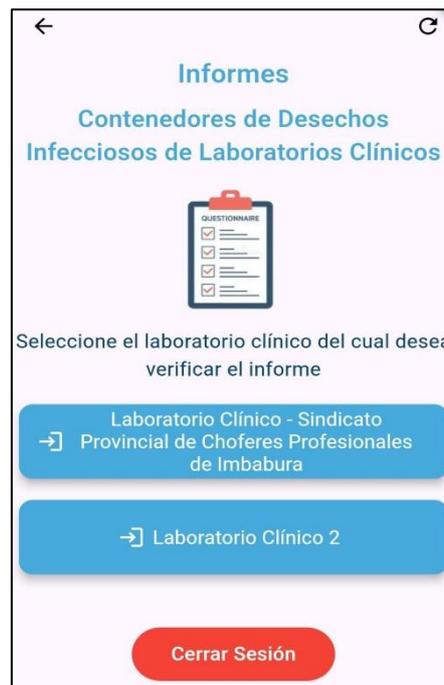
Elaborado por: La Autora

3.7.4.2. Informes de Laboratorios

La figura 69 muestra el proceso de consulta de informes sobre los niveles de llenado de contenedores de desechos infecciosos en laboratorios clínicos. En la pantalla, el usuario puede seleccionar el laboratorio clínico del cual desea obtener el informe, mostrando una lista de opciones disponibles.

Figura 53

Verificación de informes de los contenedores de los laboratorios clínicos

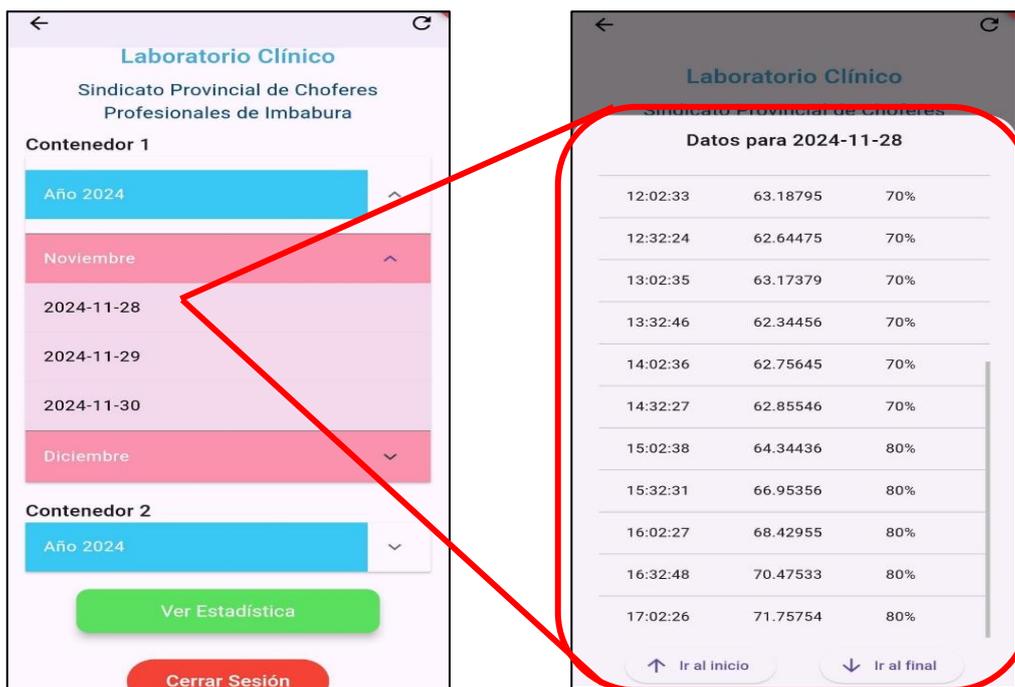


Elaborado por: La Autora

En la primera pantalla (izquierda), el usuario selecciona el contenedor, el año, el mes y finalmente un día específico para visualizar los datos registrados. Al seleccionar una fecha, se muestra en la pantalla (derecha) una lista detallada con los registros de ese día, incluyendo las horas específicas, el nivel en centímetros y el porcentaje de llenado. Además, la interfaz incluye botones para desplazarse rápidamente al inicio o al final de los registros del día seleccionado, como se observa en la figura 70.

Figura 54

Informes de los contenedores de los laboratorios clínicos

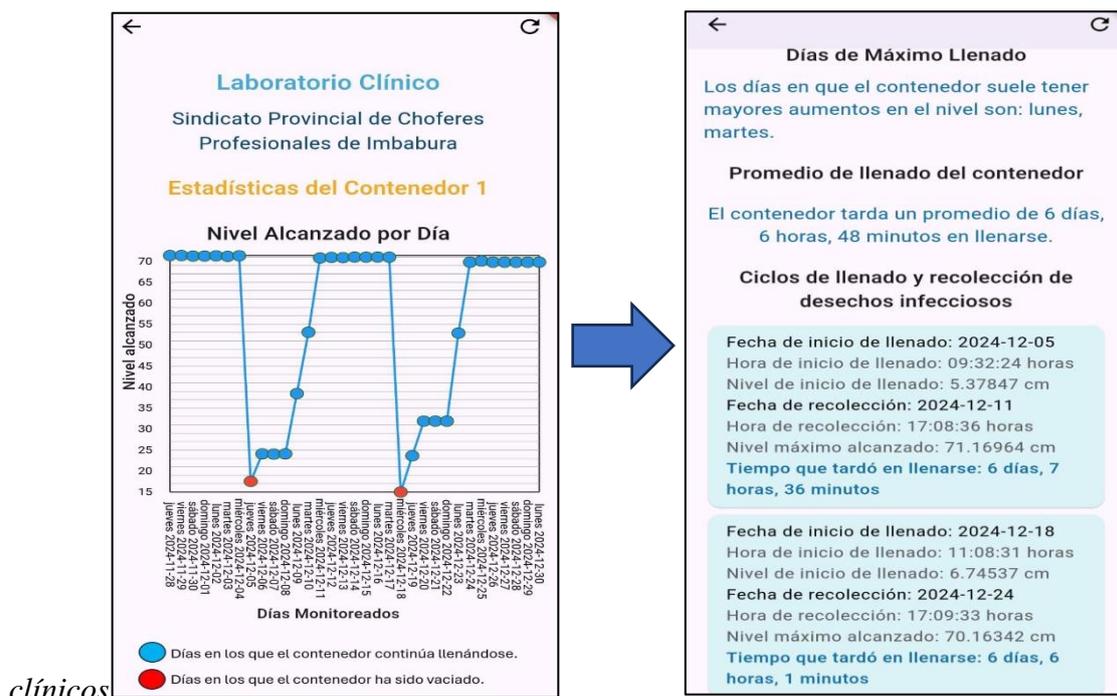


Elaborado por: La Autora

La figura 71 presenta un análisis detallado de los contenedores de desechos infecciosos, mostrando su comportamiento de llenado y vaciado, así como los patrones observados. En la imagen de la izquierda, se visualiza un gráfico que representa el nivel alcanzado diariamente por un contenedor en un laboratorio clínico. Los puntos azules indican los días en los que el contenedor continúa llenándose, mientras que los puntos rojos señalan los días en que ha sido vaciado. En la imagen de la derecha se incluye un análisis más profundo, identificando los días con mayor nivel de llenado, los cuales suelen ser los lunes y martes. También se detallan los ciclos de llenado y recolección, proporcionando información específica como las fechas y horas de inicio y fin del llenado, el nivel máximo alcanzado y el tiempo exacto que tomó llenar el contenedor antes de su vaciado.

Figura 55

Análisis de los contenedores de desechos infecciosos ubicados en los laboratorios



clínicos

Elaborado por: La Autora

3.7.4.3. Informes de Recolectores

La figura 72 muestra la interfaz para consultar informes relacionados con los recolectores de desechos infecciosos. En la parte superior, el usuario dispone de opciones de búsqueda avanzadas que incluyen filtros como nombre, apellido, ID, una segunda palabra opcional para acotar la búsqueda, y la selección de un rango de fechas mediante campos para la fecha de inicio y fin.

En la parte inferior, se despliegan los resultados de la consulta, con detalles específicos para cada recolector. Cada registro incluye el nombre completo, ID, correo electrónico del recolector, el laboratorio asociado al servicio de recolección, la fecha y la hora en que se realizó la recolección. Esta funcionalidad facilita el acceso a información detallada y organizada para el monitoreo y seguimiento de las actividades de recolección.

Figura 56

Listado de Recolecciones Realizadas

The screenshot displays a mobile application interface with the following elements:

- Header:** "Informes de Recolectores de Desechos Infecciosos" in blue text.
- Search Filters:** Four rounded rectangular buttons with search icons:
 - "Buscar por nombre, apellido, ID, etc."
 - "Buscar (segunda palabra opcional)"
 - "Seleccionar fecha de inicio"
 - "Seleccionar fecha de fin"
- Results:** "Resultados encontrados: 24" followed by two identical result cards.
 - Card 1:**
 - Nombre: Melannie Calero
 - ID: 1718495334
 - Correo: macalero@utn.edu.ec
 - Laboratorio: Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura
 - Fecha de recolección: 2025-01-07
 - Hora de recolección: 18:37:00
 - Card 2:**
 - Nombre: Melannie Calero
 - ID: 1718495334
 - Correo: macalero@utn.edu.ec
 - Laboratorio: Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura
 - Fecha de recolección: 2024-12-20
 - Hora de recolección: 16:20:32

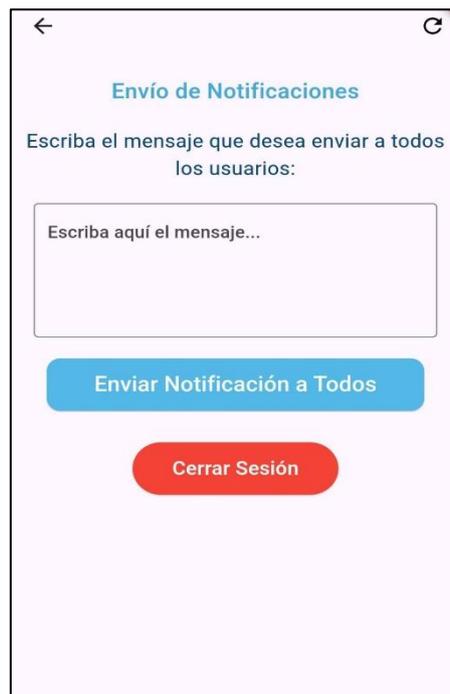
Elaborado por: La Autora

3.7.4.4. Envío de Notificaciones

La figura 73, muestra una pantalla destinada al envío de notificaciones masivas a todos los usuarios, diseñada para que el administrador pueda editar y personalizar el mensaje que desea enviar. La interfaz incluye un campo de texto donde el administrador puede redactar el contenido del mensaje, asegurando que la información sea clara y específica para los usuarios. Una vez redactado, al presionar el botón "Enviar Notificación a Todos", el sistema envía el mensaje personalizado a cada usuario, garantizando una comunicación efectiva. Además, la funcionalidad incluye mensajes de confirmación que aseguran que las notificaciones se han enviado correctamente.

Figura 57

Envío de notificaciones desde usuario administrador



The screenshot shows a mobile application interface with a light pink background. At the top left is a back arrow icon, and at the top right is a refresh icon. The title "Envío de Notificaciones" is centered in blue. Below the title, the text "Escriba el mensaje que desea enviar a todos los usuarios:" is displayed. A text input field with a light pink border contains the placeholder text "Escriba aquí el mensaje...". Below the input field is a blue button with the text "Enviar Notificación a Todos". At the bottom is a red button with the text "Cerrar Sesión".

Elaborado por: La Autora

4. Capítulo IV: RESULTADOS Y ANÁLISIS

En este capítulo, se presenta una evaluación completa del sistema de monitoreo para laboratorios clínicos, incluyendo pruebas en condiciones reales con los nodos sensores en los contenedores. Se analizan los resultados de funcionalidad, usabilidad y rendimiento de la aplicación móvil, así como la durabilidad de los materiales de los sensores. Las pruebas realizadas, detalladas en las Tablas 18 y 19, incluyen verificaciones del funcionamiento general del sistema y verificación de los logros cumplidos, proporcionando una visión clara del desempeño del sistema.

Tabla 18

Evaluación del Funcionamiento General del Sistema

ID de Prueba	Descripción de la Prueba	Objetivo	Criterios de Éxito	Resultado Esperado
1	Verificación de Funcionamiento del Módulo de Inicio de Sesión de Usuarios	Asegurar el correcto inicio de sesión de usuarios	Inicio de sesión exitoso sin errores	Acceso exitoso para el usuario
2	Verificación de Registro, Edición y Eliminación de Usuarios en el Sistema	Verificar registro, edición y eliminación de usuarios	Operaciones de usuario realizadas correctamente	Usuarios gestionados correctamente en el sistema
3	Verificación de la capacidad del nodo sensor para adaptarse a contenedores de diferentes tamaños	Asegurar que el sensor mida correctamente en contenedores de diversos tamaños	Mediciones precisas y consistentes en todos los tamaños de contenedores	Sensor funcionando correctamente en contenedores de diferentes capacidades

Nota: Las pruebas del funcionamiento general del sistema presentes en la tabla 18 permiten verificar el correcto funcionamiento de los módulos principales relacionados con la autenticación y gestión de usuarios, asegurando que cada operación se ejecute de manera exitosa y sin errores dentro del sistema.

Tabla 19

Evaluación de logros del sistema

ID de Prueba	Descripción de la Prueba	Objetivo	Criterios de Éxito	Resultado Esperado
1	Evaluación del Sistema de Monitoreo de	Evaluar la efectividad del	Sistema registra y muestra datos	Datos de residuos

	Contenedores de Residuos Infecciosos	sistema de monitoreo de residuos infecciosos	de monitoreo correctamente	infecciosos monitoreados efectivamente
2	Validación de Mapas para Cálculo de la Ruta más corta entre Origen y Destino	Asegurar que los mapas calculen la ruta más corta correctamente	Ruta más corta calculada y mostrada correctamente	Ruta óptima calculada entre los puntos especificados
3	Verificación del Envío de Notificaciones Personalizadas para Usuarios	Verificar el envío de notificaciones personalizadas	Notificación enviada y recibida correctamente	Usuarios reciben notificaciones como configurado
4	Evaluación de Informes de los Contenedores	Evaluar la calidad y precisión de los informes generados	Informes generados cumplen con precisión requerida	Informes precisos y completos sobre el estado de contenedores

Nota: Las pruebas evaluadas aseguran que el sistema cumpla con los objetivos clave de monitoreo, optimización de rutas, notificaciones personalizadas y generación de informes que permitan realizar análisis.

4.1. Evaluación del Funcionamiento General del Sistema

Se evaluará cómo interactúan entre sí los distintos módulos del sistema para asegurar que funcionen de manera conjunta sin errores y cumplan con los requisitos establecidos, garantizando una operación estable y eficiente.

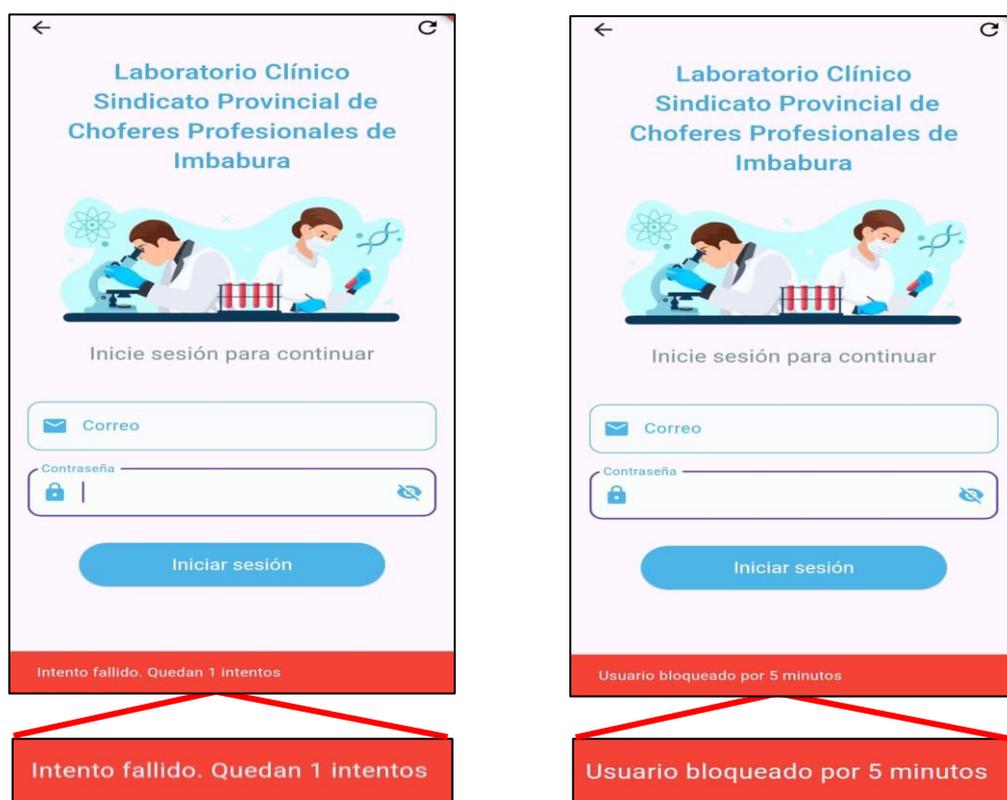
4.1.1. Verificación de Funcionamiento de Inicio de Sesión de Usuarios

Se han realizado 52 pruebas de inicio de sesión correcto, cubriendo todos los tipos de usuarios registrados, y se ha verificado que en todos los casos el proceso de autenticación se llevó a cabo de manera exitosa. Los resultados muestran que cada inicio de sesión se completó correctamente, mostrando la pantalla correspondiente al usuario autenticado, tal como se ilustra en la figura 74, con el mensaje de "Ingreso exitoso" claramente visible en la pantalla.

Figura 58*Inicio de sesión correcto**Elaborado por: La Autora*

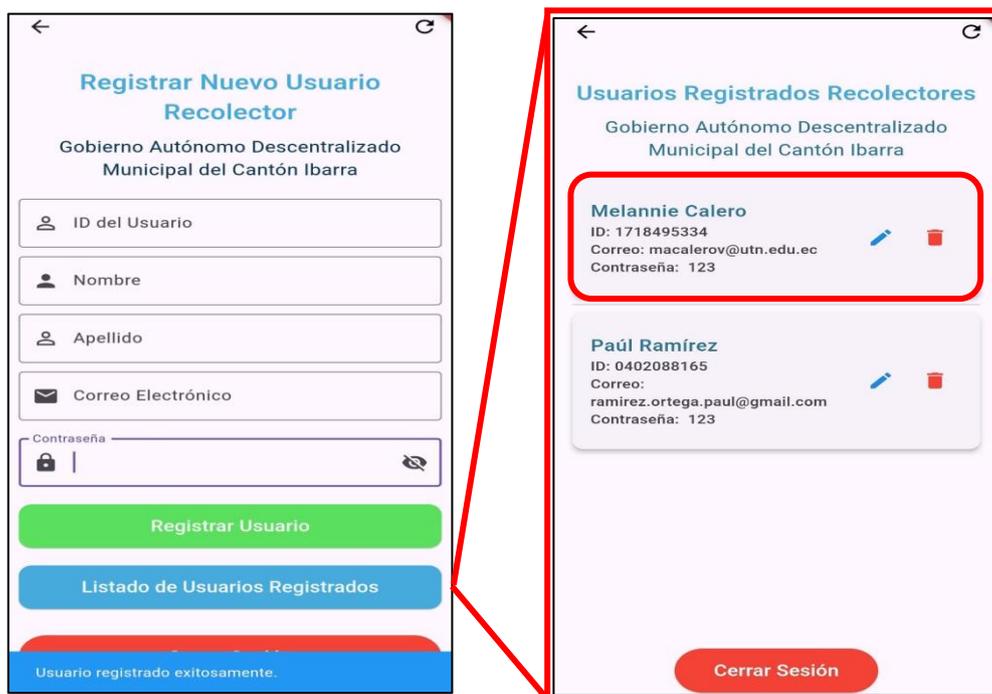
Se han realizado 50 intentos de inicio de sesión incorrecto para validar el sistema de seguridad en casos de credenciales mal ingresadas. En cada intento, el sistema mostró el mensaje indicando que solo se tienen 5 intentos para corregir las credenciales, como se observa en la imagen de la izquierda de la figura 75. Al superar este límite de intentos, el usuario se bloquea automáticamente durante 5 minutos, impidiendo el acceso, incluso si se regresa a la página anterior, se recarga la página o se cierra y vuelve a abrir la aplicación.

Este proceso ha sido probado exhaustivamente para asegurar su efectividad en la protección del sistema contra accesos no autorizados.

Figura 59*Ingreso de sesión incorrecto**Elaborado por: La Autora*

4.1.2. Verificación de Registro, Edición y Eliminación de Usuarios

Se realizaron 53 pruebas de registro para validar el correcto funcionamiento del sistema al añadir nuevos usuarios. En la figura 76, en la pantalla de la izquierda, se ingresan los datos necesarios para registrar un usuario, y al finalizar, se muestra un mensaje de confirmación de registro exitoso. En la pantalla de la derecha, se observa la lista de usuarios registrados, incluyendo sus datos básicos y opciones para editar o eliminar cada registro. Estas pruebas permitieron confirmar que el sistema gestiona de manera efectiva la incorporación y visualización de usuarios en la lista sin errores.

Figura 60*Registro de nuevos usuarios*

Elaborado por: La Autora

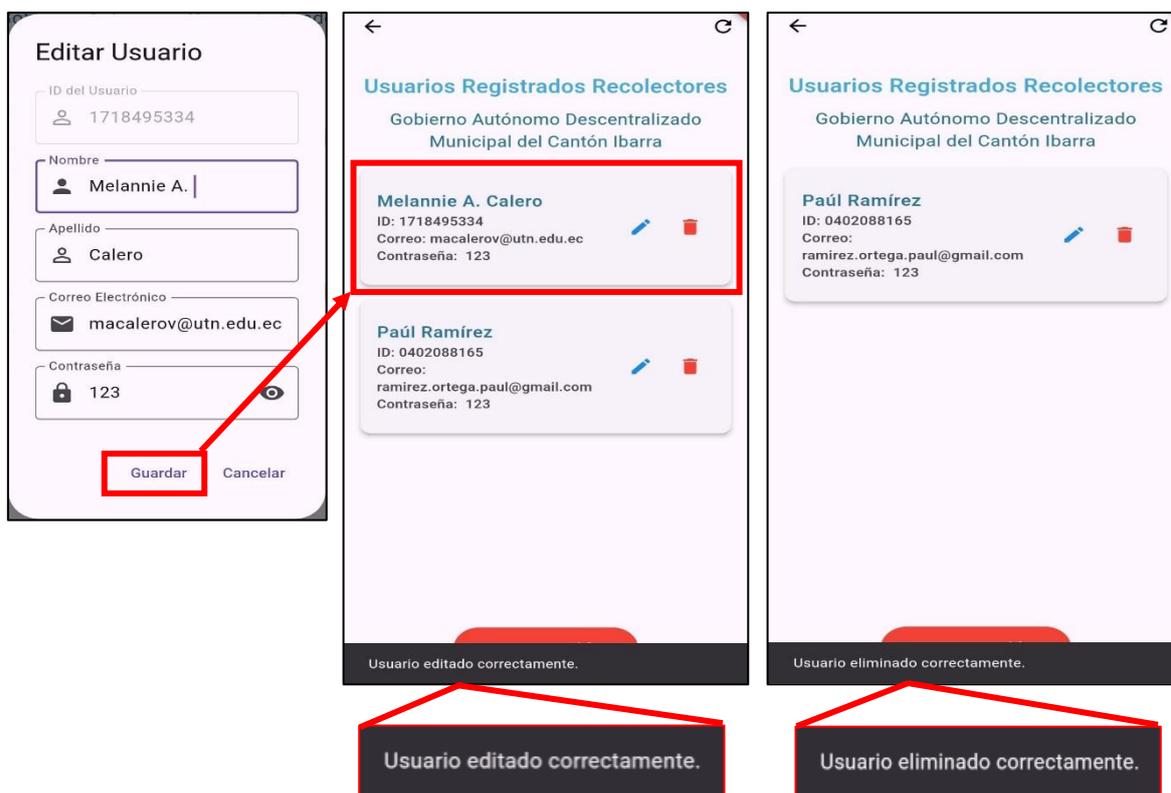
Para validar el funcionamiento correcto en la gestión de usuarios, se llevaron a cabo 53 pruebas de edición y eliminación de registros. En la pantalla de la izquierda de la figura 77, se muestra el formulario de edición donde es posible modificar todos los datos del usuario, a excepción del ID, que permanece inalterable. Al guardar los cambios, el sistema confirma la edición exitosa con un mensaje. En la pantalla del centro, se observa el cambio reflejado en la lista de usuarios. En la pantalla de la derecha, se evidencia la eliminación de un usuario, acompañada de un mensaje de confirmación. Estas pruebas aseguraron que el sistema permite editar y eliminar usuarios sin errores.

Figura 61*Edición y eliminación de usuarios*

Edición de usuarios

Resultado de edición de usuario

Eliminación de usuarios



Elaborado por: La Autora

4.1.3. Verificación de la capacidad del nodo sensor para adaptarse a contenedores de diferentes tamaños

Antes de realizar las mediciones del nivel de llenado de un contenedor, el nodo sensor verifica automáticamente el tamaño total del mismo para garantizar que las mediciones sean precisas y adaptadas al contenedor en uso. Si el tamaño del contenedor ya está almacenado, se recupera directamente de la memoria no volátil del ESP32. Si no es así, el sensor HC-SR04 mide el tamaño por primera vez, un proceso que tarda aproximadamente 5 minutos. Si el resultado es válido, se guarda en la memoria para futuros usos; de lo contrario, se imprime un mensaje de error. La figura 78 muestra 52 pruebas realizadas utilizando diferentes tipos y tamaños de contenedores, confirmando que el nodo sensor se adapta de manera eficiente a cualquier tamaño.

Figura 62*Medición de tamaño del contenedor**Elaborado por: La Autora*

4.2. Evaluación de Logros del Sistema

Se verificarán que las funcionalidades del sistema cumplan con los objetivos propuestos, evaluando su efectividad en la gestión y monitoreo de residuos, el cálculo de rutas óptimas y la generación de notificaciones e informes precisos.

4.2.1. Evaluación del Sistema de Monitoreo de Contenedores de Residuos

Infeciosos

Se evaluó el sistema de monitoreo de residuos infecciosos mediante 58 pruebas de funcionamiento. Una de las funciones más relevantes es el acceso como usuario recolector, cuya pantalla principal, mostrada en la parte izquierda de la figura 79, ofrece una visión general de los niveles de llenado de los contenedores en los distintos laboratorios clínicos. El etiquetado por colores resulta especialmente útil, ya que permite al recolector identificar rápidamente los laboratorios que requieren una recolección

urgente; por ejemplo, los contenedores que han alcanzado su capacidad máxima se resaltan en color rojo. Por otro lado, en la pantalla derecha se presentan datos más detallados sobre cada contenedor, junto con un indicador que confirma si el dispositivo ubicado en el contenedor está funcionando correctamente, es decir, si está conectado a la red Wi-Fi y registrando datos de manera adecuada.

Figura 63

Sistema de monitoreo de contenedores de desechos infecciosos



Elaborado por: La Autora

4.2.2. Validación de Mapas para Cálculo de la Ruta más corta entre Origen y Destino

Para asegurar la precisión en el cálculo de la ruta más corta entre el punto de origen y el destino, se llevaron a cabo 72 pruebas aplicando el algoritmo de Dijkstra. Este algoritmo calcula la ruta más óptima basándose únicamente en la distancia total, permitiendo identificar el camino más corto en términos de longitud entre los dos puntos. Los resultados de todas las pruebas fueron consistentes, mostrando siempre la misma ruta

como la opción óptima, ya que el cálculo se realiza utilizando distancias fijas entre los nodos y no considera factores adicionales como el tráfico o las condiciones de tiempo. Esta metodología se refleja en la imagen de la izquierda de la figura 80, donde se muestra la ruta calculada únicamente en función de la distancia.

Además, se realizó una comparación con las rutas generadas por Google Maps. En la imagen de la derecha de la figura 80 se presentan tres rutas, junto con los tiempos estimados para recorrerlas: la primera ruta toma 7 minutos, la segunda también 7 minutos y la tercera 6 minutos. Esta última ruta, que representa el menor tiempo de recorrido, coincide con la ruta más corta determinada por el algoritmo de Dijkstra, lo que valida su precisión y efectividad en la optimización de trayectos.

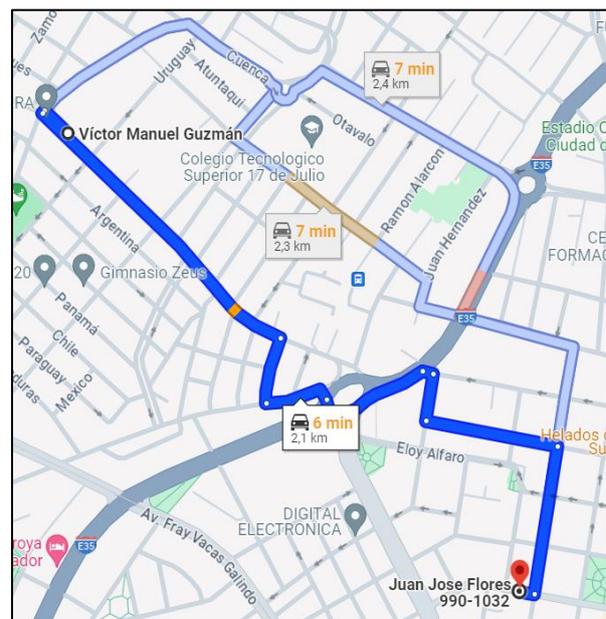
Figura 64

Verificación de la ruta más corta

*Ruta más corta generada en la
Maps
APP Bio Nivel Monitor*



Ruta más corta generada en Google



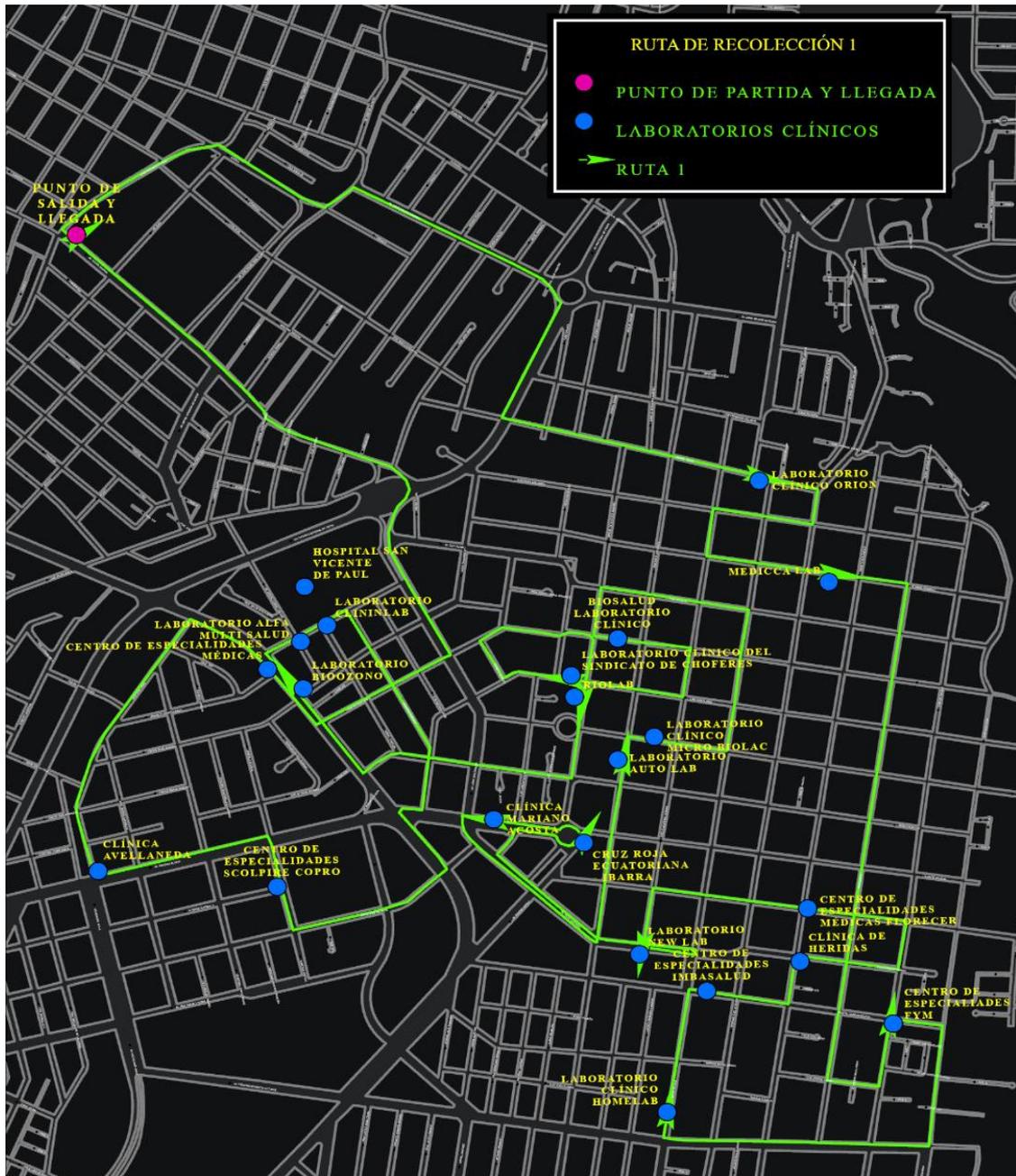
Elaborado por: La Autora

También se realizó una comparativa entre las rutas de recolección del GADM-I presente en la figura 81 y las generadas por el algoritmo de Dijkstra de la app móvil Bio Nivel Monitor. Los resultados evidenciaron que las rutas del GADM-I no son óptimas, ya que no consideran factores como tráfico, clima o distancias. Además, los recolectores suelen visitar laboratorios clínicos distribuidos por la ciudad, incluidos aquellos que se encuentran en los alrededores o son de reciente incorporación, donde con frecuencia encuentran contenedores casi vacíos. Esto genera pérdida de tiempo y recursos, afectando la eficiencia de la recolección en otros centros médicos. Como resultado, algunos laboratorios son atendidos con intervalos de hasta dos semanas y sin horarios fijos.

Es por ello que, el sistema de monitoreo de la app Bio Nivel Monitor se presenta como una solución eficiente para optimizar la gestión operativa. Cabe destacar que la figura 81 no muestra el mapa completo por razones de confidencialidad.

Figura 65

Rutas de recolección 1 – Mapa de rutas utilizado por el GADM-I (Filtrado por laboratorios clínicos)



Elaborado por: La Autora

4.2.3. Verificación del Envío de Notificaciones Personalizadas

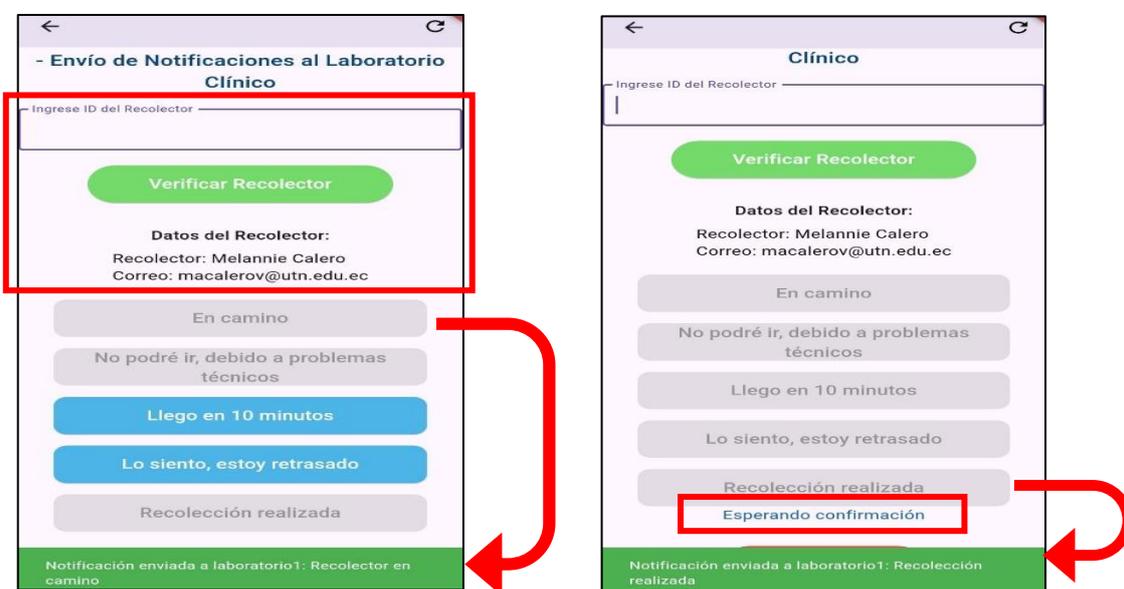
A continuación, se presentan las pruebas realizadas para evaluar el envío de notificaciones, verificando que cada usuario reciba información adecuada según su rol y nivel de acceso.

4.2.3.1. Envío de Notificaciones desde el usuario Recolector hacia el usuario de Laboratorio Clínico

Se llevaron a cabo 65 pruebas para evaluar el envío de notificaciones desde el usuario recolector hacia el usuario del laboratorio clínico. En la pantalla principal del recolector, es necesario ingresar la ID correspondiente para identificar a la persona responsable de la recolección, como se muestra en la imagen izquierda de la figura 82. Una vez identificado, se habilitan los botones para el envío de notificaciones. Al finalizar la recolección, el recolector debe seleccionar la opción “Recolección realizada” para completar el proceso. Esta acción no solo envía la notificación correspondiente, sino que también pone al sistema en estado de espera, aguardando la confirmación del usuario del laboratorio clínico, tal como se muestra en la imagen derecha. Además, en la parte inferior de la pantalla, aparece un mensaje indicando que el envío de cada notificación se realizó exitosamente.

Figura 66

Envío de notificaciones desde el usuario Recolector hacia el usuario de Laboratorio Clínico

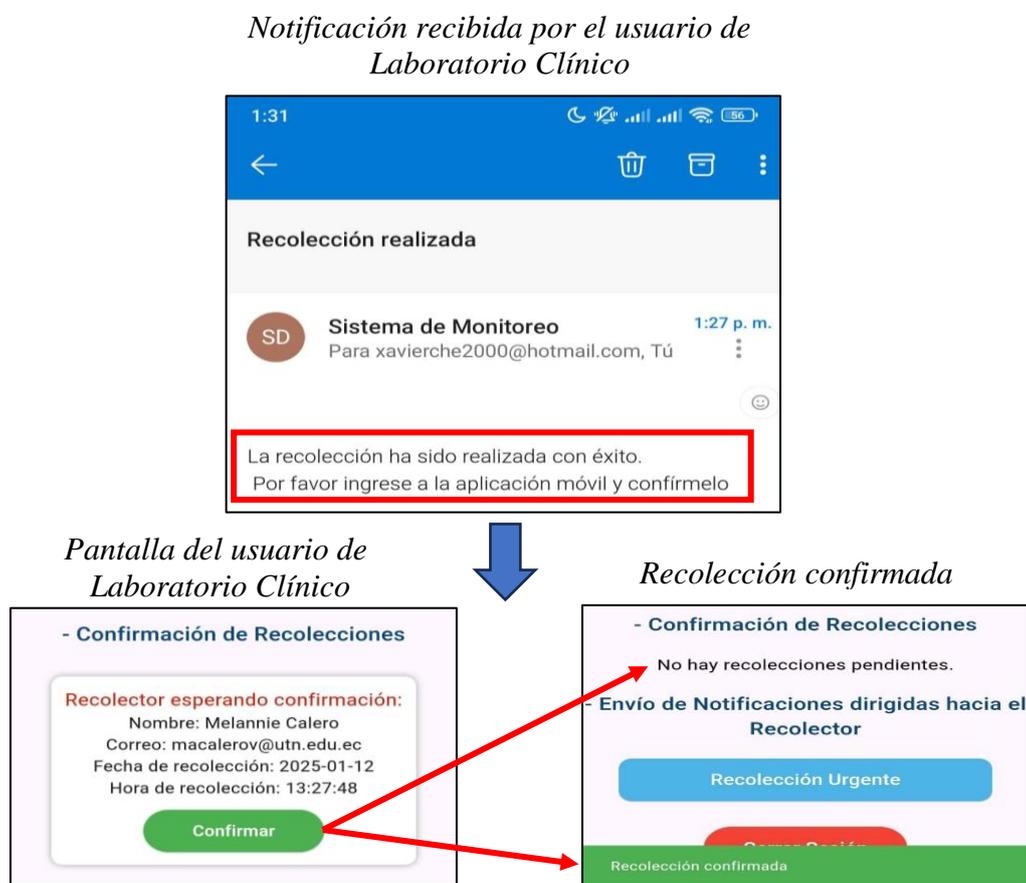


Elaborado por: La Autora

Cuando se envía la notificación de “Recolección realizada,” el usuario del laboratorio clínico recibe un aviso, como se muestra en la imagen superior de la figura 83, indicando que debe ingresar a la app para confirmar la recolección. Al acceder a la aplicación, el usuario del laboratorio clínico puede visualizar una notificación donde se informa que el recolector está a la espera de la confirmación, tal como se observa en la imagen inferior izquierda. Es importante destacar que esta notificación de confirmación solo está disponible durante un periodo de 10 minutos. Si no se confirma dentro del tiempo establecido, el sistema registra automáticamente el evento en la base de datos como "no confirmado."

Figura 67

Envío de notificaciones Recolector - Laboratorio Clínico



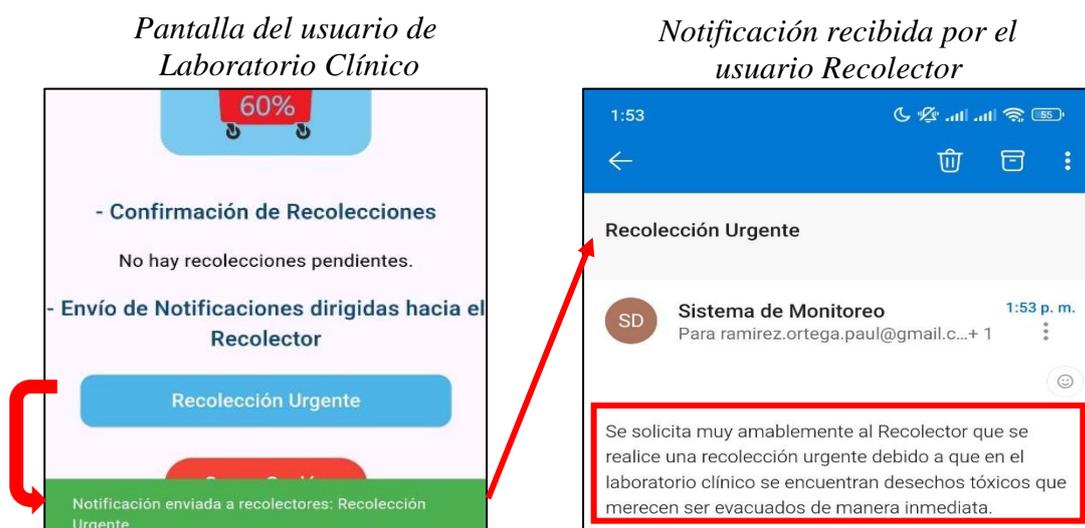
Elaborado por: La Autora

4.2.3.2. *Envío de Notificaciones desde el usuario de Laboratorio Clínico hacia el usuario Recolector*

Se llevaron a cabo 74 pruebas para evaluar el envío de notificaciones desde los usuarios de laboratorios clínicos hacia los recolectores. En estas pruebas, el usuario del laboratorio clínico accede a la app y, desde la sección de envío de notificaciones (visible en la pantalla izquierda de la figura 84), tiene la opción de enviar una solicitud de recolección urgente. Esta funcionalidad está diseñada para casos excepcionales en los que el laboratorio clínico necesita retirar desechos médicos con carácter de urgencia. En la pantalla derecha de la figura 84, se muestra cómo los usuarios recolectores reciben la notificación, acompañada de un mensaje que detalla la solicitud de retiro urgente.

Figura 68

Envío de notificaciones desde el usuario de Laboratorio Clínico hacia el usuario Recolector



Elaborado por: La Autora

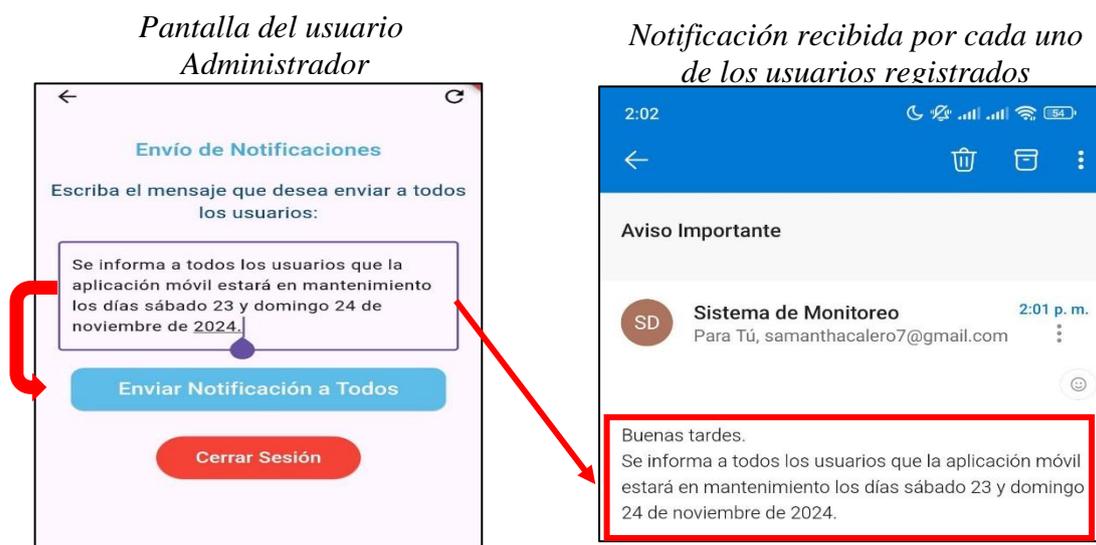
4.2.3.3. *Envío de Notificaciones desde el usuario Administrador hacia todos los usuarios*

Se realizaron 66 pruebas exitosas para verificar el correcto funcionamiento del envío de notificaciones desde el usuario administrador hacia todos los demás usuarios registrados. Durante las pruebas, el administrador accede a la app y, desde el área de envío de notificaciones, puede redactar un mensaje personalizado. Una vez que se selecciona el botón de "Enviar," el mensaje es enviado automáticamente a todos los usuarios registrados, como se muestra en la pantalla izquierda de la figura 85.

En la pantalla derecha, se puede observar cómo los usuarios de laboratorio clínico, recolectores y otros administradores reciben la notificación, lo que demuestra una comunicación efectiva.

Figura 69

Envío de notificaciones desde el usuario Administrador hacia todos los usuarios registrados



Elaborado por: La Autora

4.2.4. Evaluación de Informes de los Contenedores

Las pruebas del funcionamiento del sistema se realizaron desde el día 28 de noviembre hasta el día 30 de diciembre del 2024, en el Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura ubicado entre las calles Juan José Flores y Manuel de la Chica Narváez.

Con el objetivo de validar los resultados obtenidos a través de la aplicación móvil, se realizó una encuesta al personal del Laboratorio Clínico. Este ejercicio permitió contrastar la percepción del personal con los datos generados automáticamente por la herramienta, evaluando su precisión y utilidad.

4.2.4.1. Encuesta realizada por el personal del Laboratorio Clínico

La encuesta fue aplicada de forma presencial a cuatro miembros del personal del laboratorio clínico, entre los cuales se encontraban un doctor médico general, un doctor fisioterapeuta, un auxiliar de laboratorio clínico y un auxiliar de aseo.

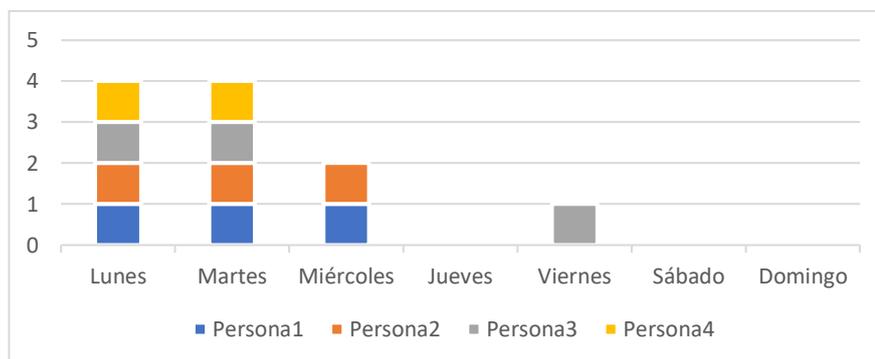
Se formularon un total de veintiún preguntas relacionadas con los días de mayor afluencia de pacientes y los patrones de llenado de los contenedores. Las respuestas se recopilaron en formato físico y posteriormente se analizaron.

Resultados de la Encuesta

Los resultados obtenidos de las encuestas realizadas se encuentran detallados en el **ANEXO 2**. Entre las preguntas más relevantes destacan aquellas relacionadas con los días de mayor generación de desechos infecciosos. Según las respuestas, las 4 personas encuestadas coincidieron en que los días lunes y martes son los de mayor acumulación de este tipo de desechos, lo cual se atribuye al mayor número de pacientes que asisten durante esos días, como se observa en la figura 86.

Figura 70

Resultados de las preguntas realizadas sobre los días de mayor generación de desechos infecciosos.

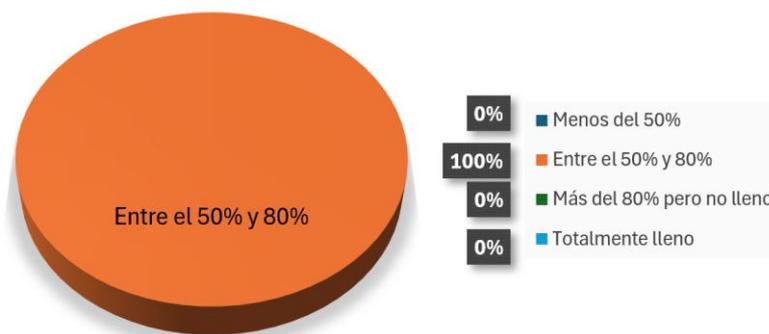


Elaborado por: La Autora

De acuerdo con las encuestas realizadas, al momento de la recolección, los contenedores suelen estar llenos entre un 50% y un 80% como se observa en la figura 87. Sin embargo, todos los participantes destacaron la importancia de incrementar la frecuencia de las recolecciones. Esto permitiría evitar la acumulación excesiva de residuos, reducir posibles problemas relacionados con olores desagradables y minimizar riesgos para la salud, especialmente en los días en que hay un mayor número de pacientes y, por ende, un aumento en la generación de desechos.

Figura 71

Resultados de las preguntas realizadas (nivel de llenado de los contenedores)



Elaborado por: La Autora

4.2.4.2. Análisis comparativo con los Informes obtenidos en la APP

En la figura 88 se presenta una gráfica estadística basada en los datos del contenedor 1. En color rojo se destacan los días en que el contenedor fue vaciado, indicando la recolección de desechos infecciosos. Por otro lado, los días en los que el contenedor sigue llenándose están marcados en color azul. La gráfica también muestra que el nivel de llenado permanece constante durante tres días seguidos, lo cual ocurre porque los sábados y domingos el laboratorio clínico está cerrado, manteniendo el nivel alcanzado el viernes. Además, se observan siete días con un nivel constante del 80%, lo que indica que el contenedor alcanzó su capacidad máxima, activándose el llenado del contenedor 2.

En la pantalla derecha se identifican los lunes y martes como los días de mayor llenado, lo que coincide con las variaciones más notorias en la gráfica de la izquierda y con los resultados de la encuesta mostrados en la figura 86, donde todos los encuestados indicaron que estos días son los de mayor generación de desechos. También se presentan los ciclos de llenado y vaciado del contenedor, con el cálculo del tiempo necesario para llenarse en cada ciclo. A partir de estos datos, se determina que el tiempo promedio para llenar el contenedor es de 6 días, 6 horas y 48 minutos.

lleva a la conclusión de que el laboratorio clínico requiere un total aproximado de 12 días para que ambos contenedores alcancen su capacidad máxima. Este dato es relevante, ya que los recolectores realizan la próxima recolección en intervalos de dos semanas, aunque sin un horario fijo de llegada.

Figura 73

Informes de los datos tomados en el Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura – Contenedor 2



Elaborado por: La Autora

4.3. Discusión de resultados

En este apartado se analizan los resultados obtenidos durante las pruebas realizadas al sistema de monitoreo propuesto, evaluando su desempeño en relación con los objetivos planteados. Las pruebas, desarrolladas en un entorno real en el Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura, permitieron validar la funcionalidad y efectividad del sistema en diversas condiciones operativas.

La implementación del sistema de monitoreo de residuos infecciosos evidenció mejoras significativas en comparación con el sistema actual de recolección utilizado por el GADM-I en los laboratorios clínicos de Ibarra. En el sistema actual, las recolecciones se realizan aproximadamente cada dos semanas, pero los recolectores no cuentan con horarios fijos para su llegada debido a problemas como el tráfico y los tiempos prolongados de desplazamiento. Esto genera acumulaciones de desechos peligrosos cuando las recolecciones se retrasan o recogidas innecesarias cuando los contenedores no han alcanzado su capacidad máxima.

Con el sistema propuesto, es posible que mediante la aplicación móvil “Bio Nivel Monitor”, se pueda conocer el nivel de llenado de los contenedores en tiempo real y presentar esta información al recolector mediante un sistema visual basado en colores, lo que facilita la identificación rápida de los laboratorios cuyos contenedores están próximos a alcanzar su capacidad máxima. Esto no solo optimiza la frecuencia de las recolecciones, sino que también permite avisar mediante notificaciones al personal del laboratorio antes de su llegada, asegurando que estén preparados para la entrega de los residuos. Además, el sistema permite a los laboratorios monitorear el nivel de llenado de sus propios contenedores, dándoles la capacidad de notificar a los recolectores cuando requieran de una recolección urgente, mejorando significativamente la coordinación entre ambas partes.

Además, la aplicación móvil integra el algoritmo de Dijkstra para calcular rutas más eficientes. En el sistema actual, las rutas tomadas por los recolectores no son siempre óptimas debido a problemas de tráfico, lo que dificulta acudir en fechas precisas y provoca demoras en la recolección. Con el uso del sistema, los recolectores pueden identificar las trayectorias más cortas y rápidas, optimizando los tiempos de traslado y mejorando

considerablemente la planificación operativa, lo que resulta en una gestión más eficiente y efectiva de los residuos.

Por último, la generación de informes de los contenedores en los laboratorios clínicos permitió identificar patrones en los niveles de llenado, lo que ayudó a determinar los días de mayor acumulación de residuos y validar los ciclos de recolección. Estos datos también facilitaron el cálculo de un tiempo promedio de llenado, permitiendo predecir con mayor precisión los intervalos necesarios para futuras recolecciones y optimizar la planificación operativa. Además, los informes generados por los recolectores permitieron llevar un registro organizado de cada recolección realizada, ya que, al finalizar el proceso, el laboratorio clínico debía confirmar la recolección a través del sistema. Esto aseguró un control eficiente y transparente de las operaciones. Estos resultados consolidan la efectividad del sistema propuesto como una solución integral para la gestión de residuos infecciosos

Conclusiones y recomendaciones

Conclusiones

- El análisis de referencias bibliográficas permitió comprender el proceso de recolección de desechos infecciosos en Ibarra, evidenciando deficiencias como la falta de monitoreo en tiempo real y planificación de rutas. Esto permitió identificar oportunidades para proponer soluciones tecnológicas que optimicen la gestión y reduzcan riesgos.
- Se comprobó que las rutas generadas por el algoritmo de Dijkstra son más eficientes que las utilizadas por el GADM-I, evidenciando fallas en el sistema actual. Este algoritmo, al calcular las rutas más cortas de manera precisa y consistente, permitió optimizar los mapas utilizados, reduciendo desplazamientos innecesarios, mejorando la planificación y maximizando el uso de tiempo y recursos en el proceso de recolección.
- El sistema permite supervisar constantemente los niveles de llenado de los contenedores de desechos infecciosos y enviar notificaciones personalizadas a cada tipo de usuario cuando se alcanzan niveles críticos, lo que mejora significativamente la eficiencia del proceso al reducir tiempos y facilitar la gestión de recursos disponibles.
- Las pruebas validaron el funcionamiento del sistema, demostrando su capacidad para medir con precisión los niveles de llenado. La comparación entre entrevistas al personal y datos del monitoreo permitió estimar tiempos de llenado, clave para anticipar recolecciones y optimizar la planificación operativa.
- En los laboratorios clínicos no existe un contenedor de desechos estándar, ya que la cantidad de desechos generados varía y requiere contenedores de diferentes tamaños. Sin embargo, esto no representa un problema, ya que el sistema de

monitoreo desarrollado se adapta de manera eficiente a cualquier tipo de contenedor, independientemente de su tamaño, garantizando un control preciso y una gestión óptima de los desechos.

Recomendaciones

- Se recomienda al Municipio de San Miguel de Ibarra implementar sistemas tecnológicos como sensores inalámbricos y aplicaciones móviles para monitorear los niveles de llenado de los contenedores. Esto mejorará la seguridad en la gestión de residuos infecciosos, reduciendo riesgos de contaminación y protegiendo la salud pública y de los trabajadores.
- En la aplicación móvil ya se ha implementado el algoritmo de Dijkstra para calcular las rutas más cortas, basándose únicamente en las distancias. Sin embargo, este algoritmo no considera factores como el tráfico y los tiempos de desplazamiento. Por lo tanto, se recomienda en futuras versiones de la aplicación implementar algoritmos adicionales que incluyan estos factores, lo que permitirá una optimización más precisa y eficiente de las rutas de recolección.
- Se recomienda, en caso de implementar el sistema en contextos reales, utilizar baterías de mayor duración que no requieran un cambio frecuente, asegurando así la continuidad operativa del sistema. Además, se sugiere capacitar al personal encargado tanto del manejo de la aplicación móvil como de la instalación y colocación de los nodos sensores, para garantizar una correcta operación y aprovechamiento del sistema, optimizando su rendimiento y fiabilidad en el tiempo.
- Basándose en los resultados obtenidos durante las pruebas del sistema, se sugiere continuar recolectando y analizando datos históricos sobre los días y horarios de mayor generación de residuos. Esta información permitirá ajustar las frecuencias

de recolección y asignar recursos de manera más eficiente, logrando una planificación operativa más precisa y adaptada a las necesidades reales.

- Se recomienda a las entidades de salud implementar un tamaño estándar de contenedor para los laboratorios clínicos. Esto facilitaría el cálculo de un promedio de los tiempos que cada laboratorio tarda en llenarlos, permitiendo realizar análisis más precisos y optimizar aún más la planificación y gestión del sistema de recolección.
- Se recomienda tomar como base este proyecto para implementarlo en sistemas más complejos, como el monitoreo de tanques de agua, silos de granos o depósitos de combustible, donde el control de niveles en tiempo real resulta esencial para garantizar una gestión eficiente y prevenir desbordes, desperdicios o interrupciones en el suministro.

Referencias Bibliográficas

- ACCESS. (04 de Enero de 2024). *Calendario Estadístico 2023*. Obtenido de Permisos de Funcionamiento Emitidos a Establecimientos de Salud 2016 - 2024:
<https://app.powerbi.com/view?r=eyJrIjoiYWl0YjhmY2YtZDVmZS00ZDVhLTk2ODgtZDEzNmQwZjhlYWFiIiwidCI6IjNmOTg3ODc0LTZjNmEtNDQyYy1hMGY0LTRmMzMzMjM0YTU5YiJ9>
- ACCESS. (Febrero de 2024). *Establecimientos de Salud Acreditados/Certificados*.
 Obtenido de <http://www.aces.gov.ec/certificaciones-acreditaciones/>
- Agency. (2018). *Metodologías de gestión de proyectos - Modelo Waterfall o en cascada*. Zemsania.
- Altamirano, F. S. (2018). Uso de las redes de sensores inalámbricas, basadas con la tecnología . *Polo del Conocimiento*, 300-301.
- Alvarado, J. (23 de Noviembre de 2020). ¡Harta capacitación en manejo de desechos hospitalarios! - EXTRA.ec. Quito, Pichincha, Sierra.
- Andramuño, C. D. (2019). *Generación interna de los residuos y desechos generados en los establecimientos de salud*. Quito, Ecuador: Ministerio de Salud Pública.
- Ávila, A. M. (2022). *Desarrollo de APPS Móviles con Flutter*. Quito, Ecuador: Universidad Hemisferios.
- Ayala. (02 de Noviembre de 2022). *GADM Ibarra*. Obtenido de Noticias:
<https://www.ibarra.gob.ec/site/2022/11/ibarra-mas-que-un-destino-una-experiencia/>

- Baltazar, C. P. (2020). *DESARROLLO DE UNA APLICACIÓN HÍBRIDA E-COMMERCE PARA LA GESTIÓN DE VENTAS DE LA EMPRESA "CALZADO ANABEL"*. Ambato, Ecuador: Universidad Técnica de Ambato.
- Cajas, C. (2023). *Implementación un prototipo de una red lora con sensores para el control medioambiental en el laboratorio de sistemas de comunicación de la carrera de Tecnología Superior en Redes y Telecomunicaciones de la Universidad de las Fuerzas Armadas ESPE*. Latacunga, Ecuador: Universidad de las Fuerzas Armadas.
- Cajas, K. (2022). *Redes de sensores inalámbricos para IoT : automatización de redes inalámbricas de sensores*. Quito: Escuela Politécnica Nacional.
- Cando, C. (Octubre de 2023). *INEC - Buenas cifras, mejores vidas*. Obtenido de Boletín Técnico, Módulo de Desechos Sanitarios en Establecimientos de Salud: https://www.ecuadorencifras.gob.ec/documentos/web-inec/Encuestas_Ambientales/Establecimientos_Salud-Residuos_Peligrosos/2020/20231025_Documento_tecnico_RAS_2020_VF_RC_B_03.pdf
- Castillo, A. (30 de septiembre de 2020). *mobivery - We are touchnology*. Obtenido de Desarrollo de apps en Flutter: descubre sus ventajas: <https://mobivery.com/desarrollo-de-apps-en-flutter-descubre-sus-ventajas/>
- Castillo, L. (2023). *REDES DE COMUNICACIÓN IoT EN CIUDADES INTELIGENTES*. Esmeraldas: Pontificia Universidad Católica del Ecuador.
- Chávez, K. R. (2020). *Diseño de una red inalámbrica de sensado remoto con aplicación en un escenario de selva baja*. Lima, Perú: Pontificia Universidad Católica del Perú.

Chuquimarca, C. T. (2022). Análisis comparativo entre arquitecturas de sistemas IoT .

RITI Journal, Vol. 10, 21, 43-44.

Daquilema, A. D. (2021). *ANÁLISIS DE BASES DE DATOS RELACIONALES Y NO*

RELACIONALES APLICADO AL PROBLEMA DE LA RUTA MÁS CORTA.

Riobamba, Ecuador: Universidad Nacional de Chimborazo.

Del Pozo Barrezueta, H. (17 de Julio de 2020). *Registro Oficial NO. 248 - Órgano de la*

República del Ecuador. Obtenido de REGISTRO OFICIAL NO. 248:

<https://www.registroficial.gob.ec/index.php/registro-oficial->

[web/publicaciones/registro-oficial/item/13201-registro-oficial-no-248](https://www.registroficial.gob.ec/index.php/registro-oficial-)

Delgado, L. d. (2020). Modelos de Desarrollo de Software. *Revista Cubana de Ciencias*

Informáticas, 46-47.

Domínguez, A. C. (2016). *Diseño e implementación de una arquitectura IoT.* Sevilla:

Dep. Ingeniería Telemática - Escuela Técnica Superior de Ingeniería.

Ecuador, C. d. (2019). *Reglamento gestión de desechos generados en establecimientos*

de salud. Acuerdo Ministerial 323 - Registro Oficial 450 de 20-mar.-2019-

Estado: Vigente.

España, N. A. (2023). *GESTIÓN INTEGRAL DE LOS RESIDUOS SÓLIDOS*

PELIGROSOS HOSPITALARIOS EN LA CIUDAD DE IBARRA. Ibarra:

Universidad Técnica del Norte.

Espinoza, G. (2022). *INSTRUCTIVO DE MANEJO ADECUADO DE DESECHOS*

INFECIOSOS. Riobamba, Ecuador: Universidad Nacional de Chimborazo.

- Gabriela, M. N. (2023). *DISEÑO DEL SISTEMA DE GESTIÓN AMBIENTAL A TRAVÉS DE LA NORMA ISO 14001:2015 PARA EL LABORATORIO CLÍNICO AJAVÍ*. Ibarra, Ecuador: Universidad Técnica del Norte.
- GADM-I. (2019). *Ordenanza Sustitutiva a la Ordenanza que regula la gestión de los desechos sanitarios peligrosos en el cantón Ibarra*. Ibarra: Administración 2019-2023.
- GADM-I. (2023). *Actualización del Plan de Desarrollo y Ordenamiento Territorial del Cantón Ibarra 2015-2023*. Ibarra, Ecuador: Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra.
- Gallego, G. (23 de noviembre de 2023). *AZ adsl zone*. Obtenido de <https://www.adslzone.net/reportajes/tecnologia/que-es-wifi-como-funciona/#370530-ieee-802-11ac>
- García, F. (2019). *Manejo Eficiente de Desechos Peligrosos Hospitalarios Generados en el Centro de Salud Velasco Ibarra Tipo C*. Machala, Ecuador: Universidad Técnica de Machala.
- García, M., Molina, H. D., & Avarado, J. L. (2020). *Internet de las Cosas*. México: TEPEXI Boletín Científico de la Escuela Superior Tepeji del Río.
- González, R. (2017). Eficacia del manejo de los residuos en el Hospital San Vicente. *UNIANDES EPISTEME: Revista de Ciencia, Tecnología e Innovación*. ISSN 1390-9150, 3-7.
- Granda, N. (2022). *Caracterización de los residuos sólidos urbanos para establecer alternativas de manejo en la parroquia*. Ibarra - Ecuador: Universidad Técnica del Norte.

- Guamán, J. (2016). *Diseño de una red de sensores para la detección y protección de incendios forestales*. Loja, Ecuador, EC110150.: MASKANA, CEDIA.
- Hurtado, F. L. (2019). *UNA RED DE SENSORES INALÁMBRICA BASADA EN HARDWARE Y SOFTWARE ABIERTO PARA COLABORACIÓN CON APLICACIONES DE ROBÓTICA A MISIONES DE BÚSQUEDA Y RESCATE*. Málaga: Universidad de Málaga.
- Incinerox. (07 de Enero de 2024). *GESTIÓN DE RESIDUOS SANITARIOS*. Obtenido de GESTIÓN Y MANEJO DE DESECHOS HOSPITALARIOS:
<https://incinerox.com.ec/servicios/residuos-sanitarios/>
- INEC. (Octubre de 2023). *INEC - Buenas Cifras, Mejores Vidas*. Obtenido de Permisos de Funcionamiento Emitidos a Establecimientos de Salud 2016-2024:
<https://app.powerbi.com/view?r=eyJrIjoiYWl0YjhmY2YtZDVmZS00ZDVhLTk2ODgtZDEzNmQwZjhlYWFiIiwidCI6IjNmOTg3ODc0LTZjNmEtNDQyYy1hMGY0LTRmMzMzMjM0YTU5YiJ9>
- istecDigital. (05 de enero de 2022). *istec. IMPULSANT CONNECTIONS*. Obtenido de GENERALITAT VALENCIANA: <https://www.istecdigital.es/internet-of-things-iot/>
- Landa, M. P. (2021). *Modelo matemático de optimización para planeación de redes FTTH a través de programación estocástica multietapa*. Ambato, Ecuador: UNIVERSIDAD TÉCNICA DE AMBATO.
- Landy, S. C. (2021). *CARACTERIZACIÓN DE RESIDUOS SÓLIDOS HOSPITALARIOS Y DISEÑO DE UN PLAN DE MANEJO EN EL HOSPITAL SAN JUAN DE DIOS DE LA CIUDAD DE CUENCA*. Cuenca, Ecuador: Universidad Politécnica Salesiana Sede Cuenca.

- Manzo, J. C. (2021). *ANÁLISIS DE LAS VULNERABILIDADES DE LAS REDES INALÁMBRICAS DE RG INTELLIGENT DE LA PARROQUIA LA UNIÓN, PARA MEJORAR LA DISPONIBILIDAD DEL SERVICIO*. Babahoyo: UNIVERSIDAD TÉCNICA DE BABAHOYO.
- Martínez, R. F. (2009). *Redes inalámbricas de sensores: Teoría y Aplicación Práctica*. La Rioja, España: Integrantes del Grupo de Investigación EDMANS, Universidad de la Rioja.
- Mayorga, C. D. (2019). *Diseño e Imlementación de un Sistema de Adquisición de Video y Medición de Distancia Durante el Retroceso de un vehículo en un Dispositivo Móvil*. Quito, Ecuador: Universida Politécnica Nacional.
- Mendoza, A. R. (2022). *Elementos de redes para sistemas de comunicación inalámbrica*. Chetumal Quintana ROO, Mexico: Universidad Autónoma del Estado de Quintana ROO.
- Miriam, H. (24 de junio de 2021). *Google Developers*. Obtenido de <https://developers-latam.googleblog.com/2021/06/novedades-de-flutter-22.html>
- MSP. (2019). *Gestión Interna de los Residuos y Desechos Generados en los Establecimientos de Salud*. Quito, Ecuador.
- Muñoz, A. M. (2019). *REVISIÓN SISTEMÁTICA DEL USO DE LOS DIFERENTES PROTOCOLOS DE INTERCONEXIÓN PARA INTERNET DE LAS COSAS (IoT)*. Bogotá, Colombia: Universidad Católica de Colombia.
- Oliver, S. G. (2016). *Programación Android con Android Studio*. Quito, Ecuador: safeCreative.

- Ortega, M. P. (11 de marzo de 2019). Funcionamiento, representación y comportamiento de diferentes algoritmos frente al cálculo de un small world en ciencia. *IX Congreso ISKO-España*, págs. 6-7.
- Ortiz, M. G. (2020). *Desarrollo de una red de sensores inalámbricos utilizando tecnología LoRa para el monitoreo de un sistema*. Quito: Universidad Politécnica Salesiana - Sede Quito.
- Pupiales, N. (2023). *Gestión integral de los residuos sólidos peligrosos hospitalarios en la ciudad de Ibarra*. Ibarra - Ecuador: Universidad Técnica del Norte.
- Rivera, A. (2023). *Estudio de Impacto Ambiental ex-post y Plan de Manejo Ambiental*. Ibarra, Ecuador: Novaclínica Moderna "CLIMODER S.A..".
- Romero, V. (2017). *La Tecnología ZigBee estudio de las*. Pereira, Colombia: Scientia Et Technica, ISSN: 0122-1701.
- Rubio, J. (2020). *Resoluciones ACCESS*. Obtenido de AGENCIA DE ASEGURAMIENTO DE LA CALIDAD DE LOS SERVICIOS DE SALUD Y MEDICINA PREPAGADA –ACCESS-: <http://www.acess.gob.ec/normativa/>
- Ruiz, R. (23 de Septiembre de 2023). Ibarra, la ciudad a la que siempre se vuelve. *La Hora*, págs. 2-3.
- Saavedra González, A. (2018). *PLATAFORMA IOT PARA EL REGISTRO DE DATOS METEOROLÓGICOS Y SU VISUALIZACIÓN MEDIANTE UNA APLICACIÓN MÓVIL*. Quito, Ecuador: Grado en Ingeniería Informática de Gestión y Sistemas de Información.
- Salazar, J. (2019). Redes Inalámbricas. *TECH Pedia*, 7-16.

- Salazar, K. (2018). *Diseño e implementación de una red con sensores inalámbricos (wsn) con un protocolo abierto de comunicación basado en IEEE 802.15.4 (XBEE) para prácticas universitarias*. Guayaquil, Ecuador: Universidad Politécnica Salesiana Sede Guayaquil.
- Saltos, J. (2023). El internet de las cosas. Desafíos para la participación y el aprendizaje infantil. *Saberes del Conocimiento (RECIMUNDO)*, 336-347.
- SENPLADES. (18 de Noviembre de 2013). *DOCPLAYER*. Obtenido de PROYECTO ANALISIS DE VULNERABILIDADES A NIVEL MUNICIPAL PERFIL TERRITORIAL CANTON SAN MIGUEL DE IBARRA:
<https://docplayer.es/98039861-Proyecto-analisis-de-vulnerabilidades-a-nivel-municipal-perfil-territorial-canton-san-miguel-de-ibarra.html>
- SENPLADES. (2021). *Plan de Creación de Oportunidades 2021-2025. Gobierno del encuentro*. Quito: República del Ecuador.
- Sosa, M. G. (2020). *Desarrollo de una Red de Sensores Inalámbricos Utilizando Tecnología Lora Para el Monitoreo de un Sistema*. Quito, Ecuador: Universidad Politécnica Salesiana Sede Quito.
- Talón, E. M. (2016). *Despliegue de las Redes Inalámbricas*. España: Edición 177.
- Tomalo, E. L. (2018). *Diseño, simulación y despliegue de redes inalámbricas de sensores con contiki*. Quito, Ecuador: Escuela Politécnica Superior de Ingeniería y Sistemas de Telecomunicaciones.
- Tuqueres, C. R. (2019). *EVALUACIÓN DE TECNOLOGÍAS INALÁMBRICAS EN REDES DE ÁREA DOMÉSTICA PARA OBTENER LA CURVA*

CARACTERÍSTICA DE CARGA EN EDIFICIOS INTELIGENTES. Quito:
UNIVERSIDAD POLITÉCNICA SALESIANA - SEDE QUITO.

Vargas, N. A. (2020). *DESARROLLO DE UN PROTOTIPO PARA LA
GEOLOCALIZACIÓN Y MONITORIZACIÓN DE FRECUENCIA CARDIACA
DE MASCOTAS EN LA CIUDAD DE QUITO*. Quito, Ecuador: Universidad
Politécnica Salesiana Sede Quito.

Veolia. (14 de Enero de 2024). *Gestión de Residuos*. Obtenido de
<https://www.veolia.com/latamib/es/soluciones/gestion-residuos>

Anexos

ANEXO 1 – Encuesta realizada al personal del Laboratorio Clínico



Encuesta para el Personal Encargado del Manejo de Desechos Infecciosos en el Laboratorio Clínico “Sindicato Provincial de Choferes Profesionales de Imbabura”

Objetivo: Identificar los días de mayor generación de desechos infecciosos, las temporadas de mayor afluencia de pacientes, los niveles de llenado de los contenedores y los detalles relacionados con la recolección, con el fin de optimizar el sistema de monitoreo.

1. Datos Generales

1.1. Nombre (opcional): _____

1.2. Puesto/Cargo: _____

1.3. Tiempo en el puesto:

- Menos de 1 año
- 1-3 años
- Más de 3 años

2. Días de Mayor Generación de Desechos

2.1. Según su experiencia, ¿cuáles días de la semana considera que se generan más desechos infecciosos en el laboratorio? (Seleccione los días que correspondan)

- Lunes
- Martes
- Miércoles
- Jueves
- Viernes
- Sábado
- Domingo

2.2. En los días seleccionados, ¿qué factores cree que influyen en la mayor generación de desechos? (Seleccione las opciones que apliquen)

- Mayor número de pacientes atendidos esos días
- Realización de procedimientos específicos que generan más desechos (ejemplo: análisis sanguíneos, cultivos, etc.)
- Otros: _____

3. Temporadas de Mayor Afluencia

3.1. ¿Existen temporadas específicas del año en las que note un aumento en la cantidad de pacientes?

- Enero - Marzo
- Abril - Junio
- Julio - Septiembre
- Octubre - Diciembre
- No hay variación significativa

3.2. ¿A qué se atribuye este aumento?

- Épocas de enfermedades estacionales (gripe, dengue, etc.)
- Programas de salud pública (campañas de vacunación, etc.)
- Otros: _____

4. Niveles de Llenado de los Contenedores

4.1. Cuando los recolectores recogen los desechos, ¿qué nivel de llenado alcanza normalmente el contenedor?

- Menos del 50%
- Entre el 50% y el 80%
- Más del 80% pero no lleno
- Totalmente lleno

4.2. ¿Cree que los contenedores deberían recolectarse con mayor o menor frecuencia para evitar problemas de llenado?

- Mayor frecuencia
- Menor frecuencia
- La frecuencia actual es adecuada

5. Recolección de Desechos

5.1. ¿Con qué tipo de agencia de recolección de desechos trabaja el laboratorio?

- Pública (municipal)
- Privada
- Ambas
- No estoy seguro/a

5.2. ¿Cuál es el nombre de la agencia que realiza la recolección de desechos?

5.3. ¿Con qué frecuencia se realiza la recolección de los desechos infecciosos?

- Diario
- Semanal
- Quincenal
- Mensual
- Otro: _____

5.4. ¿Ha ocurrido alguna vez que los recolectores no asistan en las fechas establecidas?

- Sí
- No

5.5. Si respondió “Sí”, ¿con qué frecuencia ocurre esta situación?

- Rara vez (1-2 veces al año)
- Ocasionalmente (3-5 veces al año)
- Frecuentemente (más de 5 veces al año)

5.6. ¿Cuáles son las razones más comunes para estas fallas según su conocimiento?

- Problemas técnicos (daño en vehículos, etc.)
- Problemas administrativos (retrasos en programación)
- Factores externos (clima, tráfico, etc.)
- No se informa el motivo
- Otros: _____

6. Registro de Desechos

6.1. ¿El laboratorio lleva un registro del peso de los desechos infecciosos?

- Sí
- No

6.2. Si la respuesta anterior es “Sí”, ¿se registra el peso de los desechos infecciosos en cada recolección?

- Siempre
- Algunas veces
- Nunca

6.3. ¿Quién realiza el registro del peso de los desechos infecciosos?

- Personal del laboratorio clínico
- Personal de la empresa recolectora
- Ambos

7. Conexión a Internet

7.1. ¿El laboratorio clínico cuenta con una red Wi-Fi disponible para su personal?

- Sí
- No
- No estoy seguro/a

7.2. Si la respuesta anterior es “Sí”, ¿se realizan actualizaciones periódicas en el equipo de red para asegurar la estabilidad de la conexión?

- Sí, regularmente
- Sí, de forma ocasional
- No, nunca se actualiza
- No lo sé

7.3. ¿El área de almacenamiento de desechos infecciosos tiene acceso a la red Wi-Fi del laboratorio?

- Sí
- No
- No lo sé

7.4. ¿La señal Wi-Fi es estable y accesible en toda el área de almacenamiento de desechos infecciosos?

- Sí, es estable en toda el área
- Sí, pero con algunas interrupciones
- No, es deficiente en varias zonas
- No lo sé

8. Sugerencias

8.1. ¿Tiene alguna sugerencia sobre cómo optimizar el manejo y monitoreo de los desechos infecciosos?

Evidencias de la encuesta realizada

A continuación, se presenta las evidencias tomadas durante las encuestas realizadas al personal del Laboratorio Clínico del Sindicato Provincial de Choferes Profesionales de Imbabura.

Figura 74

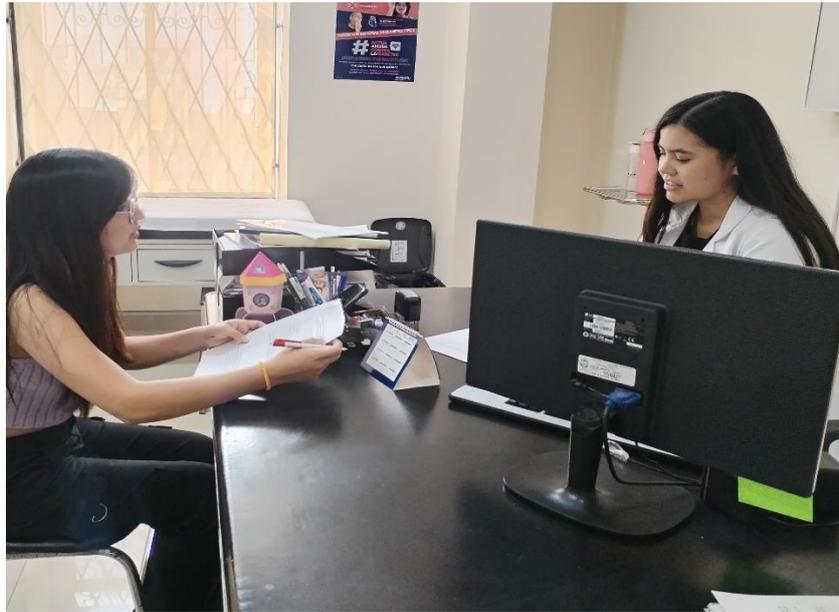
Encuesta realizada a la Fisioterapeuta Johana Chacón Bustamante.



Elaborado por: La Autora

Figura 75

Encuesta realizada a la Dra. Fressia Rosas Médico General.



Elaborado por: La Autora

ANEXO 2 – Resultados obtenidos de la encuesta

La encuesta fue aplicada de forma presencial a cuatro miembros del personal del laboratorio clínico, incluyendo un médico general, un fisioterapeuta, un auxiliar de laboratorio clínico y un auxiliar de limpieza. El cuestionario constó de 21 preguntas divididas en ocho temas clave: datos generales, días de mayor generación de desechos, temporadas de mayor afluencia, niveles de llenado de los contenedores, recolección de desechos, registro de desechos, conexión a internet y sugerencias.

1. Datos Generales

De las cuatro personas encuestadas, dos han trabajado en el laboratorio clínico durante un período de 1 a 3 años, lo que refleja una experiencia intermedia en sus labores. Las otras dos personas tienen más de 3 años de experiencia, lo que les otorga un conocimiento más amplio sobre el funcionamiento del laboratorio y los patrones relacionados con la generación de desechos y la afluencia de pacientes.

2. Días de Mayor Generación de Desechos

La tabla 20 muestra los días identificados por las personas encuestadas como los de mayor afluencia de pacientes y generación de desechos infecciosos en el laboratorio clínico. Todas las personas coincidieron en señalar los días lunes y martes como los de mayor actividad, reflejando un patrón consistente en el inicio de la semana. Adicionalmente, algunas personas también destacaron el miércoles (Persona1 y Persona2) y el viernes (Persona3) como días relevantes, aunque con menor frecuencia. Estos resultados refuerzan que los lunes y martes son los días críticos en términos de manejo de desechos y atención de pacientes.

Tabla 20

Días de mayor afluencia de pacientes (mayor generación de desechos infecciosos)

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
Persona1	x	x	x				
Persona2	x	x	x				
Persona3	x	x			x		
Persona4	x	x					

Nota. En la tabla 20 se menciona las respuestas de los entrevistados donde se menciona que los días de mayor generación de desechos infecciosos son los días lunes y martes.

3. Temporadas de Mayor Afluencia

La tabla 21 muestra las respuestas de las personas encuestadas respecto a las temporadas de mayor afluencia de pacientes en el laboratorio clínico. Todas las personas encuestadas coincidieron en señalar el periodo de julio a septiembre como la temporada de mayor afluencia de pacientes en el laboratorio clínico. Este incremento se atribuye principalmente al aumento en las matrículas de estudiantes que buscan obtener licencias de conducir, lo que genera una mayor demanda de servicios durante estos meses específicos.

Tabla 21

Temporadas de mayor afluencia de pacientes

	Enero - Marzo	Abril - Junio	Julio - Septiembre	Octubre - Diciembre	No hay variación significativa
Persona1			x		
Persona2			x		
Persona3			x		
Persona4			x		

Nota. En la tabla 21 se muestra que todos los encuestados seleccionaron el trimestre de Julio a Septiembre como la temporada con mayor cantidad de pacientes debido a las matrículas estudiantiles.

4. Niveles de Llenado de los Contenedores

De acuerdo con las respuestas de todas las personas encuestadas, cuando los recolectores realizan la recolección de los contenedores, estos suelen tener un nivel de llenado que varía entre el 50% y el 80%. Además, todos coincidieron en que los desechos deberían ser recolectados con mayor frecuencia para evitar la acumulación prolongada de residuos, prevenir la generación de malos olores y minimizar los riesgos sanitarios, especialmente durante los días de mayor afluencia de pacientes.

5. Recolección de Desechos

En la sección de recolección de desechos, los encuestados indicaron que la agencia encargada de este servicio en el laboratorio clínico es el GAD Municipal, el cual realiza la recolección de desechos infecciosos con una frecuencia de 15 días. Además, mencionaron que, aunque en general el servicio se cumple en las fechas establecidas, en raras ocasiones no han podido asistir debido a problemas técnicos, lo que ha generado acumulación de los desechos infecciosos.

6. Registro de Desechos

Con respecto al registro de los desechos, los encuestados destacaron que tanto el laboratorio clínico como la agencia de recolección llevan un registro detallado del peso de los desechos recolectados en cada una de las recolecciones. Este control permite mantener un seguimiento preciso de la cantidad de desechos generados y garantiza la trazabilidad en el manejo adecuado de los residuos infecciosos.

7. Conexión a Internet

En el área de conexión a internet, todos los encuestados mencionaron que el laboratorio clínico cuenta con una red Wi-Fi que cubre todas sus instalaciones, incluyendo el área de almacenamiento de desechos infecciosos, donde se dispone de una conexión

estable. Además, destacaron que se realiza un mantenimiento periódico de los equipos para garantizar el correcto funcionamiento de la red y evitar interrupciones en el servicio.

8. Sugerencias

En la tabla 3 se presentan las sugerencias propuestas por cada uno de los encuestados con respecto al sistema actual de recolección de desechos implementado por la agencia recolectora del GAD Municipal de la ciudad de Ibarra. Estas propuestas reflejan observaciones y recomendaciones para mejorar la frecuencia de recolección y garantizar un manejo más eficiente de los residuos infecciosos, contribuyendo así a un mejor funcionamiento del sistema.

Tabla 22

Sugerencias de las personas entrevistadas

Sugerencias	
Persona1	Recolectar de manera ordenada y correcta.
Persona2	Un monitor que calcule el llenado óptimo para que sea vaciado por la empresa recolectora.
Persona3	Sería bueno que se llevaran los desechos infecciosos cada 8 días.
Persona4	No hay sugerencias.

Nota. La tabla 22 muestra las sugerencias propuestas por las personas entrevistadas con respecto al sistema actual de recolección de la ciudad de Ibarra.

ANEXO 3 – Certificado de Validación de Pruebas del Funcionamiento de la Sistema
Emitido por el Sindicato Provincial de Choferes Profesionales de Imbabura



SINDICATO PROVINCIAL DE CHOFERES PROFESIONALES DE IMBABURA
Fundado el 18 de marzo de 1928
Acuerdo Ejecutivo N° 745
Filial de la Federación de Choferes Profesionales del Ecuador
ESCUELA DE CAPACITACION

Quien suscribe *Fulgo. Carlos Edwin Vásquez M.* en calidad de **SECRETARIO GENERAL DEL SINDICATO DE CHOFERES PROFESIONALES DE IMBABURA** y **DIRECTOR ADMINISTRATIVO DE LA ESCUELA DE CAPACITACIÓN IBARRA**, tengo a bien extender el presente certificado en debida y legal forma.

CERTIFICADO

A petición verbal de la señorita **CALERO VALLEJOS MELANNIE ADRIANA**, portadora de la cédula de ciudadanía No. **1718495334..**

Certifico que la señorita **CALERO VALLEJOS MELANNIE ADRIANA**, realizó las pruebas de funcionamiento del sistema de desarrollo del Proyecto de Tesis titulado.

“SISTEMA DE MONITOREO DE CONTENEDORES DE RESIDUOS INFECCIOSOS CON NOTIFICACIÓN DE RECOLECCIÓN INDICANDO LA MEJOR RUTA, PARA LABORATORIOS CLÍNICOS DE LA CIUDAD DE IBARRA”

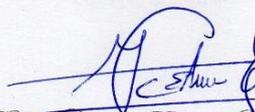
La mencionada señorita, realizó dichas pruebas en las instalaciones del Laboratorio Clínico, perteneciente a nuestra institución desde el día 28 de noviembre hasta el 30 de diciembre 2024.

Certificado que servirá para presentar los requisitos que exige la Universidad.

Es todo cuanto puedo autenticar en honor a la verdad. Autorizo al interesado dar a la presente certificación el uso que más convenga a sus intereses, excepto para trámites legales

Ibarra, 16 de enero 2025

Atentamente
“JUSTICIA Y TRABAJO”




Fulgo. Carlos Edwin Vásquez Mera
SECRETARIO GENERAL S.P.CH.P.I.
DIRECTOR ADMINISTRATIVO E.C

Flores N°. 10-12 y Chica Narváez Teléfonos: 099 368 6550 Ibarra – Ecuador
www.choferesimbabura.com / escuelaschpi@hotmail.com / sichopimb@hotmail.com / Facebook: Schopro Ibarra

**ANEXO 4 – Certificado de Validación del Sistema de Monitoreo tras Verificación
Personal de la Doctora Fressia Rosas del Sindicato Provincial de Choferes
Profesionales de Imbabura**

CERTIFICADO DE FUNCIONAMIENTO DEL SISTEMA

Yo, la Dra. Fressia Rosas Echegaray, Médico General del Sindicato Provincial de Choferes Profesionales de Imbabura, certifico que estuve presente y supervisé personalmente las pruebas realizadas para evaluar el sistema desarrollado por la Srta. Melannie Calero, correspondiente a su proyecto de tesis titulado:

“SISTEMA DE MONITOREO DE CONTENEDORES DE RESIDUOS INFECCIOSOS CON NOTIFICACIÓN DE RECOLECCIÓN INDICANDO LA MEJOR RUTA PARA LABORATORIOS CLÍNICOS DE LA CIUDAD DE IBARRA”

Mi participación incluyó la verificación del correcto funcionamiento del sistema en todos sus aspectos principales, incluyendo:

- Verifiqué la precisión y confiabilidad de los datos generados por los dispositivos ubicados en los contenedores.
- Validé que los datos se mostraran de forma clara y precisa en la plataforma móvil, facilitando decisiones informadas por parte de los usuarios.
- Aseguré el correcto envío y recepción de notificaciones entre los distintos usuarios del sistema.

De esta manera, certifico que el sistema cumple con los estándares requeridos y demuestra ser una herramienta funcional y efectiva para el monitoreo de residuos infecciosos.

En constancia de lo anteriormente expuesto, extendiendo el presente certificado en la ciudad de Ibarra, a los 13 días del mes de enero de 2025.

Atentamente,



SINDICATO PROVINCIAL
DE CHOFERES DE IMBABURA
Dra. Fressia Rosas E.
MÉDICO GENERAL
C.I. 100253281-6

.....
Dra. Fressia Rosas Echegaray

Médico General del Sindicato Provincial de Choferes Profesionales de Imbabura

ANEXO 5 – Código del Nodo Sensor

NodoSensor.ino

```

1  #include <WiFi.h> // Conexión Wi-Fi
2  #include <FirebaseESP32.h> // Interacción con Firebase
3  #include <Preferences.h> // Almacenamiento no volátil
4  #include <time.h> // Manejo de fechas y horas
5
6  // Credenciales de conexión a la red Wi-Fi
7  const char* ssid = "Melannie"; //Nombre red Wi-Fi
8  const char* password = "12345678"; //Contraseña red Wi-Fi
9
10 //Datos de conexión a Firebase: URL de la base de datos y Token de autenticación
11 #define FIREBASE_HOST "https://contenedores-3a130-default-rtdb.firebaseio.com"
12 #define FIREBASE_AUTH "HehHAgRp9o0bbpjyG05hNheujLjmSj21cYBg5Kq7"
13
14 FirebaseData fbdo; // Objeto para manejar datos de Firebase
15 FirebaseAuth auth; // Objeto para la autenticación en Firebase
16 FirebaseConfig config; // Objeto de configuración de Firebase
17
18 // Pines del sensor ultrasónico
19 const int trigPin = 4; // Pin de disparo (Trigger)
20 const int echoPin = 5; // Pin de recepción (Echo)
21
22 // La primera vez que se utiliza el nodo sensor se mide el tamaño del contenedor
23 float tamanoContenedor = 0.0;
24
25 Preferences preferences;
26
27 unsigned long ultimaLectura = 0; // Tiempo de la última lectura
28 const unsigned long intervalo = 5000; // Intervalo de 5 segundos entre lecturas
29
30 const String nombreLaboratorio = "Laboratorio_1"; // Nombre del laboratorio
31 const String nombreContenedor = "Contenedor_1"; // Nombre del contenedor
32
33 unsigned long ultimaVerificacion = 0; // Tiempo de la última verificación
34 const unsigned long intervaloVerificacion = 60000; // Intervalo de 1 minuto (60000 ms)
35
36 bool primerEnvioDistanciaExitoso = false; // Indica si el primer envío de distancia y porcentaje fue exitoso
37 unsigned long tiempoUltimoEnvioDistancia = 0; // Marca de tiempo del último envío de distancia y porcentaje
38 const unsigned long intervaloEnvioDistanciaInicial = 60000; // 1 minuto
39 const unsigned long intervaloEnvioDistanciaRegular = 1800000; // 30 minutos (30 * 60 * 1000)
40
41
42 // Variables para almacenamiento temporal
43 struct DatosTemporales {
44     String marcaTiempo;
45     float distancia;
46     int porcentaje;
47 };
48
49 std::vector<DatosTemporales> bufferDatos;
50
51 //Conexión con red Wi-Fi
52 void conectarWiFi() {
53     WiFi.begin(ssid, password); //Inicialización de red Wi-Fi
54     int intentos = 0;
55     while (WiFi.status() != WL_CONNECTED && intentos < 20) { //Se intenta conectar hasta 20 veces
56         delay(500);
57         Serial.println("Conectando a WiFi...");
58         intentos++;
59     }
60
61     if (WiFi.status() == WL_CONNECTED) {
62         Serial.println("Conectado a WiFi"); //Mensaje de conexión exitosa
63     } else {
64         Serial.println("No se pudo conectar a WiFi"); //Mensaje de conexión inválida
65     }
66 }

```

```

68 void setup() {
69   Serial.begin(115200);
70
71   // Conexión WiFi
72   // Intentar conectar a WiFi
73   conectarWiFi();
74
75   //Inicialización de pines del sensor
76   pinMode(trigPin, OUTPUT);
77   pinMode(echoPin, INPUT);
78
79   // Configuración de Firebase
80   config.host = FIREBASE_HOST;
81   config.signer.tokens.legacy_token = FIREBASE_AUTH;
82   Firebase.begin(&config, &auth);
83
84   if (Firebase.ready()) {
85     Serial.println("Conexión con Firebase activa.");
86   } else {
87     Serial.println("Error en la conexión con Firebase.");
88   }
89
90   // Configuración del servidor NTP para obtener la hora
91   configTime(-5 * 3600, 0, "pool.ntp.org");
92   while (!time(nullptr)) {
93     delay(100);
94     Serial.println("Sincronizando con el servidor NTP...");
95   }
96   Serial.println("Hora sincronizada con NTP");
97
101  // Inicialización de memoria no volátil
102  preferences.begin("contenedor", false);
103
104  // Recuperar tamaño del contenedor o medirlo si no está almacenado
105  tamanoContenedor = preferences.getFloat("tamano", 0.0);
106  if (tamanoContenedor == 0.0) { // Si no está guardado, se mide por primera vez
107    Serial.println("Midiendo tamaño del contenedor por primera vez...");
108    Serial.println("Esto tomará 5 minutos. Por favor espere...");
109    // Contador para 3 minutos (180 segundos)
110    for (int i = 0; i < 300; i++) {
111      delay(1000); // 1 segundo de espera
112      if ((i + 1) % 10 == 0) { // Mostrar mensaje cada 10 segundos
113        Serial.print("Transcurrido: ");
114        Serial.print(i + 1);
115        Serial.println(" segundos...");
116      }
117    }
118    tamanoContenedor = medirTamanoContenedor();
119    if (tamanoContenedor > 0) { // Si la medición es válida, se almacena
120      preferences.putFloat("tamano", tamanoContenedor);
121      Serial.println("Tamaño del contenedor almacenado en memoria.");
122    } else {
123      Serial.println("Error: No se pudo medir el tamaño del contenedor.");
124    }
125  } else {
126    Serial.println("Tamaño del contenedor recuperado de la memoria.");
127  }
128  // Muestra el tamaño total del contenedor
129  Serial.print("Tamaño total del contenedor (cm): ");
130  Serial.println(tamanoContenedor);
131 }
133 void loop() {
134   unsigned long tiempoActual = millis();
135
136   // Verificar conexiones cada minuto
137   if (tiempoActual - ultimaVerificacion >= intervaloVerificacion) {
138     ultimaVerificacion = tiempoActual;
139     verificarConexiones();
140   }

```

```

142 // Cambiar intervalo dinámico basado en si el primer dato se envió correctamente
143 unsigned long intervaloActual = primerEnvioDistanciaExitoso ? 1800000 : 4000;
144
145 // Realizar medición si ha pasado el intervalo correspondiente
146 if (tiempoActual - ultimaLectura >= intervaloActual) {
147     ultimaLectura = tiempoActual;
148
149     Serial.println("Calculando promedio de distancia...");
150     float promediollenado = calcularDistanciaPromedio();
151
152     if (promediollenado > 0) {
153         int porcentajellenado = calcularPorcentajeLlenado(promediollenado);
154         guardarDatosTemporalmente(promediollenado, porcentajellenado);
155
156         // Intentar enviar los datos a Firebase
157         if (WiFi.status() == WL_CONNECTED && Firebase.ready()) {
158             bool envioExitoso = enviarDatosFirebaseConReintentos(promediollenado, porcentajellenado);
159             if (envioExitoso) {
160                 primerEnvioDistanciaExitoso = true; // Marcar como exitoso
161                 Serial.println("Primer envío exitoso de distancia y porcentaje.");
162             } else {
163                 Serial.println("Error al enviar datos de distancia y porcentaje después de varios intentos.");
164             }
165         } else {
166             Serial.println("No hay conexión Wi-Fi. Datos no enviados.");
167         }
168     } else {
169         Serial.println("Datos no válidos para el cálculo.");
170     }
171 }
172
173
174 void verificarConexiones() {
175     // Verifica si está conectado a Wi-Fi
176     if (WiFi.status() == WL_CONNECTED) {
177         Serial.println("Wi-Fi conectado correctamente.");
178     } else {
179         Serial.println("No se pudo conectar a Wi-Fi.");
180         conectarWiFi(); // Intentar reconectar Wi-Fi
181     }
182 }
183
184 // Verifica si Firebase está listo
185 if (Firebase.ready()) {
186     Serial.println("Conexión con Firebase activa.");
187     // Crear un identificador basado en la fecha actual (solo la fecha)
188     time_t now = time(nullptr);
189     struct tm* timeinfo = localtime(&now);
190     char fecha[11]; // Formato YYYY-MM-DD (10 caracteres + nulo)
191     strftime(fecha, sizeof(fecha), "%Y-%m-%d", timeinfo); // Solo la fecha
192     String identificador = String(fecha); // Usar fecha como identificador
193
194     // Crear los datos a enviar
195     String horaActual = String(timeinfo->tm_hour) + ":" + String(timeinfo->tm_min) + ":" + String(timeinfo->tm_sec);
196     String contenedor = "1";
197     String laboratorio = "Laboratorio Clínico 1";
198
199     // Enviar los datos a Firebase
200     if (Firebase.setString(fbdo, "/wifi/" + identificador + "/hora", horaActual) &&
201         Firebase.setString(fbdo, "/wifi/" + identificador + "/contenedor", contenedor) &&
202         Firebase.setString(fbdo, "/wifi/" + identificador + "/laboratorio", laboratorio)) {
203         Serial.println("Datos de wifi enviados correctamente a Firebase.");
204     } else {
205         Serial.println("Error al enviar datos a Firebase: " + fbdo.errorReason());
206     }
207 } else {
208     Serial.println("Error en la conexión con Firebase.");
209 }
210 }

```

```

214 void guardarDatosTemporalmente(float distanciaPromedio, int porcentajellenado) {
215     time_t now = time(nullptr); // Obtiene el tiempo actual
216     struct tm* timeinfo = localtime(&now); // Convierte el tiempo a formato local
217     char buffer[26];
218     strftime(buffer, sizeof(buffer), "%Y-%m-%d %H:%M:%S", timeinfo);
219     String marcaTiempo = String(buffer); // Convierte el formato a una cadena de texto
220     // Crea un registro con la marca de tiempo, distancia y porcentaje
221     DatosTemporales dato = {marcaTiempo, distanciaPromedio, porcentajellenado};
222     bufferDatos.push_back(dato); // Agrega el registro al buffer de datos
223     Serial.println("Datos almacenados temporalmente en el ESP32."); // Mensaje de confirmación
224 }
225
226 void enviarDatosFirebase() {
227     if (WiFi.status() == WL_CONNECTED && Firebase.ready()) { // Verifica conexión Wi-Fi y Firebase
228         for (auto it = bufferDatos.begin(); it != bufferDatos.end(); ) { // Recorre los datos en el buffer
229             String nodoLaboratorio = nombreLaboratorio; // Nodo de Firebase para el laboratorio
230             String nodoContenedor = nombreContenedor; // Nodo de Firebase para el contenedor
231
232             // Envía distancia y porcentaje a Firebase
233             if (Firebase.setFloat(fbdo, "/" + nodoLaboratorio + "/" + nodoContenedor + "/distancias/" + it->marcaTiempo, it->distancia) &&
234                 Firebase.setInt(fbdo, "/" + nodoLaboratorio + "/" + nodoContenedor + "/porcentajes/" + it->marcaTiempo, it->porcentaje)) {
235                 Serial.println("Datos enviados a Firebase: " + it->marcaTiempo); // Confirmación de envío exitoso
236                 it = bufferDatos.erase(it); // Elimina el dato enviado del buffer
237             } else {
238                 Serial.println("Error al enviar datos: " + fbdo.errorReason()); // Muestra error si falla el envío
239                 break; // Sale del bucle en caso de error
240             }
241         }
242     } else {
243         Serial.println("No hay conexión a WiFi. Datos permanecerán en el ESP32."); // Mensaje si no hay conexión con Wi-Fi
244     }
245 }
246
247 float leerDistanciaBruta() {
248     digitalWrite(trigPin, LOW);
249     delayMicroseconds(2);
250     digitalWrite(trigPin, HIGH);
251     delayMicroseconds(10);
252     digitalWrite(trigPin, LOW);
253
254     // Leer duración del pulso en microsegundos
255     long duracion = pulseIn(echoPin, HIGH);
256
257     // Calcular distancia en centímetros
258     float distancia = duracion * 0.0343 / 2;
259
260     // Validar valores extremos
261     if (distancia <= 2 || distancia > 400) { // Rango típico del sensor HC-SR04
262         return -1; // Retornar -1 si la medición no es válida
263     }
264
265     return distancia; // Retorna la distancia válida en cm
266 }
267
268 float medirTamanoContenedor() {
269     Serial.println("Midiendo tamaño total del contenedor...");
270     float tamanoMedido = 0.0;
271     int medicionesValidas = 0;
272
273     for (int i = 0; i < 5; i++) { // Reducido a 5 mediciones para mayor rapidez
274         float distancia = leerDistanciaBruta();
275         if (distancia > 0) { // Solo usar valores válidos
276             tamanoMedido += distancia;
277             medicionesValidas++;
278         }
279         delay(4000); // Reducido el delay entre mediciones
280     }
281
282     if (medicionesValidas > 0) {
283         return tamanoMedido / medicionesValidas; // Retorna el promedio de las mediciones válidas
284     } else {
285         Serial.println("Error: No se obtuvieron mediciones válidas.");
286         return 0; // Devuelve 0 si no se midió nada válido
287     }
288 }

```

```

290 float calcularDistanciaPromedio() {
291     float sumaDistancias = 0.0;
292     int medicionesValidas = 0;
293     Serial.println("Tomando 15 mediciones (calculando nivel de llenado:");
294     for (int i = 0; i < 15; i++) {
295         // Leer distancia bruta desde el sensor
296         float distanciaMedida = leerDistanciaBruta();
297         if (distanciaMedida <= 0) {
298             // Si la medición no es válida, saltarla
299             Serial.print("Medición ");
300             Serial.print(i + 1);
301             Serial.println(": Valor inválido (ignorado).");
302             continue;
303         }
304         // Calcular nivel de llenado
305         float nivelllenado = tamanoContenedor - distanciaMedida;
306         Serial.print("Medición ");
307         Serial.print(i + 1);
308         Serial.print(": Nivel de llenado calculado: ");
309         Serial.println(nivelllenado);
310         // Considerar solo valores positivos para el promedio
311         if (nivelllenado > 0) {
312             sumaDistancias += nivelllenado;
313             medicionesValidas++;
314         }
315         delay(4000); // Esperar 4 segundos entre mediciones
316     }
317     // Calcular promedio si hay mediciones válidas
318     if (medicionesValidas > 0) {
319         float promedio = sumaDistancias / medicionesValidas;
320         Serial.print("Promedio de nivel de llenado: ");
321         Serial.println(promedio);
322         return promedio;
323     } else {
324         Serial.println("No hay datos positivos para calcular el promedio.");
325         return -1; // Retorna -1 si no hubo datos válidos
326     }
327 }

330 int calcularPorcentajelllenado(float promediolllenado) {
331     if (tamanoContenedor <= 0) { // Verifica si el tamaño del contenedor es válido
332         Serial.println("Error: Tamaño del contenedor no válido.");
333         return 0; // Retorna 0 si el tamaño no es válido
334     }
335     // Calcular el tamaño de cada segmento del 10%
336     float segmento = tamanoContenedor / 10.0;
337
338     // Determinar el porcentaje de llenado basado en el promedio
339     int porcentajeLlenado = 0;
340     for (int i = 1; i <= 10; i++) {
341         if (promediolllenado <= segmento * i) {
342             porcentajeLlenado = i * 10; // Asigna el porcentaje correspondiente
343             break; // Detiene el bucle al encontrar el porcentaje
344         }
345     }
346     // Asegurar que el porcentaje esté dentro del rango 0-100
347     porcentajeLlenado = max(0, min(100, porcentajeLlenado));
348     return porcentajeLlenado; // Retorna el porcentaje de llenado
349 }

350
351
352 bool enviarDatosFirebaseConReintentos(float distanciaPromedio, int porcentajelllenado) {
353     String nodoLaboratorio = nombreLaboratorio; // Nodo para el laboratorio en Firebase
354     String nodoContenedor = nombreContenedor; // Nodo para el contenedor en Firebase
355
356     // Genera la marca de tiempo actual
357     time_t now = time(nullptr);
358     struct tm* timeinfo = localtime(&now);
359     char buffer[26];
360     strftime(buffer, sizeof(buffer), "%Y-%m-%d %H:%M:%S", timeinfo);
361     String marcaTiempo = String(buffer);
362
363     Serial.println("Enviando datos de distancia y porcentaje a Firebase con reintentos...");

```

```
365 // Intentar enviar datos hasta 3 veces en caso de fallo
366 const int maxReintentos = 3;
367 for (int intento = 1; intento <= maxReintentos; intento++) {
368   if (Firebase.setFloat(fbdo, "/" + nodoLaboratorio + "/" + nodoContenedor + "/distancias/" + marcaTiempo, distanciaPromedio) &&
369       Firebase.setInt(fbdo, "/" + nodoLaboratorio + "/" + nodoContenedor + "/porcentajes/" + marcaTiempo, porcentajeLlenado)) {
370     Serial.println("Datos enviados correctamente a Firebase.");
371     return true; // Envío exitoso
372   } else {
373     Serial.print("Error al enviar datos a Firebase. Intento ");
374     Serial.print(intento);
375     Serial.print(" de ");
376     Serial.println(maxReintentos);
377     Serial.println("Razón del error: " + fbdo.errorReason());
378   }
379
380   // Pausa entre reintentos
381   delay(2000); // Espera de 2 segundos entre intentos
382 }
383
384 Serial.println("Error: No se pudieron enviar los datos a Firebase después de 3 intentos.");
385 return false; // Envío fallido tras los reintentos
386 }
```