



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**  
**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**OPTIMIZACIÓN DE RUTAS PARA LA RECOLECCIÓN DE DESECHOS SÓLIDOS EN  
LA CIUDAD DE IBARRA PROVINCIA DE IMBABURA**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
SISTEMAS COMPUTACIONALES**

**AUTOR:**

**FAUSTO BOLÍVAR LUCANO CHÁVEZ**

**DIRECTOR:**

**MSC. EDGAR ALBERTO MAYA OLALLA**

**Ibarra - 2019**



# UNIVERSIDAD TÉCNICA DEL NORTE

## BIBLIOTECA UNIVERSITARIA

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

#### 1. IDENTIFICACIÓN DE LA OBRA

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

DATOS DEL CONTACTO	
<b>CÉDULA DE IDENTIDAD:</b>	0401394747
<b>NOMBRES Y APELLIDOS:</b>	FAUSTO BOLÍVAR LUCANO CHÁVEZ
<b>DIRECCIÓN:</b>	MIGUEL SÁNCHEZ Y AV. RICARDO SÁNCHEZ
<b>E-MAIL</b>	fblucanoc@utn.edu.ec
<b>TELÉFONO MÓVIL</b>	0982644719

DATOS DE LA OBRA	
<b>TÍTULO:</b>	OPTIMIZACIÓN DE RUTAS PARA LA RECOLECCIÓN DE DESECHOS SÓLIDOS EN LA CIUDAD DE IBARRA PROVINCIA DE IMBABURA.
<b>AUTOR:</b>	FAUSTO BOLÍVAR LUCANO CHÁVEZ
<b>FECHA:</b>	04/04/2019
<b>PROGRAMA:</b>	<input checked="" type="checkbox"/> <b>PREGRADO</b> <input type="checkbox"/> <b>POSGRADO</b>
<b>TÍTULO POR QUE OPTA:</b>	INGENIERO EN SISTEMAS COMPUTACIONALES
<b>ASESOR</b>	Msc. EDGAR ALBERTO MAYA OLALLA

## **2. CONSTANCIAS**

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

Ibarra, a los 04 días del mes de Abril de 2019

**EL AUTOR:**



Fausto Bolívar Lucano Chávez



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

**CERTIFICACIÓN DIRECTOR DE TESIS**

Por medio del presente yo Msc. Edgar Maya, certifico que el trabajo de grado **“OPTIMIZACIÓN DE RUTAS PARA LA RECOLECCIÓN DE DESECHOS SÓLIDOS EN LA CIUDAD DE IBARRA PROVINCIA DE IMBABURA.”**, ha sido desarrollado en su totalidad por la Sr. Lucano Chávez Fausto Bolívar, portador de cédula de identidad Nro. 0401394747.

Es todo cuanto puedo certificar en honor a la verdad.

Msc. Edgar Maya

DIRECTOR

# CERTIFICADO



## UNIVERSIDAD TÉCNICA DEL NORTE



Resolución No. 001-073 CEAACES-2013-13

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

Ibarra, 22 de Noviembre del 2018

Msc. Ingeniero

Pedro Granda

COORDINADOR DE LA ESCUELA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

Presente.

De mi consideración:

Reciba un cordial saludo, en la convocatoria para financiamientos por parte del CUICYT 2017, se aprobó el proyecto de investigación denominado "OPTIMIZACIÓN DE LAS RUTAS DE RECOLECCIÓN Y TRATAMIENTO ANAEROBIO PARA LA INDUSTRIALIZACIÓN ENERGÉTICA DE LOS DESECHOS SÓLIDOS MUNICIPALES". En referencia a los objetivos y actividades de viabilidad de implementación del mencionado proyecto, se sustenta el trabajo de grado titulado "Optimización de rutas para la recolección de desechos sólidos en la ciudad de Ibarra, provincia de Imbabura", planteada por el Sr. Fausto Lucano, el cual se le brinda las facilidades técnicas para su desarrollo, con la Tutoría del MSc. Edgar Maya.

Seguros de contar con su favorable atención, anticipamos nuestros debidos agradecimientos.

Atentamente,

Dr. Gerardo Collaguazo

DIRECTOR DEL PROYECTO

## DEDICATORIA

Considero a este proyecto como la culminación de otra etapa de mi vida, lo dedico principalmente a mi madre y mi hermana, fieles acompañantes en mi continua lucha, gracias a ellas grandes metas se han cumplido y su ejemplo ha sido fiel guía hacia el éxito.

También dedico este trabajo a quienes con su conocimiento han colaborado en la ejecución de este trabajo.



**Fausto Bolívar Lucano Chávez**

## AGRADECIMIENTOS

Agradezco de manera especial a la Universidad Técnica del Norte por haberme brindado la oportunidad de ser parte de sus miembros estudiantiles, gracias a ésta institución tan apreciada he logrado adquirir altos conocimientos y formarme tanto de manera intelectual, cómo también de manera ética.

También agradezco abundantemente la guía de mi tutor de tesis el MSc. Edgar Maya por haber donado su valioso tiempo y conocimiento para convertir mi trabajo en una obra de alto valor académico.

Finalmente, un agradecimiento especial a mi madre y mi hermana por su total apoyo y por haberme sostenido en los momentos más difíciles, su entrega y dedicación a mi vida son incalculables.



**Fausto Bolívar Lucano Chávez**

## RESUMEN

El presente proyecto se refiere sobre una optimización de rutas para la recolección de desechos sólidos en la ciudad de Ibarra provincia de Imbabura, realizado con herramientas libres como Framework Laravel con base de datos PostgreSQL, y más componentes que son descritas en los siguientes capítulos.

El aplicativo de optimización de rutas consiste en optimizar la recolección de desechos sólidos en la ciudad de Ibarra, por consiguiente, el diseño e implementación de los módulos de registro de usuarios, registro de rutas, registro de contenedores, representación de zonas, seguimiento de vehículos recolectores, medición de volumen y peso de contenedores, y dar reportes detallados del estado de cada uno de los contenedores y vehículos de recolección.

En la Introducción, se detalla el planteamiento de problema, situación actual, prospectiva, objetivos, alcance y justificación para el inicio del proyecto de la tesis.

En el Capítulo I, se muestra el Marco Teórico referente a las competencias municipales articuladas a los objetivos del Plan Nacional y a sus Objetivos de desarrollo sustentable, así como también las herramientas a ser aplicadas, así como también la metodología de optimización de rutas.

En el Capítulo II, se realiza el desarrollo del Aplicativo Web de optimización de rutas para la recolección de los desechos sólidos, empleando las fases de la metodología XP.

En el Capítulo III, se presenta los resultados obtenidos de la aplicación y su respectiva validación.

En el capítulo IV, se define las conclusiones, recomendaciones del aplicativo con los resultados en la tesis.

Palabras Clave: Aplicativo web, Framework Laravel, Metodología XP



## **ABSTRACT**

This project refers to an optimization of routes for the collection of solid waste in the city of Ibarra province of Imbabura, made with free tools such as Laravel Framework with PostgreSQL database, and more components that are described in the following chapters.

The route optimization application consists of optimizing the collection of solid waste in the city of Ibarra, consequently the design and implementation of user registration modules, route registration, container registration, zone representation, tracking of collection vehicles, measurement of volume and weight of containers, and give detailed reports on the status of each of the containers and collection vehicles.

In the Introduction, the problem statement, current situation, prospective, objectives, scope and justification for the start of the thesis project are detailed.

In Chapter I, the Theoretical Framework is shown regarding the municipal competences articulated to the objectives of the National Plan and its Sustainable Development Objectives, as well as the tools to be applied as well as the route optimization methodology.

In Chapter II, the development of the Web Application for the optimization of routes for the collection of solid waste is carried out, using the phases of the XP methodology.

In Chapter III, the results obtained from the application and its respective validation are presented.

In chapter IV, the conclusions, recommendations of the application are defined with the results in the thesis.

Key Words: Web application, Laravel Framework, XP Methodology

## INDICE DE CONTENIDOS

INTRODUCCION .....	1
<i>Antecedentes</i> .....	1
<i>Situación Actual</i> .....	1
<i>Prospectiva</i> .....	2
<i>Planteamiento del Problema</i> .....	2
<i>Objetivos</i> .....	3
<i>Objetivo General</i> .....	3
<i>Objetivos Específicos</i> .....	3
<i>Justificación</i> .....	4
<i>Alcance</i> .....	5
CAPÍTULO I .....	6
1. MARCO TEÓRICO .....	6
1.1. <i>Plan Nacional de Gestión Integral de Residuos Sólidos</i> .....	6
1.1.1. Componentes.....	6
1.1.2. Estrategias.....	7
1.2. <i>Planes de Desarrollo en el Ecuador</i> .....	8
1.2.1. Objetivos del Plan Nacional para El Buen Vivir.....	9
1.2.2. Objetivos Nacionales de Desarrollo “ <i>Toda Una Vida</i> ” .....	10
1.3. <i>Competencias Exclusivas de GADs Municipales</i> .....	11
1.4. <i>Modelo de ruteo vehicular CVRP</i> .....	12
1.4.1. Variante del modelo VRP .....	14
1.4.2. Ruteo de Vehículos tradicional .....	15
1.4.3. Métodos exactos y aproximados existentes.....	15
1.4.3.1. Métodos exactos.....	15
1.4.3.2. Métodos aproximados.....	16
1.5. <i>Arquitectura WEB</i> .....	17
1.5.1. ¿Qué es Arquitectura Web?.....	17
1.5.2. Componentes semánticos de la Web .....	18
1.5.3. Componentes software de la Web .....	18
1.6. <i>Framework Laravel</i> .....	20
1.6.1. ¿Qué es un Framework? .....	20
1.6.2. Composición de un Framework .....	20
1.6.3. Laravel.....	21
1.6.4. Ventajas de desarrollar con Laravel.....	22
1.7. <i>Base de Datos PostgreSQL</i> .....	22
1.8. <i>Componente PostGIS</i> .....	23
1.9. <i>Metodología XP</i> .....	24
1.9.1. Introducción.....	24
1.9.2. VALORES XP .....	25
1.9.3. Roles XP.....	26
1.9.4. Ventajas Y Desventajas .....	27
1.9.5. Fases.....	27
1.9.5.1. Planeación.....	28
1.9.5.2. Diseño .....	29
1.9.5.3. Desarrollo.....	29
1.9.5.4. Pruebas .....	30
CAPÍTULO II .....	31
2. DESARROLLO DE LA APLICACIÓN .....	31
2.1. <i>ESTRUCTURACIÓN DE APLICACIÓN BASADA EN XP</i> .....	31

2.1.1.	Introducción.....	31
2.1.2.	Planificación Del Proyecto .....	31
2.1.3.	Visión General.....	32
2.1.3.1.	Propósito Y Funcionalidad .....	32
2.1.3.2.	Alcance.....	33
2.1.3.3.	Perspectiva del Producto .....	33
2.1.3.4.	Módulos del Proyecto.....	33
2.1.4.	Organización del Proyecto .....	34
2.1.4.1.	Participantes en el Proyecto .....	34
2.1.4.2.	Plan de fases .....	35
2.2.	<i>FASE DE PLANEACIÓN</i> .....	35
2.2.1.	Planificación inicial de las iteraciones.....	35
2.2.2.	Iteración 2 .....	38
2.2.3.	Iteración 3 .....	40
2.2.4.	Iteración 4.....	42
2.3.	<i>FASE DE DISEÑO</i> .....	44
2.3.1.	Diagramas de Casos de Uso .....	44
2.3.1.1.	Actores del Proceso .....	44
2.3.2.	Diagramas de Proceso .....	47
2.3.2.1.	Diagrama de Creación de Usuario .....	47
2.3.2.2.	Diagrama de ingreso y configuración de vehículos de recolección.....	48
2.3.2.3.	Diagrama de ingreso, configuración y georreferenciación de contenedores.....	48
2.3.2.4.	Diagrama de procesamiento y determinación de rutas .....	49
2.3.3.	Arquitectura del sistema web.....	50
2.3.3.1.	Diagrama de Arquitectura del Sistema .....	50
2.3.3.2.	Descripción del aplicativo web .....	50
2.3.3.3.	Descripción del servidor web y flujo de datos.....	50
2.3.3.4.	Módulos del sistema.....	51
2.4.	<i>FASE DE CODIFICACIÓN E IMPLEMENTACIÓN</i> .....	53
2.4.1.	Diagramas de base de datos .....	53
2.4.1.1.	Introducción a la base de datos.....	53
2.4.1.2.	Módulo Usuarios.....	53
2.4.1.3.	Módulo Vehículos .....	55
2.4.1.4.	Módulo Contenedores.....	56
2.4.1.5.	Módulo Mapeo de Rutas .....	57
2.4.1.6.	Módulo Reportes .....	58
2.5.	<i>FASE DE PRUEBAS</i> .....	59
2.5.1.	Ingreso de información de chofer y vehículos en el sistema con sus características. 59	
2.5.2.	Identificación y geolocalización de contenedores de Residuos sólidos .....	60
2.5.3.	Generación de rutas óptimas.....	62
2.5.4.	Generación de reportes y registro de seguimiento.....	63
CAPÍTULO III .....		64
3.	PRESENTACION DE RESULTADOS .....	64
3.1.	<i>Geolocalización de Contenedores</i> .....	64
3.2.	<i>Validación de Datos</i> .....	64
3.3.	<i>Resultados</i> .....	66
3.3.1.	Impactos generados con el proyecto.....	68
CAPÍTULO IV .....		70
4.	CONCLUSIONES Y RECOMENDACIONES.....	70
4.1.	<i>Conclusiones</i> .....	70
4.2.	<i>Recomendaciones</i> .....	70
4.3.	<i>Bibliografía</i> .....	71

## ÍNDICE DE TABLAS

Tabla 1: Ventajas y desventajas de la metodología de Programación Extrema (XP).....	27
Tabla 2: Modelo Tarjeta CRC.....	29
Tabla 3: Planificación del Proyecto .....	31
Tabla 4: Plan de Fases .....	35
Tabla 5: Plan inicial de Iteraciones.....	35
Tabla 6: <i>Historia de Usuario 1</i> .....	36
Tabla 7: <i>Historia de Usuario 2</i> .....	36
Tabla 8: <i>Historia de Usuario 3</i> .....	37
Tabla 9: <i>Historia de Usuario 4</i> .....	38
Tabla 10: <i>Historia de Usuario 5</i> .....	39
Tabla 11: <i>Historia de Usuario 6</i> .....	39
Tabla 12: <i>Historia de Usuario 7</i> .....	40
Tabla 13: <i>Historia de Usuario 8</i> .....	41
Tabla 14: <i>Historia de Usuario 9</i> .....	41
Tabla 15: <i>Historia de Usuario 10</i> .....	42
Tabla 16: <i>Historia de Usuario 11</i> .....	42
Tabla 17: <i>Historia de Usuario 12</i> .....	43
Tabla 18: <i>Elementos básicos del Diagrama Casos de Uso</i> .....	44
Tabla 19: <i>Características de las subzonas de la ciudad de Ibarra</i> .....	65
Tabla 20: <i>Características de las subzonas de la ciudad de Ibarra</i> .....	66
Tabla 21: <i>Características de los distritos propuestos</i> .....	67
Tabla 22: <i>Especificaciones de zonas residenciales y comerciales</i> .....	67
Tabla 23: <i>Impactos generados</i> .....	68

## INDICE DE FIGURAS

Figura 1: División Administrativa de la Ciudad de Ibarra .....	2
Figura 2: Diagrama Ichikawa Planteamiento del Problema .....	3
Figura 3: Esquema del Marco Legal.....	4
Figura 4: Diagrama de la Arquitectura Propuesta.....	5
Figura 5: Modelo de Ruteo Simple VRP .....	12
Figura 6: Representación Gráfica del Modelo de Ruteo en la ciudad de Ibarra .....	13
Figura 7: Grafo de ruteo según el depósito y nodos .....	14
Figura 8: Diagrama Cliente Servidor .....	19
Figura 9: Diagrama Cliente Servidor .....	20
Figura 10: Logo Laravel .....	21
Figura 11: Logo de postgresQL .....	23
Figura 12: Logo de PostGIS.....	23
Figura 13: Esquema de las fases de la Metodología Programación Extrema (XP).....	28
Figura 14: Diagrama de caso de uso general.....	46
Figura 15: Diagrama de caso de uso del rol de Administrador .....	46
Figura 16: Diagrama de caso de uso del rol de Cliente .....	46
Figura 17: Diagrama de caso de uso del rol de Invitado.....	47
Figura 18: Diagrama de la creación de usuario .....	47
Figura 19: Diagrama de ingreso y configuración de vehículos recolectores .....	48
Figura 20: Diagrama de ingreso, configuración y georreferenciación de contenedores.....	48
Figura 21: Diagrama de Procesamiento y Determinación de rutas.....	49
Figura 22: Diagrama de la Arquitectura del aplicativo Web .....	50
Figura 23: Tablas de BD módulo Usuarios.....	54
Figura 24: Tablas del módulo de vehículos recolectores.....	55
Figura 25: Tablas del módulo de contenedores.....	56
Figura 26: Tablas de BD módulo Mapeo de Rutas.....	57

Figura 27: Tablas de BD módulo Reportes .....	58
Figura 28: Formulario de configuración del vehículo .....	59
Figura 29: Visor de Georreferenciación de Contenedores.....	60
Figura 30: Visor geográfico para el Ruteo.....	62
Figura 31: Reportes y estadísticas Gráficos.....	63
Figura 32: Visor de monitoreo de Contenedores.....	64
Figura 33: Mapeo de Zonas propuestas para la Gestión de Residuos Sólidos.....	68

## **INTRODUCCION**

### **Antecedentes**

Según menciona el Ministerio de Ambiente (MAE, 2002) en su Programa Nacional para la Gestión Integral de Desechos Sólidos, la generación de residuos en nuestro país es aproximadamente de 4,06 millones de toneladas métricas al año y una generación per cápita de 0,74 kg., en este ámbito es necesario un manejo integral y planificado de los residuos sólidos en el Ecuador.

El Plan de Gestión Integral de los desechos sólidos garantiza la calidad, la frecuencia en los servicios de recolección, transporte, aprovechamiento, tratamiento y disposición final de los desechos sólidos, en este ámbito el Plan estimula la utilización de procesos y tecnologías alternativos para el tratamiento de los Desechos Sólidos Municipales (Web, 2002)

Según datos oficiales, Ibarra es la ciudad que más desperdicios produce, con un promedio de 130 toneladas al día, mientras que en el resto de cantones de la provincia se recolectan alrededor de 250 toneladas diarias, lo cual representa el 63% del volumen de basura orgánica que se obtiene en la ciudad, con lo cual se produciría a futuro biofertilizantes para plantaciones MAE (2002).

La recolección diferenciada inorgánica en Ibarra se clasifica básicamente en: plástico, metal, vidrio, cartón, papel, neumáticos, aceites usados.

### **Situación Actual**

En Ibarra, no existe ningún sistema de optimización de rutas para la recolección de los desechos sólidos, ni un medio para determinar capacidad, peso y volumen de los contenedores que se encuentran ubicados en cada manzana de la ciudad; esto se debe al desconocimiento sobre las características técnicas de cada contenedor, lo cual provoca una mala toma de decisiones acerca de las rutas para la respectiva recolección.

Actualmente la ciudad de Ibarra está dividida administrativamente por Parroquias Urbanas y Barrios, tenemos cinco parroquias urbanas: El Sagrario, San Francisco, Caranqui, Alpachaca y la Dolorosa de Priorato, y siete parroquias rurales: Ambuquí, Angochagua, La Carolina, La Esperanza, Lita, Salinas y San Antonio.

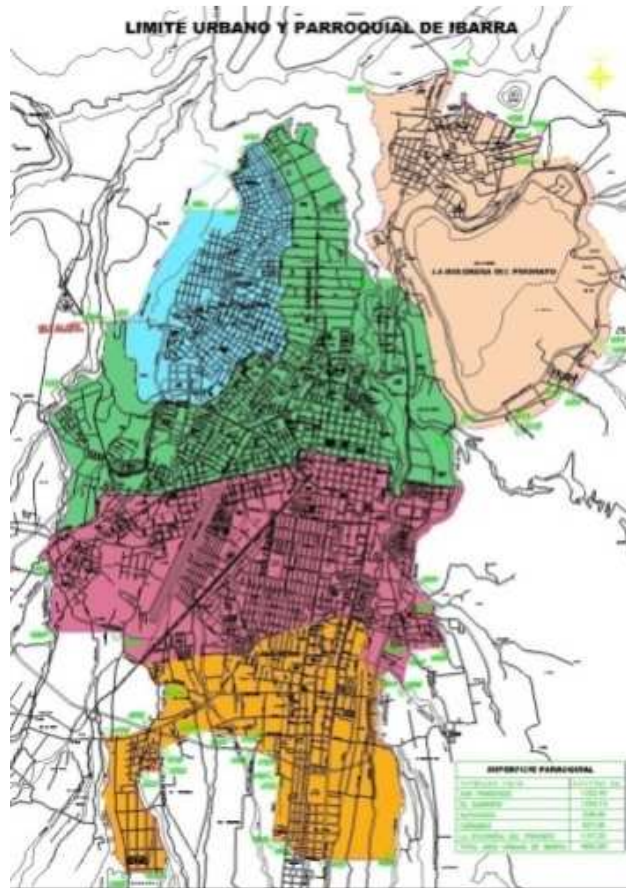


Figura 1: División Administrativa de la Ciudad de Ibarra

*Fuente: GAD Municipal año 2017*

Además, cuenta con varios contenedores, mismos que han sido ubicados en cada una de las manzanas, con el fin de brindar el servicio de recolección a la colectividad, de la misma forma la municipalidad de Ibarra dispone de unidades móviles para la recolección de los desechos sólidos

### **Prospectiva**

Disponer de una herramienta tecnológica de optimización de las rutas de recolección de los desechos sólidos municipales, brindando un aporte significativo en la gestión segura de los desechos sólidos y de esta manera aportar a la reducción de la contaminación ambiental.

### **Planteamiento del Problema**

La ausencia de una coordinación efectiva en la formulación de planes o proyectos de las instancias de gobiernos autónomos descentralizados, es una de las causas de la persistencia de problemas técnicos, operativos para resolver la problemática de los residuos sólidos. A su vez dificulta obtener un análisis minucioso de las diferentes rutas de



recolección considerando además que los contenedores tienen su determinada capacidad y volumen.

En el aspecto tecnológico existen mecanismos para lograr determinar en tiempo real las rutas efectivas de recolección de desechos sólidos, además de las características del estado actual de cada uno de los contenedores ubicados en la ciudad.

La contaminación ambiental es una de las problemáticas que se genera al no disponer de información, que permita verificar el volumen y peso de los contenedores, por tal motivo algunos contenedores quedan llenos ocasionando molestias a la ciudadanía, que a su vez son expuestos a enfermedades respiratorias o virales.



Figura 2: Diagrama Ichikawa Planteamiento del Problema

Fuente: Elaboración Propia

## Objetivos

### Objetivo General

Diseñar un aplicativo Web de optimización de rutas para la recolección de desechos sólidos, utilizando tecnologías de posicionamiento global y sistemas de información geográfica.

### Objetivos Específicos

- Desarrollar un aplicativo tipo Web de optimización de rutas, utilizando la metodología XP y framework Laravel como herramienta de software libre.
- Aplicar el Modelo de ruteo vehicular CVRP en la optimización de rutas.
- Identificar y geolocalizar los contenedores existentes en la ciudad de Ibarra.
- Validar información recopilada en la ciudad de Ibarra para determinar rutas óptimas.

## Justificación

Justificación Tecnológica. - El problema de ruteo de vehículos es uno de los temas más importantes en las redes de distribución; con diversas aplicaciones en transporte, logística y telecomunicaciones. Debido a la dificultad que implica resolver instancias de gran tamaño con métodos exactos, la mayoría de investigaciones se han centrado en resolver problemas de la vida real a través de técnicas metaheurísticas. En este ámbito, se generan soluciones aproximadas usando algoritmos heurísticos.

Justificación Teórica. - Los Objetivos de la Agenda del Desarrollo Sostenible permiten incorporar los desafíos globales a los que nos enfrentamos día a día, como la pobreza, la desigualdad, el clima, la degradación ambiental, la prosperidad, y la paz y la justicia. Además, permiten garantizar la gestión segura de los desechos sólidos y la reducción de la contaminación ambiental, en este ámbito la articulación viene dada por:



Figura 3: Esquema del Marco Legal

*Fuente: Elaboración Propia*

El Art. 264 de la Constitución de la República los gobiernos municipales "tendrán entre otras competencias exclusivas el prestar los servicios públicos de manejo de desechos sólidos y actividades de saneamiento ambiental;"

El Art. 55 del COOTAD en concordancia con lo establecido en la Constitución de la República, ratifica como una competencia exclusiva de los gobiernos autónomos descentralizados municipales la prestación, de entre otros servicios, los de manejo de desechos sólidos

Justificación Metodológica. - Los sistemas SIG (Sistemas de información geográfica) son herramientas esenciales para la localización y ubicación de los contenedores, así como también los vehículos de recolección, permiten además estimar distancias y separaciones

entre lugares. de la misma manera el aplicativo de desarrollar con una metodología ágil de programación denominada XP.

### Alcance

Brindar un aporte significativo en la gestión de recolección segura de los desechos sólidos, a través de un aplicativo Web que permita llevar un control adecuado de contenedores y camiones de recolección, con tecnología GPRS y GSM; aportando de esta manera a la reducción de la contaminación ambiental,

Con la metodología XP permitirá desarrollar el aplicativo con arquitectura MVC y la documentación reglamentaria. Además de la utilización de herramientas de software libre con lenguaje de programación en PHP y LARAVEL como Framework.

Utilizando herramientas de Sistemas de Información Geográfica se obtendrá un mapa temático de la ciudad de Ibarra permitiendo identificar con claridad cada uno de los elementos geográficos como son los contenedores, predios y vías.

A través del modelo de ruteo vehicular permitirá determinar el conjunto de rutas de los vehículos de recolección, mejorando de esta manera la organización en el servicio.

La utilización de dispositivos tecnológicos en contenedores permitirá determinar la capacidad en peso y volumen de biomasa existente; de la misma manera el dispositivo en vehículos de recolección les ayudará a definir las rutas específicas en función a su capacidad de carga y localización de contenedores.

### Arquitectura Propuesta

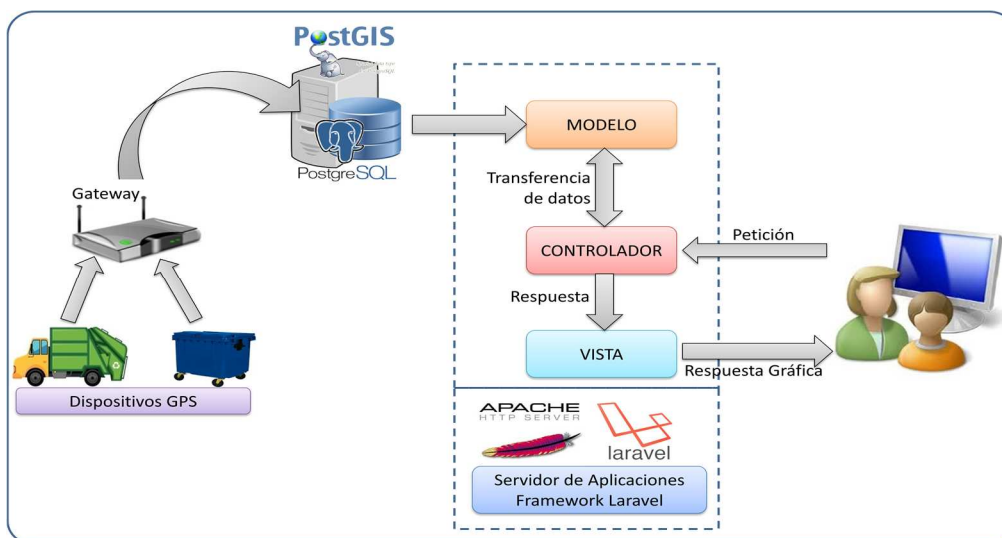


Figura 4: Diagrama de la Arquitectura Propuesta

Fuente: Elaboración Propia

## **CAPÍTULO I**

### **1. MARCO TEÓRICO**

#### **1.1. Plan Nacional de Gestión Integral de Residuos Sólidos**

Según menciona el Ministerio de Ambiente (MAE, 2002) en su Programa Nacional para la Gestión Integral de Desechos Sólidos, la generación de residuos en nuestro país es aproximadamente de 4,06 millones de toneladas métricas al año y una generación per cápita de 0,74 kg., en este ámbito es necesario un manejo integral y planificado de los residuos sólidos en el Ecuador.

Las metas definidas por el Programa contemplaban el que un 70% de la población del Ecuador disponga sus desechos en un relleno sanitario técnicamente manejado hasta el año 2014.

Actualmente el Programa Nacional para la Gestión Integral de Desechos Sólidos (PNGIDS, 2010) brinda el apoyo técnico necesario a los diferentes municipios del país, además de dotación de estudios e incentivos en función a la necesidad y urgencia que presente cada uno, de esta forma realizan un trabajo conjunto con el Programa mencionado MAE (2002).

##### **1.1.1. Componentes**

El Ministerio de Ambiente (MAE, 2002) realiza su gestión en base a los siguientes componentes:

a) Político:

Generar una Política de Gestión Integral para el buen manejo de los Desechos Sólidos.

b) Técnico:

Fortalecimiento de las capacidades técnicas y operativas de los municipios en el manejo integral de los desechos sólidos.

c) Participación:

Es importante el empoderamiento activo de la sociedad Para la aplicación del modelo de Gestión Integral de desechos sólidos es prioritario el involucramiento activo de la sociedad civil.

d) Inclusión económica y social:

En Ecuador existen aproximadamente 20.000 recicladores, grupo altamente vulnerable que ha venido realizando las labores de recolección de residuos reciclables de manera informal, el programa pretende gestionar la inclusión social y económica de este grupo dentro de la cadena de valor de los residuos.

e) Optimización de servicios:

Garantizar la calidad y la frecuencia en los servicios de recolección, transporte, aprovechamiento, tratamiento y disposición final de los desechos sólidos.

f) Responsabilidad y corresponsabilidad:

Extendida del productor y producción limpia.

g) Control, seguimiento y Monitoreo MAE (2002).

### **1.1.2. Estrategias**

El Ministerio de Ambiente (MAE, 2002) menciona que para la aplicación de la política de gestión integral de residuos sólidos según el Plan Nacional para la Gestión Integral de Desechos Sólidos contempla las siguientes estrategias:

a) Categorización y priorización de los GADs:

Los parámetros de clasificación de los GADs responden a criterios de población así se clasifican en: Grandes con una población que va desde 300.001 a 5´000.000 de habitantes (2,26 % de los GADs); Medianos con una población de 50.001 a 300.000 habitantes (23,08% de los GADs); Pequeños con una población de 15.001 a 50.000 habitantes (41,18% de los GADs); y Micros con una población menor a 15.000 habitantes (33,48% de los GADs). Es necesario considerar las realidades técnicas y económicas en el manejo de los Desechos de los GADs a fin de priorizar los municipios con menos capacidad de gestión para intervenir.

b) Diseño e implementación de un modelo estandarizado de Gestión Integral de Residuos Sólidos

El PNGIDS MAE diseñará un Modelo para la Gestión Integral de Desechos Sólidos adaptable a nivel país, la aplicabilidad en cada municipio dependerá de su capacidad de gestión y del análisis de su realidad en cuanto a población e impacto derivado de la generación y el manejo de los desechos.

c) Cierre técnico de botaderos y rellenos sanitarios:

En la actualidad en el Ecuador existen 144 botaderos a cielo abierto y 77 rellenos sanitarios. El PNGIDS apunta que al finalizar su gestión se hayan hecho los cierres

técnicos de todos los botaderos existentes y que los rellenos sanitarios existan como centros de disposición final de materiales no reciclables y no combustibles, materiales peligrosos, ceniza y escombros que surjan de los procesos de aprovechamiento energético de los RSM.

d) Agregación de valor

Como se ha mencionado, el aprovechamiento actual de los residuos es mínimo en todas las etapas de la cadena de tránsito de estos. Para entender la situación es necesario clasificar a los residuos en dos tipos: los Residuos Sólidos Urbanos y los Residuos Sólidos Especiales. Así entonces la cadena de valor actualmente se maneja de la siguiente manera:

La propuesta del PNGIDS es agregar valor a la cadena bajo los siguientes parámetros:

e) Reestructuración del modelo tarifario

Modificar el modelo tarifario de los servicios dados por los GADs respecto de la gestión de residuos sólidos urbanos y los residuos sólidos especiales, puesto que actualmente la inversión de los GADs en dichos servicios no es recuperada con el tiempo y tampoco existe claridad de una tarifa (monto o medio de cobro) que permita la recuperación de esa inversión.

f) Creación de mancomunidades

Para los GADs pequeños y micro la gestión individual de los Desechos sólidos representa una inversión que no se recupera con el tiempo, por ello es necesario pensar en agrupar a los GADs pequeños y Micros para el manejo de los Desechos en Mancomunidad, así también la inversión individual disminuye. MAE (2002).

## **1.2. Planes de Desarrollo en el Ecuador**

Los Objetivos del Desarrollo Sostenible de la Agenda 2030. Los Objetivos de la Agenda permiten incorporar los desafíos globales a los que nos enfrentamos día a día, como la pobreza, la desigualdad, el clima, la degradación ambiental, la prosperidad, y la paz y la justicia.

En este ámbito se hace referencia a los objetivos que pretenden aprovechar de mejor manera los recursos, así como también la reducción de la contaminación ambiental. y es así que los Objetivos 11 y 12 determinan mejorar la gestión segura de los desechos sólidos, el uso eficiente de los recursos y la energía, además de la construcción de

infraestructuras que no dañen el medio ambiente, logrando al mismo tiempo una mejor calidad de vida.

De los Objetivos del Plan Nacional de Desarrollo para el Buen Vivir 2017-2021. Los objetivos determinados en el Plan, están orientados a un enfoque de derechos y se articula con la Agenda 2030 para el Desarrollo Sostenible, además presenta los nueve objetivos nacionales, agrupados en tres ejes: Derechos para todos durante toda la vida; Economía al servicio de la sociedad; y, Más sociedad, mejor Estado (PNBV, 2017).

Dentro del Plan se menciona garantizar los derechos de la naturaleza y en especial la promoción de las buenas prácticas en el aporte a la reducción de la contaminación, así como también la adaptación a efectos del cambio climático (PNBV, 2017).

En el tercer eje del Plan, nos determina, potenciar el talento humano con procesos integrales de educación, y generar capacidades productivas con grandes inversiones en diversas áreas de la infraestructura y sectores estratégicos en el desarrollo (PNBV, 2017).

### **1.2.1. Objetivos del Plan Nacional para El Buen Vivir**

EL PNBV destinado a ser un referente en Latinoamérica, pues la región está viendo resultados concretos en el caso ecuatoriano (Ecuador R. d., 2017)

Para los proyectos culminados en el periodo 2013 a 2017, de los cuales estén ligados al Plan Nacional del Buen Vivir se tomará los siguientes objetivos:

- Consolidar el Estado democrático y la construcción del poder popular
- Auspiciar la igualdad, la cohesión, la inclusión y la equidad social y territorial, en la diversidad.
- Mejorar la calidad de vida de la población.
- Fortalecer las capacidades y potencialidades de la ciudadanía.
- Construir espacios de encuentro común y fortalecer la identidad nacional, las identidades diversas, la plurinacionalidad y la interculturalidad.
- Consolidar la transformación de la justicia y fortalecer la seguridad integral, en estricto respeto a los derechos humanos.
- Garantizar los derechos de la naturaleza y promover la sostenibilidad ambiental territorial y global.
- Consolidar el sistema económico social y solidario, de forma sostenible.
- Garantizar el trabajo digno en todas sus formas.

- Impulsar la transformación de la matriz productiva.
- Asegurar la soberanía y eficiencia de los sectores estratégicos para la transformación industrial y tecnológica.
- Garantizar la soberanía y la paz, profundizar la inserción estratégica en el mundo y la integración latinoamericana.

### **1.2.2. Objetivos Nacionales de Desarrollo “Toda Una Vida”**

Según el Plan Nacional de Desarrollo, para el período 2017-2021 aprobado el 22 de septiembre del 2017, determina tres Ejes Programáticos y nueve Objetivos Nacionales de Desarrollo, en base a la sustentabilidad ambiental y el desarrollo territorial (Ecuador, 2017)

Los proyectos que estén culminados o en ejecución de acuerdo al Plan Nacional de desarrollo “Toda una vida” 2017-2021 deberán estar ligados a los siguientes objetivos:

Eje 1: Derechos para Todos Durante Toda la Vida (PNBV, 2017):

- Objetivo 1: Garantizar una vida digna con iguales oportunidades para todas las personas.
- Objetivo 2: Afirmar la interculturalidad y plurinacionalidad, revalorizando las identidades diversas.
- Objetivo 3: Garantizar los derechos de la naturaleza para las actuales y futuras generaciones.

Eje 2: Economía al Servicio de la Sociedad (PNBV, 2017):

- Objetivo 4: Consolidar la sostenibilidad del sistema económico social y solidario, y afianzar la dolarización
- Objetivo 5: Impulsar la productividad y competitividad para el crecimiento económico sostenible de manera redistributiva y solidaria
- Objetivo 6: Desarrollar las capacidades productivas y del entorno para lograr la soberanía alimentaria y el Buen Vivir rural

Eje 3: Más sociedad, mejor Estado (PNBV, 2017).

- Objetivo 7: Incentivar una sociedad participativa, con un Estado cercano al servicio de la ciudadanía.
- Objetivo 8: Promover la transparencia y la corresponsabilidad para una nueva ética social.
- Objetivo 9: Garantizar la soberanía y la paz, y posicionar estratégicamente el país en la región y el mundo.



### 1.3. Competencias Exclusivas de GADs Municipales

El PDOT es el Plan de Desarrollo y Ordenamiento Territorial son instrumentos de la planificación previstos por la Ley, que permiten a los gobiernos autónomos descentralizados provinciales y subprovinciales, desarrollar la gestión concertada de sus territorios, orientada al logro del buen vivir.

Según el PDOT define una visión objetivo, hacia dónde va la provincia, las metas y estrategias que deben aplicarse para alcanzarla, la forma en que debe organizarse el territorio y las normas y disposiciones que deben cumplir las entidades y los ciudadanos para usar y ocupar el territorio de manera que se garantice su sostenibilidad. Definen asimismo los lineamientos que deben observarse para alcanzar el desarrollo integral que genere condiciones de buen vivir para los ciudadanos

Según la Constitución de la República del Ecuador Las competencias exclusivas de los GADs Municipales según en su Art. 264 de la determina que, "Los gobiernos autónomos descentralizados municipales tendrán las siguientes competencias exclusivas sin perjuicio de otras que determine la ley". (CONSTITUYENTE, 2008, pág. 130)

d) "Prestar los servicios públicos de agua potable, alcantarillado, depuración de aguas residuales, manejo de desechos sólidos, actividades de saneamiento ambiental y aquellos que establezca la ley". (CONSTITUYENTE, 2008, pág. 130)

Respecto a las funciones de los Gads según el Art. 54. Son funciones del gobierno autónomo descentralizado municipal las siguientes:

a) Promover el desarrollo sustentable de su circunscripción territorial cantonal, para garantizar la realización del buen vivir a través de la implementación de políticas públicas cantonales, en el marco de sus competencias constitucionales y legales;

b) Diseñar e implementar políticas de promoción y construcción de equidad e inclusión en su territorio, en el marco de sus competencias constitucionales y legales;

c) Establecer el régimen de uso del suelo y urbanístico, para lo cual determinará las condiciones de urbanización, parcelación, lotización, división o cualquier otra forma de fraccionamiento de conformidad con la planificación cantonal, asegurando porcentajes para zonas verdes y áreas comunales;

i) Implementar el derecho al hábitat y a la vivienda y desarrollar planes y programas de vivienda de interés social en el territorio cantonal;

j) Implementar los sistemas de protección integral del cantón que aseguren el ejercicio garantía y exigibilidad de los derechos consagrados en la Constitución y en los instrumentos internacionales, lo cual incluirá la conformación de los consejos cantonales, juntas cantonales y redes de protección de derechos de los grupos de atención prioritaria. Para la atención en las zonas rurales coordinará con los gobiernos autónomos parroquiales y provinciales (PDOT, 2012).

k) Regular, prevenir y controlar la contaminación ambiental en el territorio cantonal de manera articulada con las políticas ambientales nacionales.

#### 1.4. Modelo de ruteo vehicular CVRP

El problema de ruteo de vehículos (VRP) es uno de los temas más importantes en las redes de distribución; con diversas aplicaciones en transporte, logística y telecomunicaciones. Debido a la dificultad que implica resolver instancias de gran tamaño con métodos exactos, la mayoría de los investigadores se han centrado en resolver problemas de la vida real a través de técnicas metaheurísticas. En este trabajo, se generan soluciones aproximadas para el CVRP tradicional usando la heurística de barrido de dos fases y resolviendo cada problema del agente viajero con el algoritmo genético modificado de Chu-Beasley. Se valida la metodología con casos de prueba de la literatura especializada.

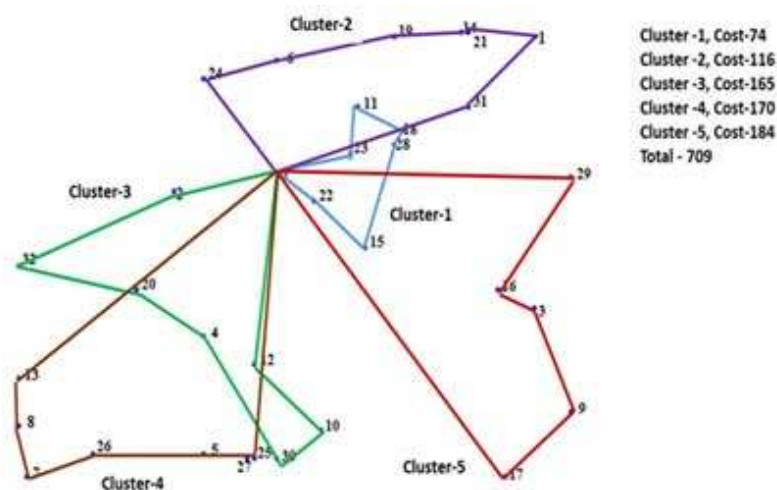


Figura 5: Modelo de Ruteo Simple VRP

Fuente: <http://www.computerscijournal.org/vol10no3/constructive-and-clustering-methods-to-solve-capacitated-vehicle-routing-problem/>

Hoy en día el problema de rutas de vehículos es materia de investigación debido a los múltiples casos que se presenta en la vida diaria y es ahí que los grafos permiten resolver

y optimizar rutas, sin embargo, es necesario tomar en cuenta otros parámetros que ayudaran al ruteo.

Los problemas de rutas de vehículos intentan determinar un conjunto de rutas de una flota de vehículos para brindar un determinado servicio. Según (Ramser, 1964) quienes fueron los primeros tratar estos problemas en 1959, cuando describieron una aplicación real concerniente a la distribución de gasolina para estaciones de servicio, además, se propuso una formulación matemática del problema, y una aproximación algorítmica (Suresh Nanda Kumar, 2015).

Según la revista científica en el problema de enrutamiento del vehículo (VRP), los métodos de optimización exactos pueden ser complejos de resolver, en tiempos de CPU aceptables, cuando el problema presenta conjuntos de datos reales que son bastante grandes. Para obtener soluciones en la determinación de rutas realistas y muy cercanas a la solución óptima, en ese caso uno tiene que usar heurísticas y meta-heurísticas (Suresh Nanda Kumar, 2015).

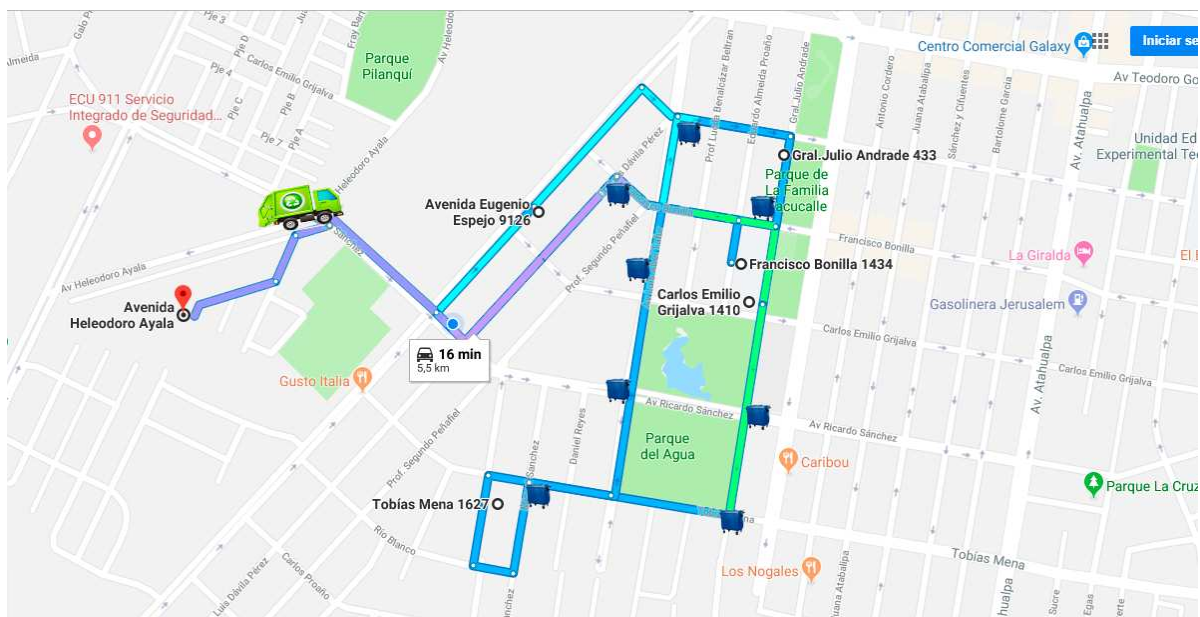


Figura 6: Representación Gráfica del Modelo de Ruteo en la ciudad de Ibarra

Fuente: Elaboración Propia

Los problemas de recolección de desechos sólidos se los resuelve a través del Modelo CVRP que permite la preparación y planificación de las diferentes rutas de vehículos de recolección, usando sistemas de navegación GPS.

El Problema CVRP trata de determinar los recorridos de vehículos con capacidad limitada, inicia en un origen, y pasa por donde se encuentran los contenedores para recoger los

desechos sólidos según su estado y llevarlos finalmente al relleno sanitario, de manera que la distancia transitada por el conjunto de vehículos recolectores sea mínima. En el tipo de problema más sencillo no se tiene en cuenta el horario.

Según el modelo CVRP se obtienen las siguientes consideraciones (Villalobos, 2017):

- Minimizar el número total de vehículos recolectores requeridos para la recolección.
- Minimizar los costes fijos asociados al uso de los vehículos.
- Minimizar el coste total de transporte.
- Balancear las rutas, por tiempo de viaje o carga de vehículo.

#### 1.4.1. Variante del modelo VRP

El problema de ruteo de vehículos consiste en determinar un conjunto de rutas de costo mínimo que comiencen y terminen en los depósitos o almacenamientos, de modo que los vehículos de recolección visiten a los clientes por atender sólo una vez.

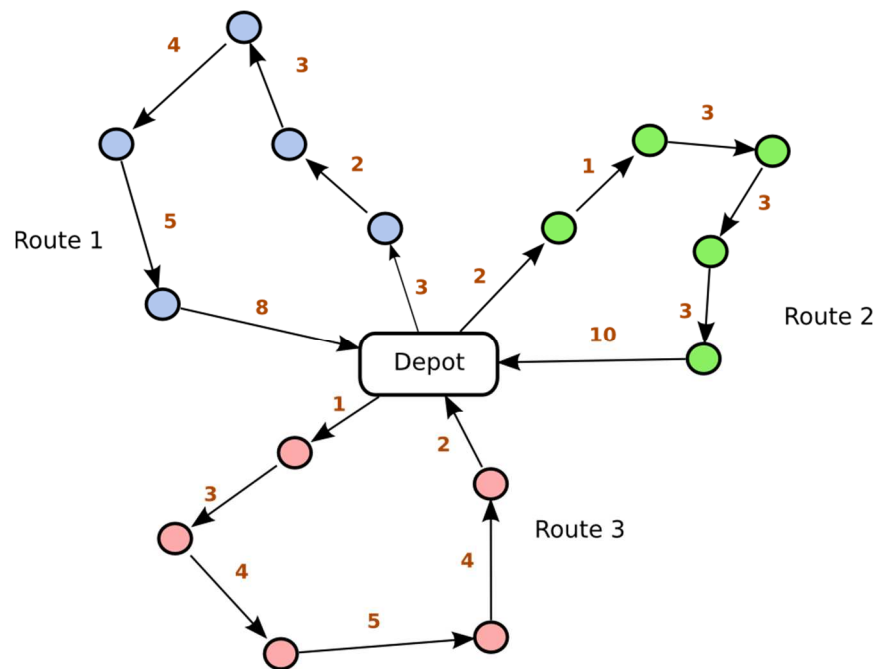


Figura 7: Grafo de ruteo según el depósito y nodos

Fuente: (M. A. H. Akhand, 2017)

En la figura se muestra el problema de ruteo de vehículos no obstante es posible aceptar más de un depósito.

Según (Olivera, 2005) los Problemas de Ruteo de Vehículos son Problemas de Optimización Combinatoria.

El ruteo de vehículos asciende como un problema central tanto en el ámbito del transporte, como en la distribución y logística. Según (Gallart Suárez, 2009) la utilización de los

métodos automatizados da lugar a los ahorros significativos que se extienden a partir de 5% hasta el 20% en los costes totales (Toth & Vigo, 2002).

Matemáticamente el problema de ruteo se puede detallar de la siguiente manera (U.MÁLAGA, 2007):

Sea un grafo  $G(V,E)$  en donde:

$V = \{ v_0 v_1 v_2 \dots v \}$  es un conjunto de vértices (nodos) del grafo, en donde  $v_0$  representa el depósito y los demás vértices representan a los nodos.

$E$  = conjunto de aristas del grafo.

Según (U.MÁLAGA, 2007) consiste en precisar un conjunto de rutas para vehículos en donde el coste total sea mínimo, empezando y terminando en un depósito, tal que cada vértice  $V'$  sea visitado una sola vez U.MÁLAGA (2007).

#### **1.4.2. Ruteo de Vehículos tradicional**

El Problema de Ruteo de Vehículos tradicional fundamenta en agregar a todos los vehículos con una capacidad uniforme, de tal manera que deban satisfacer la demanda de clientes (U.MÁLAGA, 2007).

El objetivo es de este método es reducir al mínimo la cantidad de vehículos, así como la suma del tiempo recorrido en realizar el reparto, además la demanda total de las materias para cada ruta no debe exceder la capacidad del vehículo de reparto (U.MÁLAGA, 2007).

Dentro de la Viabilidad del ruteo la solución es realizable si la cantidad total fijada en cada ruta no excede la capacidad del vehículo de recolección (Gallart Suárez, 2009).

#### **1.4.3. Métodos exactos y aproximados existentes**

Estos tipos de problemas (VRP, VRPTW) se tiene distintos algoritmos que se pueden dividir en algoritmos exactos y algoritmos aproximados (Gallart Suárez, 2009).

##### **1.4.3.1. Métodos exactos**

Dentro de los métodos exactos que resuelven problemas de tipo VRP se tienen:

- Ramificación y acotación (Guimarans Serrano, 2012):
  - Es un método que usa la técnica de “divide y vencerás”.
  - Divide el conjunto de soluciones enteras en subconjuntos disjuntos cada vez menores.

- Determina el valor de la mejor solución del subconjunto.
- Elimina la rama del árbol si la cota indica que no se tiene la solución óptima.

Según el método de Fisher puede llegar a resolver hasta 71 nodos destino, pero esto demanda un costo computacional bastante elevado (U.MÁLAGA, 2007).

➤ Ramificación y corte (Ramos, 2011):

- Es un método híbrido que usa los métodos de ramificación y acotación.
- Selecciona un nodo para evaluar (inicio del nodo raíz).
- Se decide si se va a generar o no planos de corte.
- Por último, se aplican los criterios del método de ramificación y acotación.

Este tipo de soluciones depende mucho de la estructura del VRP, ya que al aplicar cortes a una estructura que no favorece a este algoritmo, la solución puede llegar a complicarse (Contardo Vera, 2005).

Según (Contardo Vera, 2005), estos métodos son aplicables únicamente en una cantidad de nodos máximo de 100, aplicado sólo para propósitos académicos, pero en problemas reales de VRP se tiene más de 500 nodos, por lo tanto, no son recomendables, porque se requiere de un imponente tiempo de ejecución y mucho recurso computacional.

#### **1.4.3.2. Métodos aproximados**

En los métodos aproximados se detallan los métodos heurísticos y meta heurísticos

##### **A) Métodos heurísticos**

Estos métodos permiten dar solución a problemas de tipo VRP se categorizan en:

➤ Métodos constructivos

Estos método construyen gradualmente la solución factible de acuerdo al coste, aunque no tienen una fase de mejora (U.MÁLAGA, 2007).

Este tipo de algoritmo resuelve los problemas de tipo VRP general.

➤ Métodos de inserción

Estos métodos de inserción inician con rutas vacías, luego evalúan la mejor forma de insertar un nodo en alguna ruta, y se quedan con el par de nodos y rutas que representan la mejor inserción (Contardo Vera, 2005).

##### **B) Métodos meta heurísticos**

Estos métodos meta heurísticos solucionan problemas VRP según las siguientes técnicas:

- Colonia de hormigas (David de la Fuente, 2011)
  - Solucionan las siguientes variantes de VRP:
    - Problema de Ruteo de Vehículos con Capacidad.
    - Problema de Ruteo de Vehículos con Ventana de Tiempo
    - Problema de Ruteo de Vehículos con Ventana de Tiempo y Conectividad.
    - Problema de Ruteo de Vehículos Periódico.
- Algoritmos genéticos (Savelsberg, 1995)
 

Solucionan las siguientes variantes de VRP:

  - Problema de Ruteo de Vehículos.
  - Problema de Ruteo de Vehículos con Capacidad.
  - Problema de Ruteo de Vehículos con Ventana de Tiempo.
  - Problema de Ruteo de Vehículos con Ventana de Tiempo y Capacidad.
  - Problema de Ruteo de Vehículos con Plazos.
- Búsqueda tabú
 

Solucionan las siguientes variantes de VRP (Archetti, 2001)

  - Problema de Ruteo de Vehículos.
  - Problema de Ruteo y Programación de Vehículos.
  - Problema de Ruteo de Vehículos con Ventana de Tiempo.
  - Problema de Ruteo de Vehículos con Ventana de Tiempo y Entrega Partida.
  - Problema de Ruteo de Vehículos con Ventana de Tiempo y Apremios.
  - Problema de Ruteo de Vehículos con Entrega y Recogida de Productos Simultáneamente.
  - Problema de Ruteo de Vehículos con Múltiples Depósitos
- GRASP (Gallart Suárez, 2009)
  - Solucionan las siguientes variantes de VRP:
    - Problema de Ruteo de Vehículos Periódico.
    - Problema de Ruteo de Vehículos con Ventana de Tiempo.

## **1.5. Arquitectura WEB**

### **1.5.1. ¿Qué es Arquitectura Web?**

La arquitectura Web, es la disciplina encargada de estructurar, organizar y etiquetar el contenido como también establecer los puntos de acceso, sistemas de búsqueda y recuperación de información de cualquier aplicación soportada en la Web (Pérez-Montoro 2010).

### **1.5.2. Componentes semánticos de la Web**

- World Wide Web:
  - Universo de información interconectada, accesible a través de internet.
- URI: Uniform Resource Identifier.
  - Identifica los recursos web para su acceso y manipulación.
- HTML: HyperText Markup Language.
  - Lenguaje de marcas.
  - Provee una representación estándar de los documentos hipertexto en formato ASCII.
  - Permite formatear texto, integrar imágenes, referenciar otros documentos, etc.
- HTTP: Hypertext Transfer Protocol.
  - Protocolo que permite a los componentes web (cliente, servidores, etc) comunicarse de una forma estándar y bien definida.
  - Define el formato y el significado de los mensajes intercambiados entre componentes web.

### **1.5.3. Componentes software de la Web**

- Arquitectura Cliente/Servidor
  - El modelo cliente-servidor es una arquitectura software que involucra uno o más clientes solicitando servicios a uno o más servidores.
  - El cliente puede ser un proceso corriendo en una computadora o en un dispositivo como una PDA o un teléfono móvil.
  - El servidor puede ser un proceso corriendo en una computadora (normalmente de altas prestaciones).
  - En la arquitectura Web actual aparecen además elementos que se sitúan en medio (proxies, cachés).
  - El principal Beneficio es la Usabilidad/flexibilidad/interoperabilidad/escalabilidad.



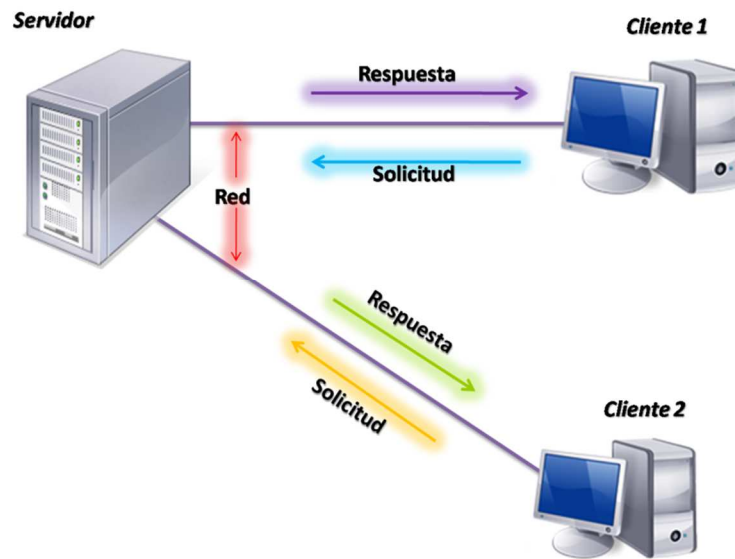


Figura 8: Diagrama Cliente Servidor

Fuente: Elaboración Propia

➤ Clientes

- Originan el tráfico web.
- Envían las peticiones y reciben las respuestas.
- Dos clases de clientes web: navegadores y robots.
- Los navegadores (Netscape, IE, etc).
- Las peticiones están dirigidas por el usuario.
- Repiten peticiones al mismo objeto cuando navegan por un sitio.
- Utilizan caches de memoria y disco.
- Robots (spiders, y agentes inteligentes).
- Las peticiones son automatizadas.
- La velocidad y carga está limitada por la velocidad de proceso, y por la velocidad de la red.

➤ Navegadores

Programa que realiza las peticiones, a solicitud de un usuario, y recibe, analiza y presenta las respuestas.

Pasos:

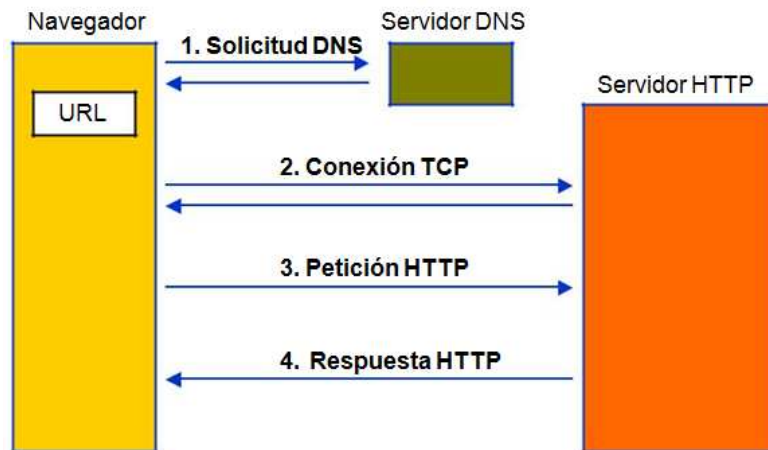


Figura 9: Diagrama Cliente Servidor

*Fuente: Elaboración Propia*

## 1.6. Framework Laravel

### 1.6.1. ¿Qué es un Framework?

Un framework es un conjunto de herramientas, artefactos y estándares que dan funcionalidad extra a una aplicación y facilita su comprensión y desarrollo; por lo tanto estos frameworks sirven para acortar el tiempo de desarrollo y reducir el costo de la inversión que esto implica; no todos los desarrolladores están de acuerdo con estas herramientas ya que las llaman fuentes de mediocridad por que los programadores más experimentados los hacen ellos solos desde cero y por lo tanto es un motivo de polémica hasta hoy en día el uso o no de un framework (CHÁVEZ CABRERA, 2016).

### 1.6.2. Composición de un Framework

La mayoría de framework usan patrones de diseños ya preinstalados, y al usarlos facilitaran al programador realizar un sinfín de aplicaciones según su fin de cada framework ya que guiará en cada proceso a desarrollar.

Los framework llevan consigo una organización de jerarquía de directorios, así el nivel de testeo es mucho mayor que realizar por cuenta propia; así un framework permitirá un trabajo en equipo sin dificultad teniendo un conocimiento del framework más que la aplicación misma.

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

### 1.6.3. Laravel



Figura 10: Logo Laravel

Fuente: <https://laravel.com/> (01/07/2018)

Laravel, propone una forma de desarrollar aplicaciones web de un modo mucho más ágil. Por ejemplo, en Laravel opcionalmente podemos usar el patrón de diseño MVC (Modelo-Vista-Controlador) tradicional, donde al igual que otros frameworks PHP, el controlador es programado como una clase.

Por lo tanto, un Controlador es una clase PHP que dispone de métodos públicos que son el punto de entrada final de una petición HTTP (Request PHP) a nuestra aplicación. Pero, Laravel propone además una forma distinta y más directa de responder a la solicitud HTTP, que veremos enseguida.

Hoy en día existen múltiples frameworks para PHP con ventajas que le permiten a un programador facilitar la creación de una aplicación web. Entre ellos Symfony 2, Codeigniter, Zend Framework y Laravel.

Aunque no vamos a discutir cada uno de ellos en esta publicación, vale la pena mencionarlos ya que siguen siendo utilizados para el desarrollo de aplicaciones web.

Nosotros decidimos utilizar Laravel por ser hoy el framework con mayor crecimiento, popularidad, comunidad involucrada y sobre todo, presentar un código limpio, seguro y constantemente actualizado con los mejores lineamientos en el mundo PHP.

#### **1.6.4. Ventajas de desarrollar con Laravel**

➤ Beneficios

Facilidad de implementación con un par de comandos ya tenemos listos nuestro framework para iniciar el desarrollo de nuestra aplicación.

➤ Seguridad

Laravel ya trae como base un sistema de seguridad que cubre los problemas más comunes en PHP, como el Cross Site Scripting. Además, la facilidad para implementar metodologías de seguridad adicionales.

➤ Orden

La estructura actual de Laravel sigue un orden lógico que permite la facilidad de lectura para cualquier programador y obliga a un nuevo desarrollador a ser ordenado en la creación de su código. Además, esta estructura asegura la agilidad en el funcionamiento de la aplicación.

➤ Actualización

Laravel está constantemente siendo actualizado, hoy en su versión 5.1, y aunque muchos programadores miran esto como un problema por la necesidad de aprender los nuevos cambios, genera mejoras inmediatas que pueden ser implementadas en la aplicación para el óptimo funcionamiento de la misma.

➤ Paquetes

Las mejores aplicaciones no han sido creadas por una sola persona, y la comunidad de Laravel no se queda para nada atrás. El framework da las herramientas básicas para iniciar, pero la comunidad ha creado una diversidad de paquetes que facilitan una gran cantidad de necesidades, desde sistemas de creación de usuarios, roles, autenticación con redes sociales, hasta paquetes sencillos como la creación de URL's óptimas para SEO.

#### **1.7. Base de Datos PostgreSQL**

Es un sistema de base de datos relacional de objetos, es de código abierto dirigido a la comunidad de desarrollo activo que ha valido en una sólida reputación de fiabilidad, robustez de las características y rendimiento (PostgreSQL, 2018).

Es frecuente que las bases de datos comerciales sean instaladas en más servidores de lo que permite la licencia. Algunos proveedores comerciales consideran a esto la principal

fuente de incumplimiento de licencia, con PostgreSQL, no hay costo asociado a la licencia del software.



Figura 11: Logo de postgresQL

Fuente: <https://www.postgresql.org> (01/07/2018)

### Ventajas de PostgreSQL

- Modelos de negocios más rentables con instalaciones a gran escala.
- No existe la posibilidad de ser auditado para verificar cumplimiento de licencia en ningún momento.
- Flexibilidad para hacer investigación y desarrollo sin necesidad de incurrir en costos adicionales de licenciamiento.
- Mejor soporte que los proveedores comerciales
- Multiplataforma PostgreSQL está disponible en casi cualquier Unix (34 plataformas en la última versión estable), y una versión nativa de Windows está actualmente con actualizaciones constantes.
- 

### 1.8. Componente PostGIS



Figura 12: Logo de PostGIS

Fuente: <https://postgis.net/> (01/07/2018)

PostGIS es un módulo que se incorpora a PostgreSQL con un soporte de objetos geográficos, convirtiéndola de esta forma en una base de datos espacial para su utilización en Sistema de Información Geográfica (Medical, 2018). Se publica bajo la Licencia Pública General de GNU.

Actualmente es la base de datos espacial de código abierto más ampliamente utilizada. Muchas y muy variadas organizaciones de todo el mundo usan PostGIS, incluyendo

agencias gubernamentales de riesgos adversos y organizaciones que almacenan terabytes de datos y sirven millones de peticiones web al día.

Debido a que está construido sobre PostgreSQL, PostGIS hereda automáticamente sus características, así como los estándares abiertos.

Características del componente:

- Permite el acceso de usuarios simultáneos
- Funciones espaciales mediante SQL y trabajar con topología
- PostGIS incluye un visor de geometrías integrado en pgAdmin 4
- Permite trabajar con triggers
- Enrutamiento
- Tipos de geometría
- Es rápido y seguro
- Sencilla importación y exportación de datos
- Permite importar y exportar datos fácilmente a través de varias herramientas conversoras (shp2pgsql, pgsq2shp, ogr2ogr, dxf2postgis).
- Clientes SIG de escritorio que trabajan con PostGIS: uDig, QGIS, mezoGIS, OpenJUMP, SpatialKit para ArcGIS, gvSIG, GRASS, ArcGIS, Manifold, GeoConcept, MapInfo, AutoCAD Map 3D, Mapserver, GeoServer, MapGuide, ArcGIS Server.

## **1.9. Metodología XP**

### **1.9.1. Introducción**

La Programación Extrema (XP), es una metodología de desarrollo formulada por Kent Beck en su libro Extreme Programming Explained (Beck, Extreme Programming Explained, 1999), su principal meta es la de ampliar la productividad al momento de desarrollar un proyecto software poniendo más énfasis en la adaptabilidad. Esta metodología ágil promueve el trabajo en equipo, logrando un aprendizaje continuo de los desarrolladores y se apoya en la retroalimentación continua entre un cliente y un determinado grupo de desarrolladores.

La simplicidad es la base de la programación extrema, significa que simplifica el diseño para dar rapidez en el desarrollo y facilitar el mantenimiento respectivo.

## 1.9.2. VALORES XP

Esta metodología presenta 5 valores básicos para todo trabajo, cada uno representa un motor para los procesos o actividades que se ejecuten en un determinado proyecto, estos valores son:

- Comunicación

En la comunicación para el grupo de programadores el código comunica mejor mientras sea más simple, en cambio cuando es complejo hay que esforzarse en hacerlo más comprensible.

El código autodocumentado es más confiable que los comentarios internos, por otro lado, las pruebas unitarias son otra manera de comunicarse ya que destalla el diseño de las clases y los métodos, mostrado ejemplos concretos en el uso de su funcionalidad.

- Simplicidad

En la simplicidad es necesaria la refactorización del código para mantener un código simple a medida que va creciendo, de la misma manera se aplica a la documentación facilitando la autoría del código en la programación por parejas, de tal forma que todo el equipo más y mejor un sistema completo.

- Retroalimentación

En esta parte el cliente es valioso puesto que su opinión respecto al proyecto lo conoce en tiempo real y le permite conocer sobre el respectivo seguimiento del mismo.

Los resultados se muestran con rapidez debido al manejo de ciclos cortos, minimizando así requisitos que no suplan y concentrándose solo en lo más importante.

### **Retroalimentación del software implementado:**

Para conocer el estado en el que se encuentra el sistema es necesario realizar pruebas unitarias y de integración.

### **Retroalimentación del cliente (usuario):**

Dentro de las pruebas de aceptación es importante la presencia del cliente y los analistas para diseñarlas conjuntamente. De esta manera se conoce el estado actual del sistema.

Dicha revisión se la realiza cada una o dos semanas, logrando que el cliente vaya guiando el proceso del desarrollo del software.

### **Retroalimentación del equipo:**

Hay ocasiones que el cliente lleva nuevos requerimientos, en ese caso el equipo puede directamente facilitar la estimación del tiempo que tomará efectuar la implementación.

- Valentía

Al momento de desarrollar un sistema, la valentía viene a ser un sinónimo de disciplina, ya que es frecuente, la presión para desafiar futuros requerimientos.

Los desarrolladores deben tener valentía para explicar sus inquietudes, miedos o experiencias, además de sentirse con la confianza entre miembros del equipo de desarrollo.

- Respeto

El respeto que se establece entre que todos los miembros del equipo o el cliente, es de vital importancia debido a que les permite asumir responsabilidades y tener autoridad sobre su propio trabajo.

En el trabajo en equipo, es importante que todos deban contribuir tratando con respeto y amabilidad a sus compañeros, demostrando respeto por las opiniones de los demás, certificando de esta forma altos niveles de lealtad y motivación con el proyecto asignado.

### **1.9.3. Roles XP**

La metodología XP detalla 4 roles que son los siguientes:

- Programador (Beck, Aceptando el cambio, 2005)
  - Nivel más alto de responsabilidad.
  - Responsable sobre la generación del código fuente.
  - Responsable sobre el diseño y maquetado de la aplicación.
  - Responsable de administrar las bases de datos.
  - Responsable sobre la integridad del sistema (pruebas).
  - Acepta críticas (código colectivo).
- Cliente (Beck, Aceptando el cambio, 2005)



- Define especificaciones.
  - Influye sin controlar.
  - Confía en el grupo de desarrollo.
  - Define pruebas funcionales.
- Encargado de Pruebas (Beck, Aceptando el cambio, 2005)
- Apoya al cliente en la preparación/realización de las pruebas funcionales.
  - Ejecuta las pruebas funcionales y publica los resultados.
- Entrenador (Beck, Aceptando el cambio, 2005)
- Es experto en la Metodología XP.
  - Responsable del proceso en su conjunto.
  - Identifica las desviaciones y reclama atención sobre las mismas.
  - Guía al grupo de forma indirecta (sin dañar su seguridad ni confianza).
  - Interviene directamente si es necesario.

#### 1.9.4. Ventajas Y Desventajas

Tabla 1: Ventajas y desventajas de la metodología de Programación Extrema (XP)

<b>VENTAJAS</b>	<b>DESVENTAJAS</b>
Programación organizada	Es eficiente en los proyectos a corto plazo
Menor tasa de errores	Altas comisiones por fallos
Satisfacción del programador	No es posible predecir condiciones a futuro
Nuevas versiones	Costoso
Adaptable a circunstancias	

*Fuente: Propia*

#### 1.9.5. Fases

La programación extrema comprende un conjunto de reglas y prácticas que ocurren en el contexto de 4 actividades: planeación, diseño, codificación y pruebas.

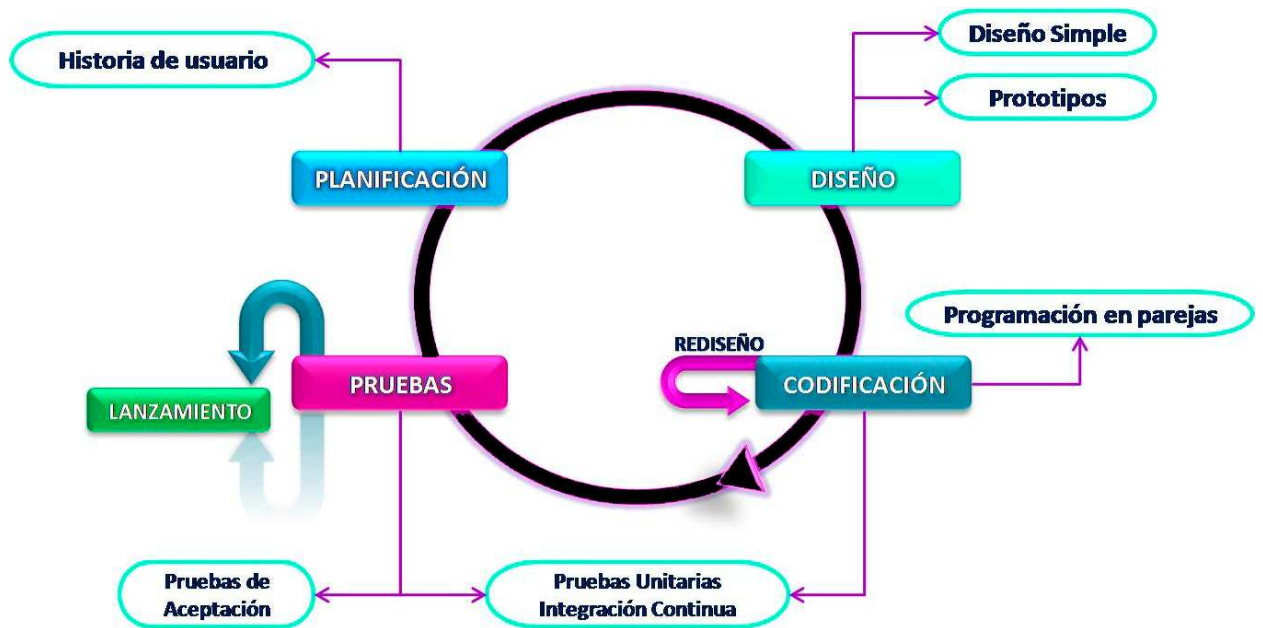


Figura 13: Esquema de las fases de la Metodología Programación Extrema (XP)

Fuente: <https://ingsoftwarekarlacevallos.wordpress.com/2015/05/08/metodologia-de-desarrollo-agil-xp-y-scrum/>

*Nota: XP: Programación Extrema*

### 1.9.5.1. Planeación

El proceso de planeación inicia con el equipo de trabajo que comprenda el contexto del negocio para el software, sus características y funcionalidades que demande el usuario.

El cliente y desarrolladores se comunican para establecer el incremento de software mediante historias de usuario, que muestran el compromiso sobre la entrega de cada uno de los requisitos, detallando además aspectos como la fecha de finalización y observaciones que se hayan establecido.

Las historias de usuario las describen los clientes y en ellas se detallan las necesidades del sistema o aplicativo, con la finalidad de verificar posteriormente que se haya efectuado el requisito correctamente.

Al tener llenado todas las historias de usuario, se procede a planificar los tiempos de implementación, formando la prioridad y versión del programa se cumplirán, se organizan dependiendo del riesgo y las iteraciones posibles que consideren el cliente y los desarrolladores.

Además, el cliente puede añadir más historias de usuario, modificar el valor de alguna o eliminarla, todo esto con el apoyo del equipo de trabajo.

### 1.9.5.2. Diseño

El diseño del sistema se limita a la simplicidad, es decir a un diseño sencillo que permita implementar las historias de usuario, permitiendo que el desarrollador suponga que cualquier otro diseño se requerirá después.

En esta fase se elige una metáfora para el sistema, esto quiere decir que es necesario tener un modelo de documentación para métodos y clases, esto proveerá la comprensión del diseño, la reutilización de código y su posterior crecimiento.

En la metodología XP recomienda pensar en el software con un contexto orientado a objetos es el uso de tarjetas CRC (clase, responsabilidad, colaborador) ya que éstas identifican y organizan objetos que son relevantes para la ampliación de software.

La estructura de la tarjeta CRC se detalla de la siguiente manera:

Tabla 2: Modelo Tarjeta CRC

<b>Nombre de la clase</b>	
<b>Responsabilidad</b>	<b>Colaboradores</b>
<i>Fuente: Propia</i>	

En la parte superior se coloca el nombre de la clase a la que pertenece el objeto, en la parte izquierda se colocan las responsabilidades u objetivos del objeto y en parte derecha las clases que ayudan con cada responsabilidad.

### 1.9.5.3. Desarrollo

En el desarrollo o codificación del sistema, XP recomienda el trabajo en parejas para la elaboración de cada historia. Mientras las parejas van terminando su trabajo, éste se integra con el resto de parejas.

En las parejas de trabajo, debe existir una dedicada a la integración del código con la finalidad de evitar inconvenientes en la compatibilidad de interfaces y lograr descubrir a tiempo los errores.

En la fase de desarrollo las optimizaciones de código se deben dejar al final y se debe respetar los horarios propuestos en la fase de planeación por cada historia de usuario.

#### **1.9.5.4. Pruebas**

Para mejorar la calidad del software es necesario realizar pruebas unitarias antes de la implantación del código y la conformación de las mismas.

Realizar test de rendimiento para proteger al software frente a fallos y en las pruebas de aceptación se realiza la evaluación del cliente.

En esta fase se incluye un análisis al servidor de alojamiento del sistema y la toma de decisiones en cuanto a las características del mismo

## CAPÍTULO II

### 2. DESARROLLO DE LA APLICACIÓN

#### 2.1. ESTRUCTURACIÓN DE APLICACIÓN BASADA EN XP

##### 2.1.1. Introducción

En el presente Plan de Desarrollo se detallarán todas las fases de desarrollo del proyecto y se facilitará la información necesaria para demostrar el funcionamiento del aplicativo de OPTIMIZACIÓN DE RUTAS PARA LA RECOLECCIÓN DE DESECHOS SÓLIDOS EN LA CIUDAD DE IBARRA PROVINCIA DE IMBABURA

Los usuarios implicados en el proyecto son 2:

- Cliente

Es la persona que expone sus requerimientos para el desarrollo del sistema, está preparada y capacitada para visualizar resultados de la aplicación y realiza sus respectivas funciones según la metodología utilizada.

- Desarrollador

Es la persona que crea el software y lo implementa, considera el alcance del proyecto en función a los requerimientos que proporcione el cliente.

##### 2.1.2. Planificación Del Proyecto

En la presente tabla, se visualiza el cronograma de las actividades planificadas.

Tabla 3: Planificación del Proyecto

*Fuente: Propia*

ETAPAS	ACTIVIDADES	PERIODO	
		Inicio	Fin
Planteamiento del Problema	Antecedentes, situación actual, planteamiento del problema	Octubre	Noviembre
Marco Teórico	Formulación de los objetivos y alcance Análisis del problema		
	Definiciones Metodología que se va utilizar Herramientas de desarrollo	Noviembre	Diciembre
Desarrollo	Plan de desarrollo	Diciembre	Enero
	Diseño Codificación Pruebas		
Conclusiones	Validación y análisis de resultados obtenidos	Enero	Febrero

### **2.1.3. Visión General**

#### **2.1.3.1. Propósito Y Funcionalidad**

Disponer de una herramienta tecnológica de optimización de las rutas de recolección de los desechos sólidos municipales, brindando un aporte significativo en la gestión segura de los desechos sólidos y de esta manera aportar a la reducción de la contaminación ambiental.

Los problemas de ruteo de vehículos tratan de determinar un conjunto de rutas para brindar un servicio a un conjunto de usuarios. Este tipo de problemas es de los más importantes en la actualidad, y además son los más estudiados dentro de los problemas de optimización de rutas.

El aporte primordial que se dará con el presente proyecto es la de resolver un problema social y ambiental de la ciudad de Ibarra, así como también mejorar la planificación de recolección de desechos de manera óptima y organizada.

Las funcionalidades del sistema web denominado “OPTIMIZACIÓN DE RUTAS PARA LA RECOLECCIÓN DE DESECHOS SÓLIDOS EN LA CIUDAD DE IBARRA PROVINCIA DE IMBABURA”, desarrollado por Fausto Lucano en aporte al proyecto de recolección y tratamiento anaerobio para la industrialización energética de los desechos sólidos:

- Búsqueda de contenedores existentes

Antes de comenzar con el ruteo, es necesario buscar información referente a contenedores y vehículos de recolección, el sistema proporcionará información de los vehículos de recorrido y contenedores debidamente localizados.

- Ruteo vehicular CVRP

El sistema calculará la ruta más óptima dependiendo el estado en el que se encuentren los contenedores en función a su volumen y peso.

- Reportes

En el reporte se establece el estado actual o tipo de historial de los contenedores que se encuentren llenos.

- Análisis de Ruteo

El sistema generará valores históricos de contenedores y rutas de acuerdo a los parámetros que el usuario considere dentro del sistema.

#### **2.1.3.2. Alcance**

Brindar un aporte significativo en la gestión de recolección segura de los desechos sólidos, a través de un aplicativo Web que permita llevar un control adecuado de contenedores y camiones de recolección, con tecnología GPRS y GSM; aportando de esta manera a la reducción de la contaminación ambiental,

Con la metodología XP permitirá desarrollar el aplicativo con arquitectura MVC y la documentación reglamentaria. Además de la utilización de herramientas de software libre con lenguaje de programación en PHP y LARAVEL como Framework.

Utilizando herramientas de Sistemas de Información Geográfica se obtendrá un mapa temático de la ciudad de Ibarra permitiendo identificar con claridad cada uno de los elementos geográficos como son los contenedores, predios y vías.

A través del modelo de ruteo vehicular permitirá determinar el conjunto de rutas de los vehículos de recolección, mejorando de esta manera la organización en el servicio.

#### **2.1.3.3. Perspectiva del Producto**

El sistema Web requiere de recursos para acceder a Internet tales como el alojamiento y disponibilidad.

Respecto a los requisitos del usuario final no es tan complejo debido a que únicamente se necesitará un browser navegador web que tenga acceso a Internet.

La utilización de dispositivos tecnológicos en contenedores permitirá determinar la capacidad en peso y volumen de biomasa existente; de la misma manera el dispositivo en vehículos de recolección les ayudará a definir las rutas específicas en función a su capacidad de carga y localización de contenedores más cercanos que se encuentren llenos.

#### **2.1.3.4. Módulos del Proyecto**

El sistema contará con los siguientes módulos:

- Usuarios

- Administración de Usuarios
- Administración de Roles.
- Vehículos
  - Registro de Chofer (tipo licencia, datos personales)
  - Registro de Vehículos Recolectores (Placa, Marca, Modelo)
  - Localización Geográfica de Vehículos.
- Contenedores
  - Registro de Contenedores (Id, color, Alto, Ancho, Peso, Volumen, Masa)
  - Georreferenciación de Contenedores.
  - Gestión del estado actual de Contenedores.
- Mapeo de Rutas
  - Ubicación e identificación de los contenedores.
  - Generación de Rutas óptimas de recolección
  - Visor de Rutas encontradas.
  - Zonificación de Gestión de Residuos Sólidos.
- Reportes
  - Frecuencia de vaciado
  - Rendimiento de recogida
  - Mapeo Temático de Contaminación Ambiental.
  - Reporte del estado actual de Vehículos y Contenedores.

#### **2.1.4. Organización del Proyecto**

##### **2.1.4.1. Participantes en el Proyecto**

Según la metodología ágil de Programación Extrema XP los partícipes del proyecto son:

- Programador: Se establecerá al Sr. Fausto Lucano, como encargado de la creación y desarrollo del aplicativo web y su respectiva implementación.
- Invitado: Usuario externo que visita el aplicativo.
- Encargado de pruebas: Se establecerá al Sr. Fausto Lucano para que realice las pruebas pertinentes de cada una de las historias de usuario.
- Entrenador: se asigna al Ing. Msc. Edgar Alberto Maya Olalla, como director del proyecto y además realizará el seguimiento de los avances del mismo.



### 2.1.4.2. Plan de fases

El desarrollo del aplicativo web se lo realizará en las 4 fases conforme a la metodología XP y se además se destinará un tiempo para cada una.

Tabla 4: Plan de Fases

<b>FASE</b>	<b>DURACIÓN</b>
<b>Planeación</b>	4 semanas
<b>Diseño</b>	4 semanas
<b>Desarrollo</b>	6 semanas
<b>Pruebas</b>	2 semanas

*Fuente: Propia*

## 2.2. FASE DE PLANEACIÓN

### 2.2.1. Planificación inicial de las iteraciones

En esta sección se detallarán algunos los requerimientos en las historias de usuario y la ordenación de las mismas en iteraciones que se exponen a continuación:

Tabla 5: Plan inicial de Iteraciones

<b>ITERACIÓN</b>	<b>DESCRIPCIÓN</b>
<b>Iteración 1</b>	Ingreso de información de chofer y vehículos en el aplicativo con sus características
<b>Iteración 2</b>	Identificación y geolocalización de contenedores de Residuos sólidos
<b>Iteración 3</b>	Generación de rutas óptimas.
<b>Iteración 4</b>	Generación de reportes y registro de seguimiento.

*Fuente: Propia*

#### Iteración 1

En esta iteración se crearán los respectivos formularios para el ingreso (creación) de choferes y vehículos con las características necesarias de geolocalización para que el aplicativo pueda ubicarlo en el respectivo mapa temático.

Además, se registrará las primeras coordenadas de partida, el chofer encargado del vehículo, mostrando al usuario como reportes.

Las Historias de Usuario para la iteración 1 se detallarán a continuación:

### Historia de Usuario 1

Se requiere establecer un responsable como chofer de un vehículo que será asignado posteriormente.

Tabla 6: *Historia de Usuario 1*

Historia de Usuario	
<b>Número:</b> 1	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Creación del usuario y formulario de chofer	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b> Creación del formulario para la inserción de choferes mismos que tienen los siguientes campos: <ul style="list-style-type: none"><li>- Cedula</li><li>- Apellidos</li><li>- Nombres</li><li>- Sexo</li><li>- Email</li><li>- Dirección</li><li>- Teléfono Local</li><li>- Teléfono Celular</li><li>- Tipo Licencia</li></ul>	
<b>Observaciones:</b> El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones. Posteriormente se van agregando otros campos adicionales.	

*Fuente: Propia*

### Historia de Usuario 2

La historia 2 permite ingresar los vehículos de recolección a la base de datos con las características esenciales y configuración inicial del mismo.

Tabla 7: *Historia de Usuario 2*

Historia de Usuario
---------------------

---

<b>Número:</b> 2	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Creación del aplicativo y formulario de vehículos	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b>	
Creación del formulario para el ingreso de vehículos recolectores con los siguientes campos importantes:	
<ul style="list-style-type: none"> <li>- Propietario</li> <li>- Chofer</li> <li>- Placa</li> <li>- Marca</li> <li>- Modelo</li> <li>- Cilindraje</li> <li>- Capacidad de Volumen (Litros)</li> <li>- Capacidad de Peso (Kilogramos)</li> <li>- Capacidad de Carga Nominal Máxima (Kilogramos)</li> </ul>	
<b>Observaciones:</b>	
El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones.	
Posteriormente se agregarán más campos adicionales	

---

*Fuente: Propia*

### Historia de Usuario 3

En esta historia el usuario visualiza las coordenadas actuales de inicio y de fin que recorre cada vehículo recolector.

Tabla 8: *Historia de Usuario 3*

---

<b>Historia de Usuario</b>	
<b>Número:</b> 3	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Creación de la vista del recorrido de cada vehículo	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b>	
Creación de una vista que permite visualizar el recorrido del vehículo, su distancia y tiempo empleado:	
<ul style="list-style-type: none"> <li>- Placa del Vehículo</li> <li>- Fecha y Hora</li> <li>- Latitud</li> <li>- Longitud</li> <li>- Distancia</li> <li>- Tiempo</li> <li>- Estado</li> </ul>	
<b>Observaciones:</b>	
El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones.	
Posteriormente se agregarán más campos adicionales	

---

Fuente: Propia

### 2.2.2. Iteración 2

En esta iteración, el usuario conoce la ubicación de los contenedores de los desechos sólidos en la ciudad de Ibarra.

Ingresará además los parámetros de coordenadas, volumen en litros y la caga total de cada contenedor.

La información de cada contenedor se organizará dependiendo de 4 factores que se considera en la configuración, para su posterior análisis:

- Latitud.
- Longitud.
- Volumen(litros).
- Carga (kilogramos).

### Historia de usuario 4

En esta iteración se necesita un formulario con los parámetros de las características de contenedores.

Tabla 9: *Historia de Usuario 4*

Historia de Usuario	
<b>Número:</b> 4	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Creación de un formulario para ingresar la configuración de contenedores	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b>	
El formulario de contenedores cuenta con los siguientes campos:	
- Código	
- Descripción	
- Latitud	
- Longitud	
- Volumen	
- Peso	
- Carga	
- Largo	
- Ancho	
- Alto	
<b>Observaciones:</b>	
El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones.	

---

Posteriormente se van agregando otros campos adicionales

---

*Fuente: Propia*

### Historia de usuario 5

En esta historia se realiza una vista con mapa base y geolocalización de contenedores.

Tabla 10: *Historia de Usuario 5*

---

Historia de Usuario	
<b>Número:</b> 5	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Creación de una vista de geolocalización con mapa base	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b> creación de una vista que contenga las opciones de geolocalización con los siguientes campos: - ID Contenedor - Coordenadas actuales (latitud, longitud) - Cantidad de Volumen actual(litros) - Estado - Novedad - Observación	
<b>Observaciones:</b> El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones. Posteriormente se agregarán más campos adicionales	

---

*Fuente: Propia*

### Historia de usuario 6

En la historia se creará el formulario de reporte actual y visualización del estado actual de cada contenedor.

Tabla 11: *Historia de Usuario 6*

---

Historia de Usuario	
<b>Número:</b> 6	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Creación de formulario de reporte actual y estado	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b> Creación del formulario de reporte y estado actual de los contenedores con los siguientes campos: - Valor actual del volumen acumulado de desechos sólidos - Valor actual del peso - Valor actual de la carga en kilogramos. - Estado actual a través puntos localizados de colores	
<b>Observaciones:</b> El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo	

---

---

validaciones.

Posteriormente se agregarán más campos adicionales

---

*Fuente: Propia*

### 2.2.3. Iteración 3

El usuario final obtiene un mapa con los puntos de cada contenedor, así como también la ubicación actual del vehículo recolector. Basándose en varios valores de volumen, peso y carga, el aplicativo logrará realizar el cálculo de las rutas en función a los parámetros establecidos:

- Dato teórico: Se basa en los valores iniciales de cada contenedor (longitud, latitud y carga).
- Dato por seguimiento: Se obtiene el estado actual, dependiendo de sus valores de carga.

### Historia de usuario 7

Esta historia detalla el ingreso de varios parámetros de cálculo necesarios para el registro rutas.

El ingreso de rutas será en una tabla dinámica que el usuario podrá visualizar fácilmente los parámetros.

Tabla 12: *Historia de Usuario 7*

---

Historia de Usuario				
<b>Número:</b> 7	<b>Usuario:</b> admin@utn.com			
<b>Nombre Historia:</b> Creación de una tabla dinámica de puntos para el cálculo de rutas				
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media			
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1			
<b>Programador responsable:</b> Fausto Lucano				
<b>Descripción:</b>	Creación de una tabla de valores dinámicos para la generación de rutas:			
<b>Id</b>	<b>origen</b>	<b>destino</b>	<b>distancia (kilómetros)</b>	<b>tiempo</b>
<b>Observaciones:</b>				
El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones.				
Posteriormente se van agregando otros campos adicionales				

---

Fuente: Propia

## Historia de usuario 8

Luego del ingreso de los parámetros de cálculo, se creará una vista de visualización del estado actual de vehículos, contenedores y rutas

Tabla 13: *Historia de Usuario 8*

Historia de Usuario	
<b>Número:</b> 8	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Creación de una vista de visualización del estado actual de vehículos, contenedores y rutas	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b> Creación de la vista general del estado actual de vehículos y contenedores para la determinación de rutas optimas de recolección.	
<b>Observaciones:</b> El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones. Posteriormente se van agregando otros campos adicionales	

**Fuente:** Propia

## Historia de usuario 9

En esta historia de usuario se pretende construir una tabla dinámica similar a la historia 7 para el ingreso del detalle de costos tiempo por cada ruta. Esto para conseguir un total de costos por seguimiento.

Tabla 14: *Historia de Usuario 9*

Historia de Usuario					
<b>Número:</b> 9	<b>Usuario:</b> admin@utn.com				
<b>Nombre Historia:</b> Creación de tablas dinámicas de cálculo de rutas óptimas					
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media				
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1				
<b>Programador responsable:</b> Fausto Lucano					
<b>Descripción:</b> Diseño de dos tablas dinámicas que permite detallar los nodos con distancia, tiempos y carga					
<b>Id</b>	<b>origen</b>	<b>destino</b>	<b>distancia (km)</b>	<b>tiempo</b>	<b>Capacidad de carga</b>
Ruta#					
		<b>Total</b>	-	-	-
Rutas generadas a partir del análisis de carga de cada contenedor					
<b>Id</b>	<b>origen</b>	<b>destino</b>	<b>distancia (km)</b>	<b>tiempo</b>	<b>Capacidad de carga</b>
<b>Observaciones:</b> El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones.					

---

Posteriormente se van agregando otros campos adicionales

---

*Fuente: Propia*

### Historia de usuario 10

Posterior al registro de tiempos variables, se procede al análisis de tiempos totales del vehículo, para esto, se necesita realizar un cálculo general de distancias y horas para mostrarlo al usuario en forma de tabla.

Tabla 15: *Historia de Usuario 10*

---

<b>Historia de Usuario</b>	
<b>Número:</b> 10	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Cálculos para el análisis de rutas y creación de vista de rutas	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b>	
Diseño de la vista para mostrar los cálculos realizados en donde se determina:	
- Cantidad de Rutas generadas	
- Capacidad actual de carga del vehículo	
- Distancia empleadas	
- Tiempos de cada ruta	
<b>Observaciones:</b>	
El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones.	
Posteriormente se van agregando otros campos adicionales	

---

*Fuente: Propia*

### 2.2.4. Iteración 4

En esta iteración, se entrega al usuario reportes del estado de vehículos y contenedores en formatos manejables y exportables con gráficos estadísticos.

### Historia de usuario 11

La actual historia de usuario describe la creación de un reporte gráfico para obtener las rutas establecidas en función al estado de los contenedores.

Tabla 16: *Historia de Usuario 11*

---

<b>Historia de Usuario</b>	
<b>Número:</b> 11	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Creación de un reporte gráfico para obtener las rutas establecidas en función al estado de los contenedores	

---



---

<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b>	
Diseño del reporte grafico estadístico considerando las variables de distancias y tiempos:	
- Estado	
- Marca	
- Largo	
- Ancho	
- Alto	
- Cilindraje	
<b>Observaciones:</b>	
El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones.	
Posteriormente se van agregando otros campos adicionales	

---

*Fuente: Propia*

## Historia de Usuario 12

En los reportes gráficos la historia de usuario describe la visualización de macrozonas para la gestión de Residuos Sólidos en la ciudad de Ibarra

Tabla 17: *Historia de Usuario 12*

<b>Historia de Usuario</b>	
<b>Número:</b> 1	<b>Usuario:</b> admin@utn.com
<b>Nombre Historia:</b> Creación de un reporte temático en donde se visualiza las macrozonas de Gestión de Residuos Sólidos	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fausto Lucano	
<b>Descripción:</b>	
Creación de una vista que contiene información referente a las macrozonas obtenidas como resultado del ruteo para la Gestión de Residuos Sólidos, en la vista se muestra información referente a:	
<ul style="list-style-type: none"> <li>• Cantidad de Biogas en la ciudad.</li> <li>• Cantidad de Macrozonas para la Gestión de Residuos Sólidos.</li> <li>• Mapeo Temático de Contaminación Ambiental</li> <li>• Reporte estadístico de Cantidades Recolectadas de Desechos Sólidos al mes y año.</li> </ul>	
<b>Observaciones:</b>	
El formulario es totalmente funcional con las operaciones de inserción y actualización incluyendo validaciones.	
Posteriormente se van agregando otros campos adicionales	

---

*Fuente: Propia*

## 2.3. FASE DE DISEÑO

En la fase de diseño se detalla procesos del aplicativo usando diagramas de flujo y la arquitectura general del componente de optimización de las rutas de recolección de los desechos sólidos.

Además, se describen las actividades de usuarios posibles de cada uno en el aplicativo.

### 2.3.1. Diagramas de Casos de Uso

El diagrama de casos de uso es una herramienta que permite resumir el uso de la aplicación o sistema por parte de los usuarios. Ayuda además a la descripción de los requisitos del usuario y la relación entre estos. (Jacobson, 1992).

Los diagramas describen 4 elementos básicos que son:

Tabla 18: *Elementos básicos del Diagrama Casos de Uso*

ELEMENTO	DESCRIPCIÓN
<b>Actor</b>	Es el usuario que interactúa directamente con el aplicativo.
<b>Caso de Uso</b>	Son los procesos que realiza un actor para lograr un objetivo.
<b>Asociación</b>	Es el actor que forma parte de un caso de uso, relacionando el uno al otro, con una flecha.
<b>Subsistema o componente</b>	Es una parte del mismo en el que se está trabajando, además indica la interacción que tiene entre el sistema y los actores.

*Fuente: (Jacobson, 1992)*

#### 2.3.1.1. Actores del Proceso

##### - Cliente

El cliente es un usuario que tiene privilegios de administrar el vehículo recolector con sus características que luego se mostrará al público.

Tiene conocimientos acerca de la capacidad de carga que tiene el vehículo e ingresa al sistema con el objetivo de configurar los parámetros esenciales que serán usados en la aplicación para realizar consultas.

Tiene privilegios sobre los formularios principales para su vehículo, además posee permisos especiales para modificar los datos de cada contenedor.



Figura 14: Diagrama de caso de uso general

Fuente: Propia

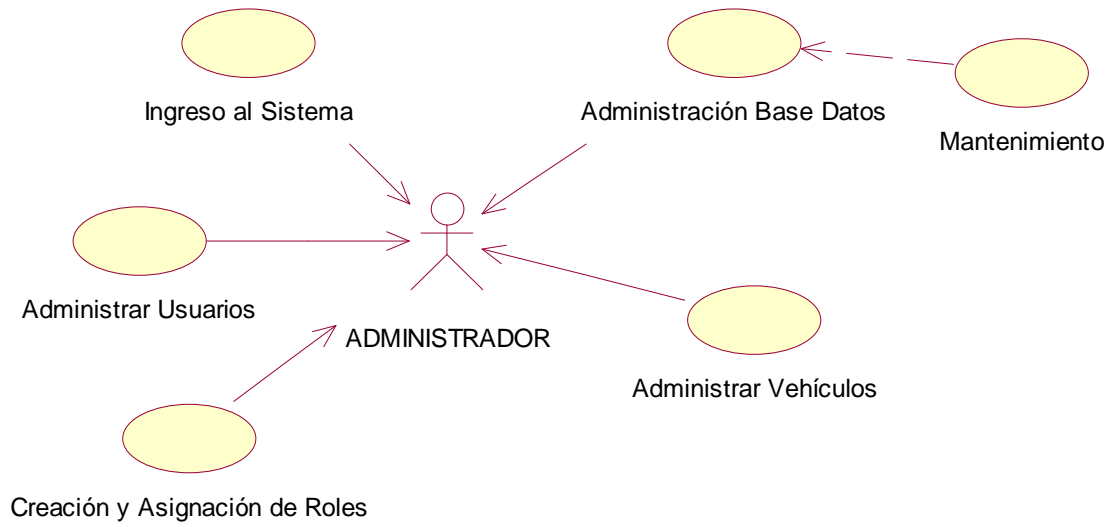


Figura 15: Diagrama de caso de uso del rol de Administrador

Fuente: Propia

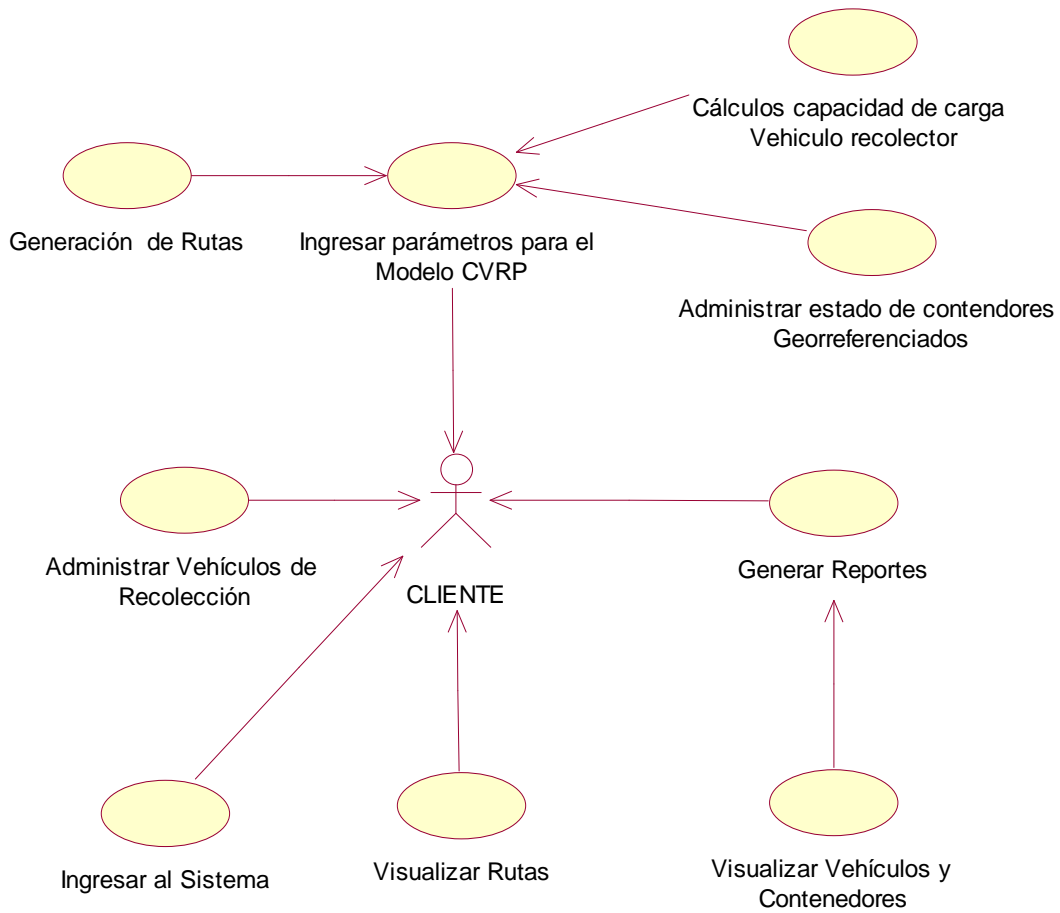


Figura 16: Diagrama de caso de uso del rol de Cliente

Fuente: Propia

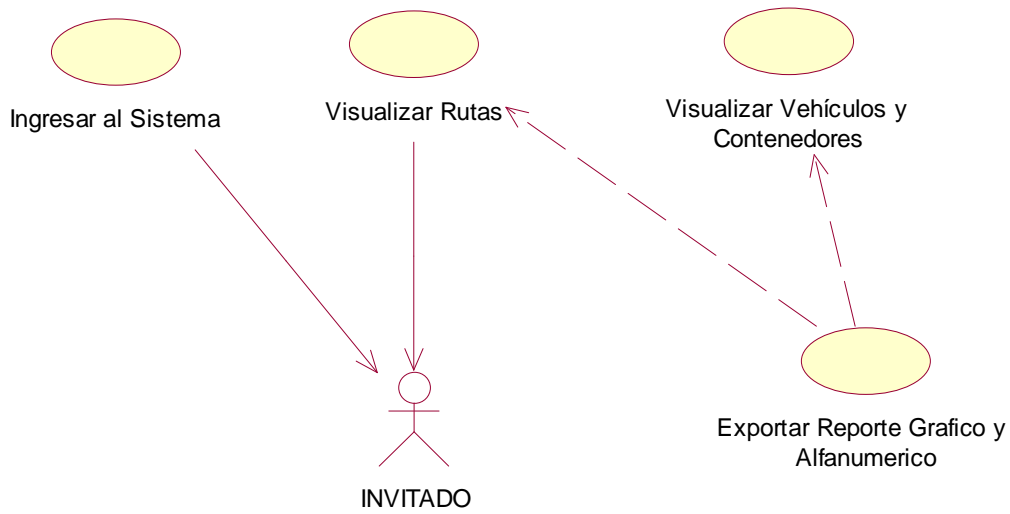


Figura 17: Diagrama de caso de uso del rol de Invitado

Fuente: Propia

## 2.3.2. Diagramas de Proceso

### 2.3.2.1. Diagrama de Creación de Usuario

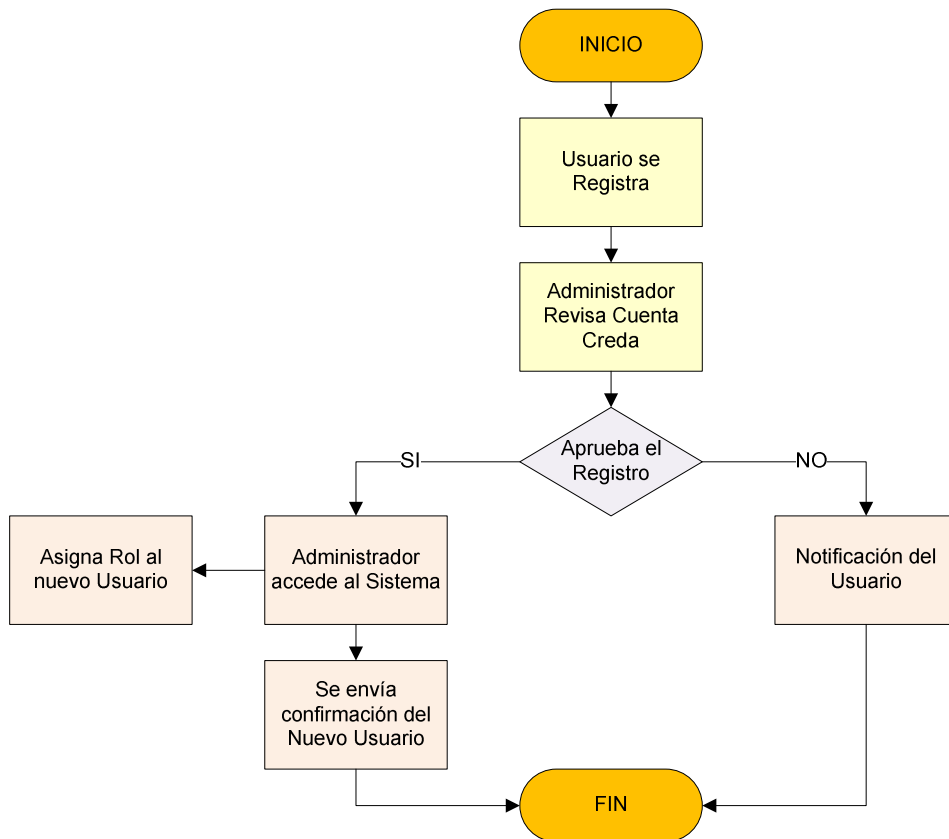


Figura 18: Diagrama de la creación de usuario

Fuente: Propia

### 2.3.2.2. Diagrama de ingreso y configuración de vehículos de recolección

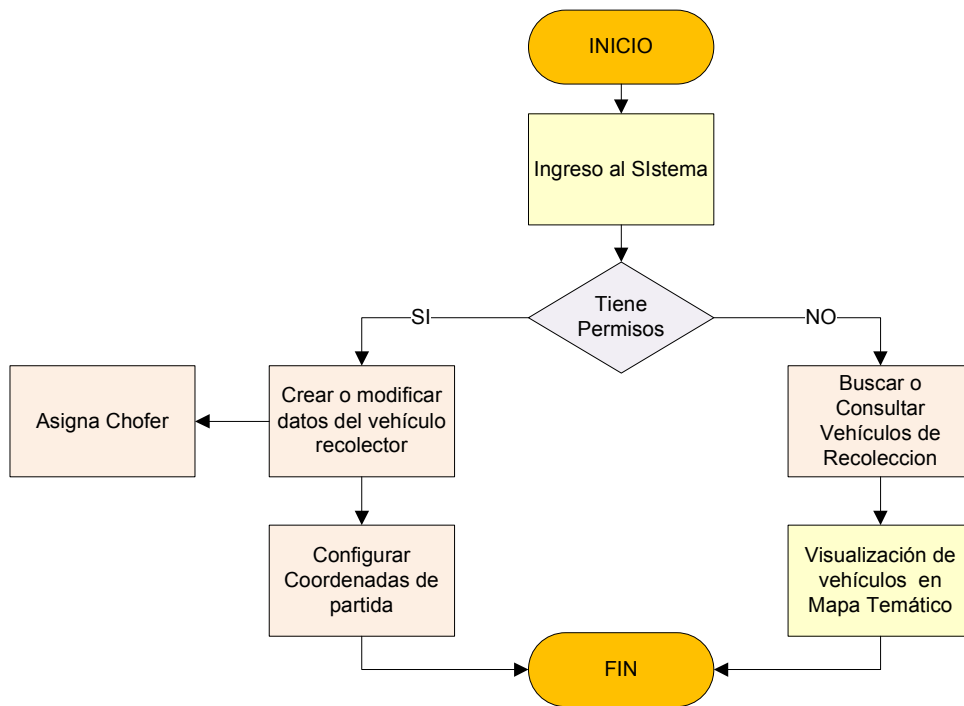


Figura 19: Diagrama de ingreso y configuración de vehículos recolectores

Fuente: Propia

### 2.3.2.3. Diagrama de ingreso, configuración y georreferenciación de contenedores

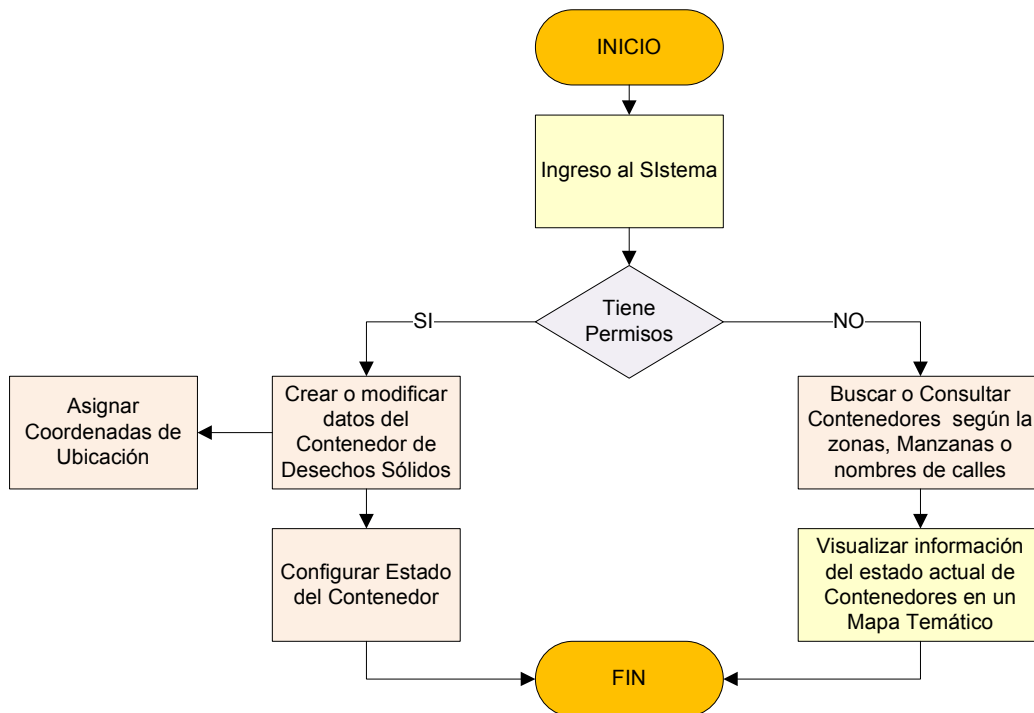


Figura 20: Diagrama de ingreso, configuración y georreferenciación de contenedores

Fuente: Propia

### 2.3.2.4. Diagrama de procesamiento y determinación de rutas

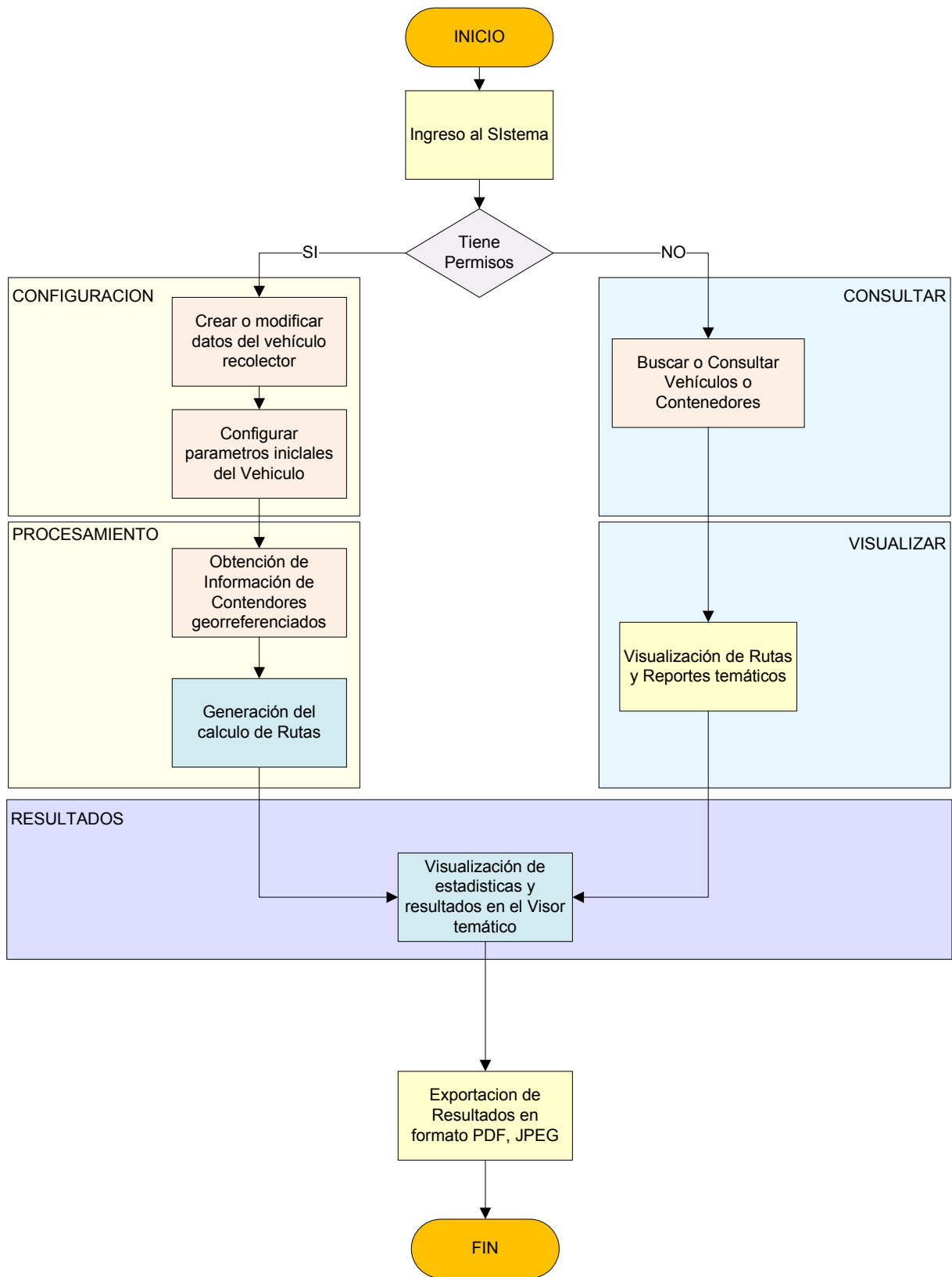


Figura 21: Diagrama de Procesamiento y Determinación de rutas

Fuente: Propia

### 2.3.3. Arquitectura del sistema web

#### 2.3.3.1. Diagrama de Arquitectura del Sistema

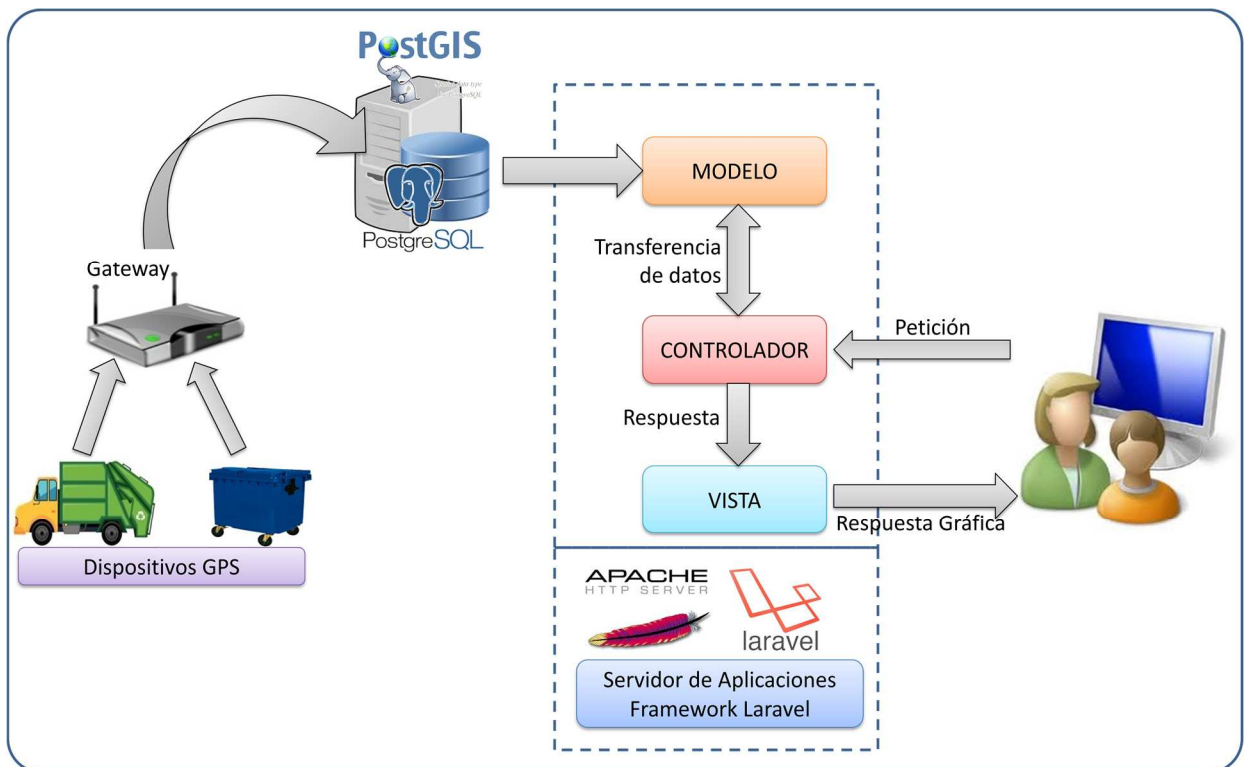


Figura 22: Diagrama de la Arquitectura del aplicativo Web

Fuente: Elaboración Propia

#### 2.3.3.2. Descripción del aplicativo web

El Aplicativo Web se encuentra desarrollado sobre un lenguaje PHP, utilizando el framework de Laravel, mismo que incluye librerías de JavaScript y CSS (jQuery y Bootstrap).

Los lenguajes y tecnologías son de libre acceso por lo que ayudan a realizar mejoras a futuro y un mantenimiento constante del aplicativo sin restricciones.

El sistema se encarga de realizar operaciones que se describen en el alcance del proyecto.

#### 2.3.3.3. Descripción del servidor web y flujo de datos

El Sistema Web estará alojado en un Servidor Web remoto con ip pública y estará disponible las 24 horas.



Respecto al funcionamiento del Aplicativo Web es necesario un Servidor Apache con motor PHP y un servidor de Base de Datos PostgreSQL.

Apache es principalmente usado para utilizar páginas web estáticas y dinámicas en la www (World Wide Web).

Apache es un servidor web del sistema XAMPP, con el componente de la base de datos PostgreSQL y los respectivos lenguajes de programación PHP/Perl/Python, en Windows: se trabaja con XAMPP, mientras que en Linux tenemos un similar llamado LAMP, y posee las características esenciales para los aplicativos webs.

Las actividades para cualquier acción que se realice en el aplicativo es:

- a. El servidor apache recibe la petición del navegador para posteriormente dirigir todas las peticiones al motor principal de PHP.
- b. El servidor apache que contiene al archivo PHP le proporciona una salida directa convirtiéndola a HTML en caso de solicitar datos a la Base de Datos se procede a realizar las respectivas consultas.
- c. Si se solicita realizar una consulta a la base de datos, el gestor del (DBMS) PostgreSQL responde con un resultado.
- d. El motor PHP se encarga de conformar la salida HTML que se presenta al cliente en su navegador.
- e. Los códigos JavaScript que se ejecutan del lado del cliente son procesadas en forma local, y permiten interactuar con eventos que necesite el usuario además de refrescar páginas.

#### **2.3.3.4. Módulos del sistema**

El sistema tiene 5 módulos importantes:

##### **Administración de Usuarios**

El sistema permite gestionar la creación, edición y asignación de roles con privilegios a los usuarios.

En este módulo se administra Operaciones, Tareas y Roles que el sistema va registrando conforme se vayan creando los controladores y las vistas.

##### **Vehículos**

Este módulo se destina a la administración de los vehículos de recolección de desechos sólidos almacenados en la base de datos, así como también el registro del respectivo chofer asignado en cada vehículo.

Tienen un CRUD básico (Create, Read, Update and Delete) con sus vistas respectivas, estas se ubican en un menú principal y son accesibles por los Clientes y Administradores.

Dentro de este módulo se tiene:

- Registro de Chofer (tipo licencia, datos personales)
- Registro de Vehículos Recolectores (Placa, Marca, Modelo)
- Localización Geográfica de Vehículos.

### **Contenedores**

En este módulo es posible realizar el registro y georreferenciación de cada contenedor, mostrando la información en un visor geográfico, además de administrar el estado actual de cada contenedor.

Dentro de este módulo se considera:

- Registro de Contenedores (Id, color, Alto, Ancho, Peso, Volumen, Masa)
- Georreferenciación de Contenedores.
- Gestión del estado actual de Contenedores.

### **Mapeo de Rutas**

El actual módulo registra la ubicación de cada contenedor generación de todos los cálculos que se realizan con valores almacenados para la determinación de rutas óptimas.

Además, permite visualizar el análisis de cada ruta generada como resultado del ruteo.

Permite realizar el seguimiento al vehículo recolector por cada recorrido, de esta manera mantiene informado al usuario de los estados de cada vehículo.

Este módulo está organizado para procesar la siguiente información:

- Ubicación e identificación de los contenedores.
- Generación de Rutas óptimas de recolección
- Visor de Rutas encontradas.
- Zonificación de Gestión de Residuos Sólidos.

### **Reportes**

Este módulo posee funcionalidades que dan al usuario la posibilidad de exportarlos y utilizarlos como considere conveniente.

Los reportes que el aplicativo presenta son:

- Frecuencia de vaciado
- Rendimiento de recogida
- Mapeo Temático de Contaminación Ambiental.
- Reporte del estado actual de Vehículos y Contenedores.

## **2.4. FASE DE CODIFICACIÓN E IMPLEMENTACIÓN**

### **2.4.1. Diagramas de base de datos**

#### **2.4.1.1. Introducción a la base de datos**

En la fase según la metodología XP, se empieza presentando la Base de Datos que se utilizó en la OPTIMIZACIÓN DE RUTAS PARA LA RECOLECCIÓN DE DESECHOS SÓLIDOS EN LA CIUDAD DE IBARRA PROVINCIA DE IMBABURA.

El diseño de la Base de Datos está organizado por tablas, separando la parte operativa de los catálogos y de la seguridad, además se detallará cada grupo de tablas, explicando la funcionalidad y lógica dentro del aplicativo web.

Los módulos de la base de datos son:

- Usuarios
- Vehículos
- Contenedores
- Mapeo de Rutas
- Reportes

#### **2.4.1.2. Módulo Usuarios**

Este módulo permite el control de usuarios y roles conjuntamente con la utilización de Laravel Framework permitirá al usuario:

- Actualización visual de los Roles
- Registrar el login y actualizar el password
- Gestión avanzada de usuarios
- Campos de perfil personalizables en línea
- Uso avanzado de correos basados en vistas

El módulo es parte de una API completa para el framework utilizado, por tanto, hace posible avanzados controles de Programación Orientada a Objetos para gestionar sesiones, monitorear, controlar eventos y más. (Salazar, 2014)

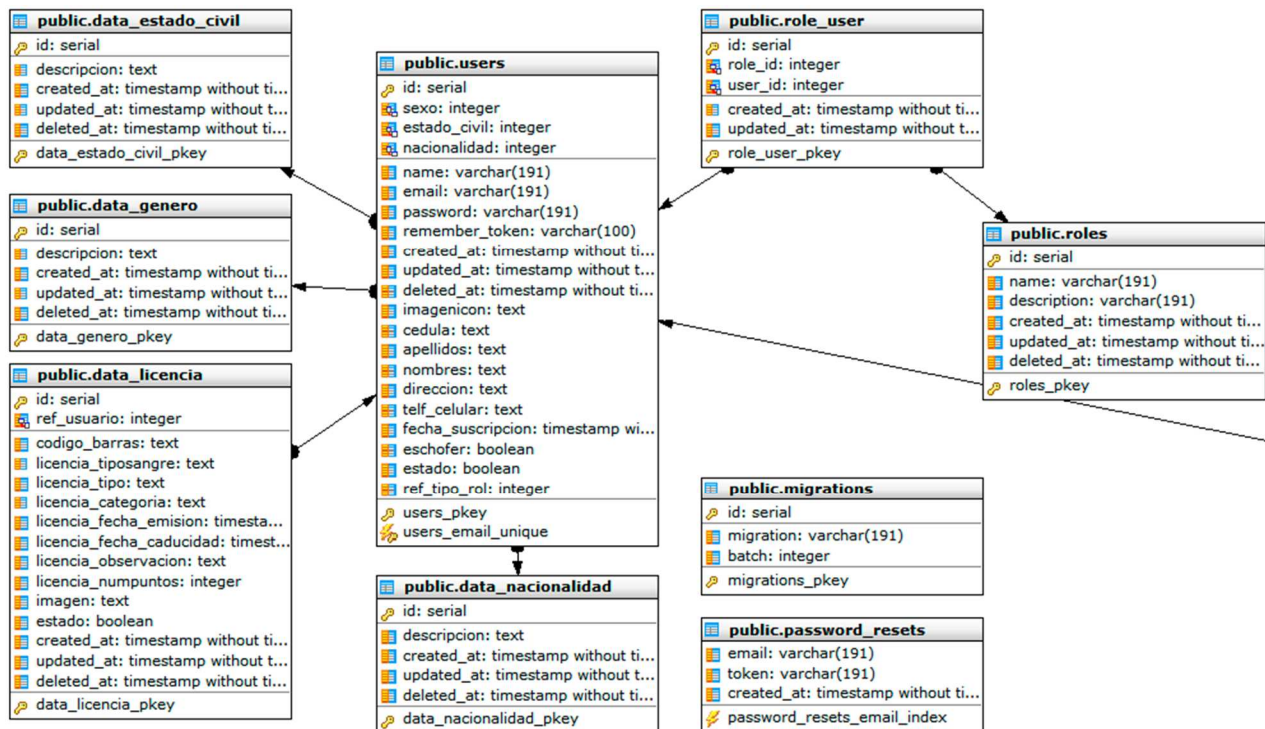


Figura 23: Tablas de BD módulo Usuarios

Fuente: Propia

**users:** Esta tabla guarda información de la persona que se ha registrado en el aplicativo.

Tiene los campos estrictos por Laravel de un usuario, email y contraseña, pero además almacena datos adicionales como los datos personales de cada usuario como el nombre apellido y cedula

El campo llamado “password” almacena la contraseña encriptada.

**roles:** Tabla que almacena los privilegios del aplicativo respecto a los usuarios y sus sesiones.

Además, permite al administrador bloquear o desbloquear el intento de iniciar el sistema por cualquier usuario.

**role\_user:** Esta tabla permite relacionar usuarios y roles con una característica de varios permisos que se les asigna varios usuarios.

**data\_estado\_civil:** Esta tabla permite ingresar o actualizar es Estado Civil de una persona.

**data\_genero:** Esta tabla permite ingresar o actualizar el Género de una persona.

**data\_licencia:** Esta tabla permite almacenar las características de la licencia que tiene el usuario chofer.

**data\_nacionalidad:** Esta tabla permite ingresar o actualizar la Nacionalidad de una persona.

### 2.4.1.3. Módulo Vehículos

En el módulo de vehículos tenemos la información constante proporcionada por las características cada vehículo en especial la capacidad de carga.

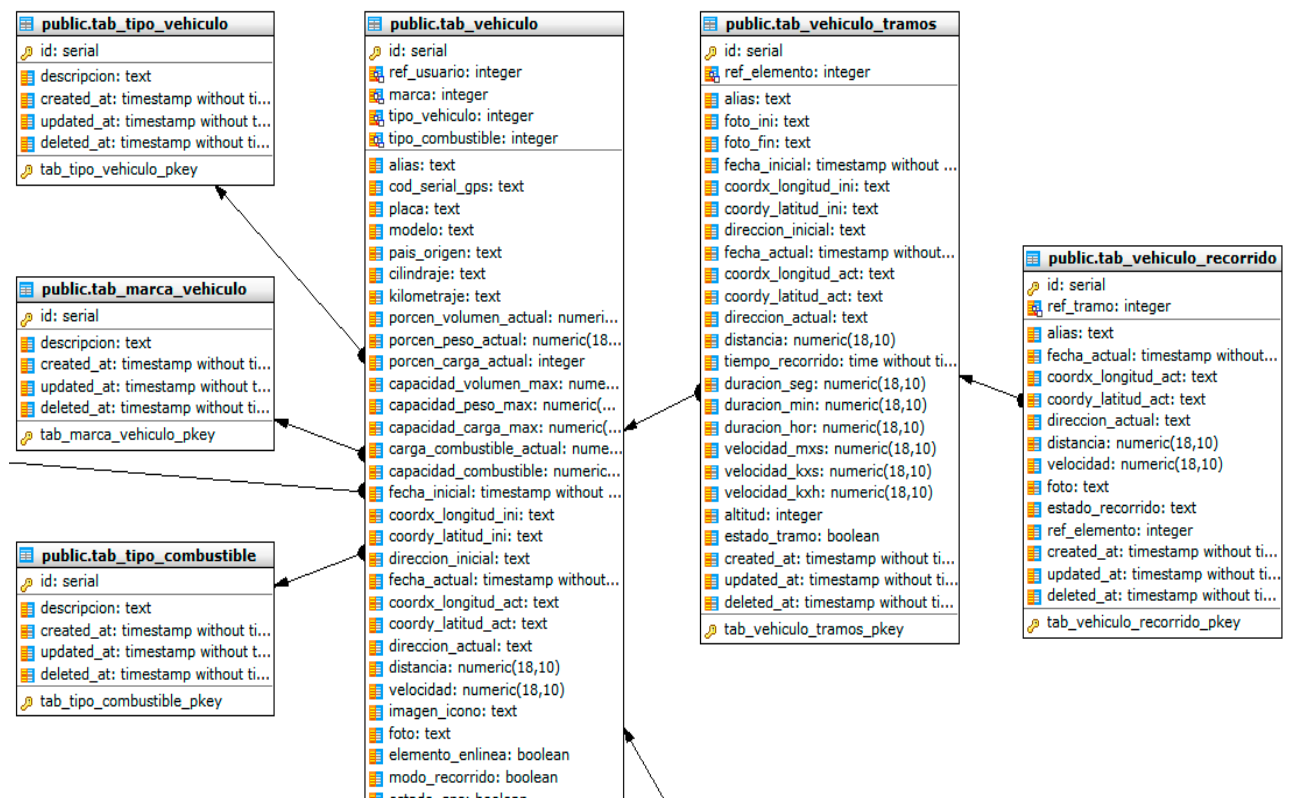


Figura 24: Tablas del módulo de vehículos recolectores

Fuente: Propia

**tab\_vehiculo:** Contiene información de las características del vehículo recolector, y sus datos no son modificables ya que son datos de fábrica.

**tab\_marca\_vehiculo:** Esta tabla tiene un catálogo de marcas de vehículos que ahorrarán al usuario el ingreso de información al momento de registrar un vehículo.

**tab\_tipo\_combustible:** Esta tabla almacena el tipo de combustible que utilizan los vehículos.

**tab\_tipo\_vehiculo:** Esta tabla almacena los tipos de vehículos.

**tab\_vehiculo\_tramos:** Esta tabla permite registrar el movimiento del vehículo desde una coordenada inicial hacia una coordenada final.

**tab\_vehiculo\_recorrido:** Esta tabla permite registrar las diferentes ubicaciones en donde se encuentra el vehículo con la finalidad de realizar cálculos de distancias recorridas.

#### 2.4.1.4. Módulo Contenedores

En el presente módulo administra todo lo referente a los contenedores de residuos sólidos de la ciudad de Ibarra identificados en este proyecto.

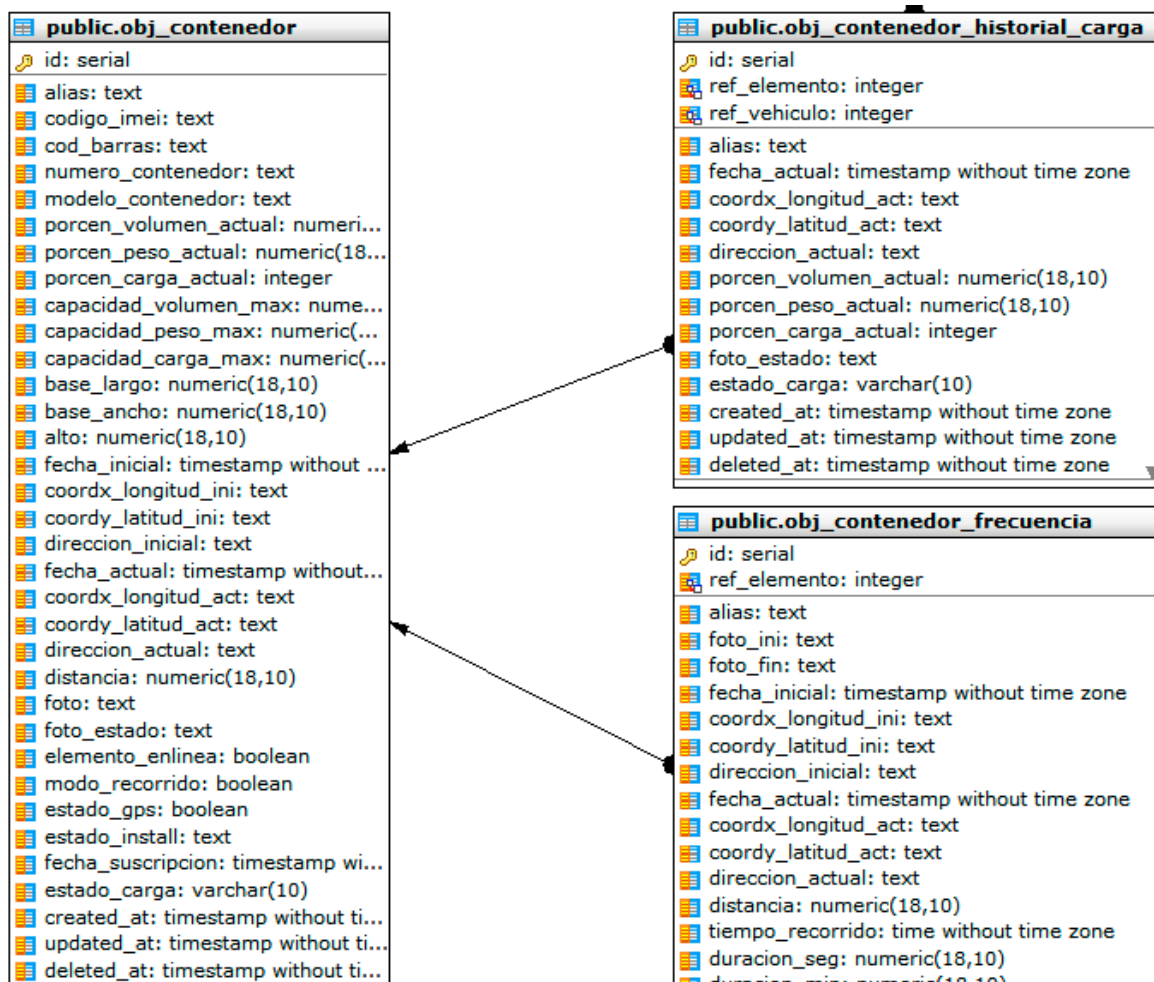


Figura 25: Tablas del módulo de contenedores

Fuente: Propia

**obj\_contenedor:** Esta tabla permite almacenar las características de los contenedores, su localización, carga actual, color, largo, ancho y alto.

**obj\_contenedor\_frecuencia:** Esta tabla permite almacenar las diferentes frecuencias de recolección con la finalidad de registrar la cantidad distancia y tiempos empleados en las diferentes mediciones de cada contenedor.

**obj\_contenedor\_historial\_carga:** Esta tabla permite almacenar el historial de carga y descarga de los contenedores medido en tiempo y distancia, esta información permitirá determinar resúmenes de recolección diaria, mensual y anual.

#### 2.4.1.5. Módulo Mapeo de Rutas

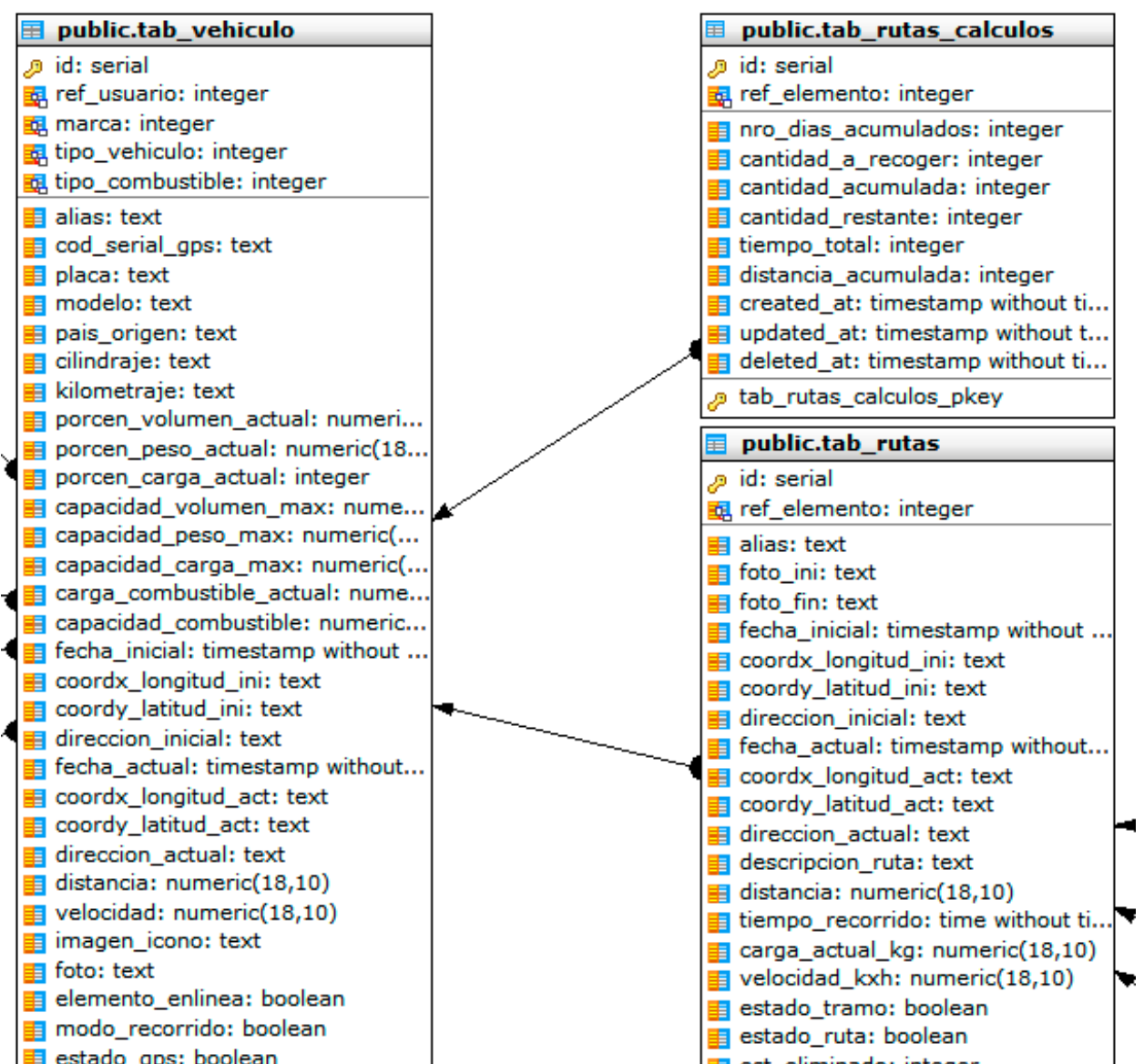


Figura 26: Tablas de BD módulo Mapeo de Rutas

Fuente: Propia

**tab\_rutas:** Esta tabla permite almacenar las rutas óptimas identificadas en una serie de cálculos realizados en función al estado de cada contenedor, además identifica cada la distancia empleada, así como también el tiempo.

**tab\_rutas\_calculos:** Esta tabla permite almacenar totales acumulados de todos los tramos recorridos óptimos que han sido guardados dentro de la tabla de ruteo.

En estas tablas se almacena lo que puede ser descrito en su forma más sencilla como una flota de vehículos con capacidades uniformes que tiene que satisfacer la demanda de un grupo de clientes a través de un conjunto de rutas que empiezan y terminan en un almacén común y que representan el menor costo posible, así como la identificación del orden de visita a los mismos.

#### 2.4.1.6. Módulo Reportes

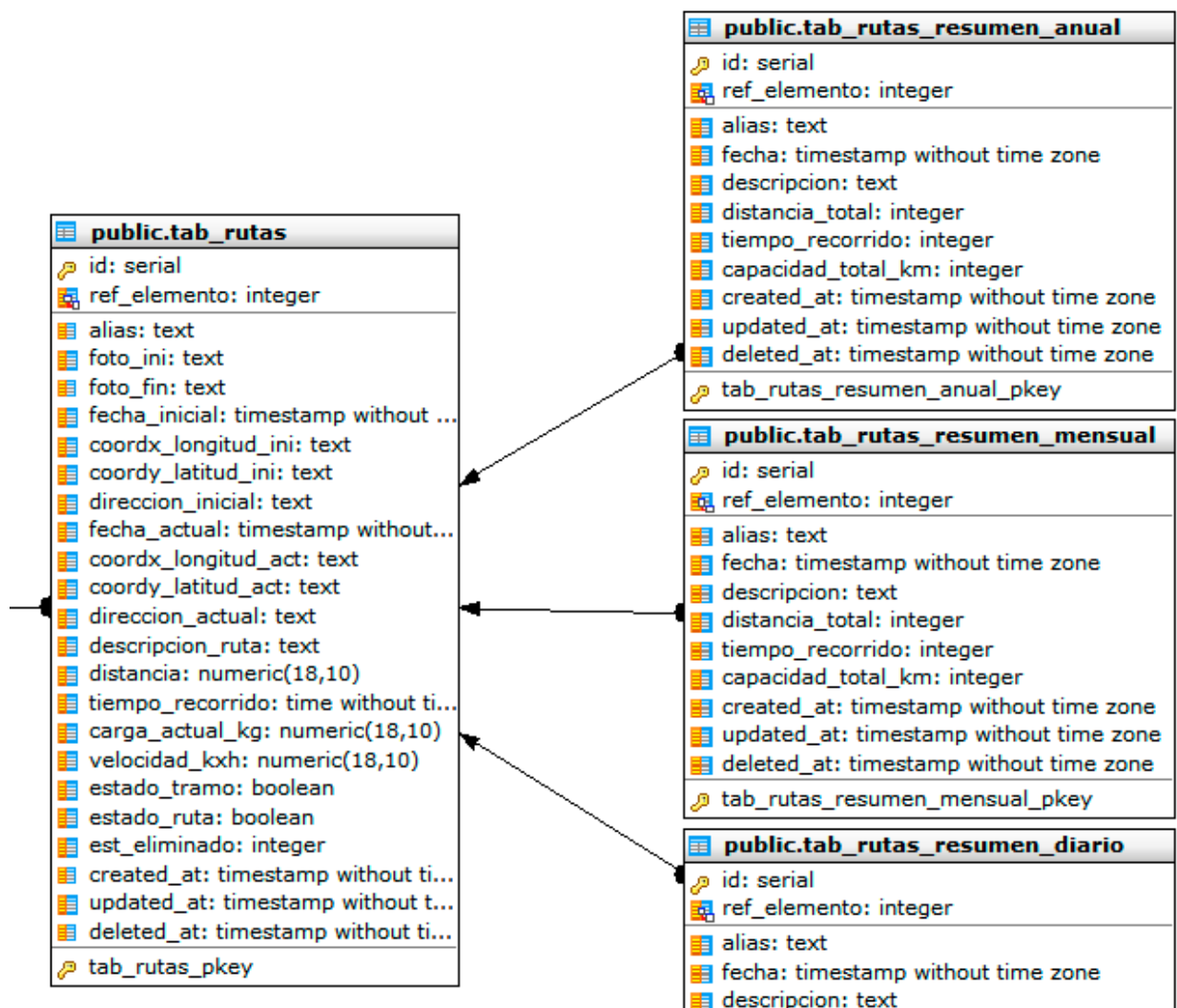


Figura 27: Tablas de BD módulo Reportes

Fuente: Propia



**tab\_rutas:** Esta tabla permite almacenar las rutas optimas identificadas en el trayecto

**tab\_rutas\_resumen\_diario:** Esta tabla permite almacenar Resumen Diario de Recolección

**tab\_rutas\_resumen\_mensual:** Esta tabla permite almacenar Resumen Mensual de Recolección.

**tab\_rutas\_resumen\_anual:** Esta tabla permite almacenar Resumen Anual de Recolección

Este tipo de resúmenes permite identificar la cantidad de Biomasa que ha venido siendo acumulada en un determinado tiempo potencializando de esta manera el aprovechamiento del material para múltiples propósitos investigativos.

## 2.5. FASE DE PRUEBAS

En esta fase se describe un conjunto de pruebas funcionales a nivel de iteraciones, cada una posee su característica descriptiva, condiciones de ejecución, datos de entrada y el resultado esperado.

Las iteraciones son las siguientes:

- Iteración 1: Ingreso de información de chofer y vehículos en el sistema con sus características
- Iteración 2: Identificación y geolocalización de contenedores de Residuos sólidos
- Iteración 3: Generación de rutas óptimas.
- Iteración 4: Generación de reportes y registro de seguimiento.

### 2.5.1. Ingreso de información de chofer y vehículos en el sistema con sus características

ID	PLACA	MARCA	MODELO
1	IBX0576	2	1999
2	PIMM0576	2	1999
3	KMM0876	2	1999

Datos Vehiculo		Capacidad del Vehiculo	
Chofer *	<input type="text"/>	Volumen(litros)	<input type="text" value="7500"/>
Alias	<input type="text"/>	Peso(Kg)	<input type="text" value="5000"/>
Placa	<input type="text"/>	Carga nominal(Kg)	<input type="text" value="10000"/>
Marca *	<input type="text"/>	Subir Foto	
Modelo	<input type="text"/>	Arrastrar y soltar aqui los archivos o Hacer Click aqui >	
cilindrada	<input type="text"/>	<input type="button" value="Subir Foto"/>	
Pais Origen	<input type="text"/>	<input type="button" value="Guardar &lt;&lt;&lt;"/> <input type="button" value="Cancelar &lt;&lt;&lt;"/>	

Figura 28: Formulario de configuración del vehículo

Fuente: Propia

## Descripción:

Una vez que el usuario a ingresado al sistema accederá al ítem del menú “Vehículos” y creará un nuevo vehículo recolector con las respectivas especificaciones técnicas de carga del contenedor, además de ingresar los datos del chofer responsable del vehículo

## Condiciones de ejecución

El usuario accede a Internet en su computador personal ingresando su usuario y clave en el aplicativo con un rol que le permita agregar vehículos.

## Entrada

Credenciales (usuario y contraseña).

Especificaciones técnicas del vehículo recolector.

Datos del Chofer.

## Resultado esperado

Con los permisos respectivos el aplicativo mostrará el formulario principal del vehículo, en el cual se validará la información requerida, una vez ingresado aparecerá otro formulario para el ingreso de los datos del chofer.

### 2.5.2. Identificación y geolocalización de contenedores de Residuos sólidos

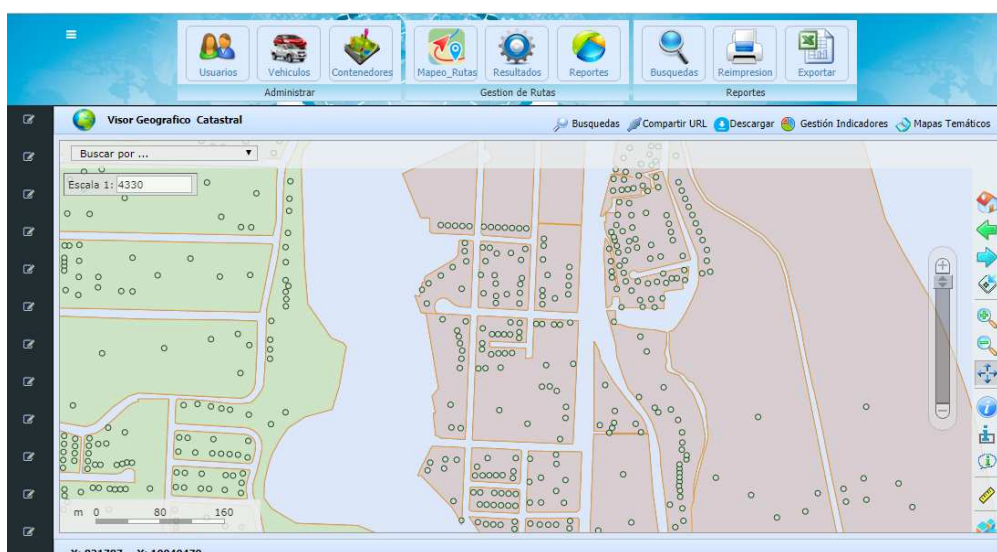


Figura 29: Visor de Georreferenciación de Contenedores

Fuente: Propia

## **Descripción**

El usuario, una vez ingresado en el sistema y en la pantalla de principal, seleccionará la opción “Contenedores”, la cual le llevará a un formulario de parámetros necesarios para ingresar las características de los contenedores y su respectiva localización.

Además, en la pantalla de información del contenedor, el usuario seleccionará la opción “Georreferenciación” y luego agregará las coordenadas importando el archivo en formato de texto en donde consta la descripción, latitud y longitud.

Para agregar coordenadas manualmente, el usuario previamente arrastrará el punto localizado en el mapa que al localizar el contenedor genera las coordenadas respectivas.

## **Condiciones de ejecución**

El usuario debe tener acceso al sistema web, además debe haber ingresado previamente los parámetros del contenedor para el cálculo de las rutas, caso contrario, no podrá visualizar el botón de “Mapeo de Rutas”.

## **Entrada**

Clic en botón de Usuario.

Clic en menú Contenedores.

Clic en Georreferenciación.

## **Resultado esperado**

Se comprobará que el usuario tenga permisos de ingresar datos para ingresar parámetros del contenedor.

El sistema validará que los datos ingresados en los formularios estén correctos, si es así, se mostrará un mensaje de confirmación y en la pantalla principal del contenedor se activarán más opciones dependientes de esta información ingresada.

### 2.5.3. Generación de rutas óptimas

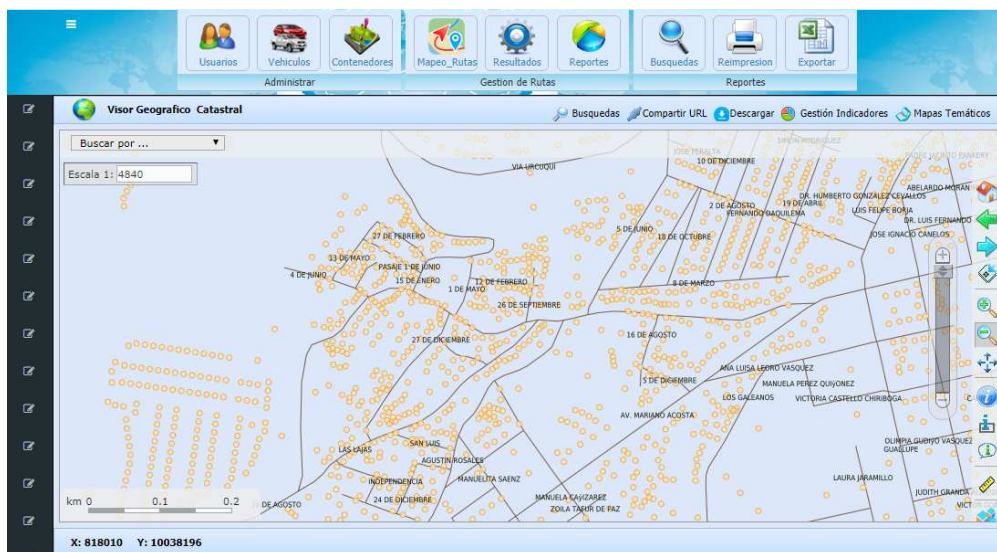


Figura 30: Visor geográfico para el Ruteo

*Fuente: Propia*

#### Descripción

El usuario ingresará parámetros del vehículo recolector, así como también los parámetros del contenedor con la finalidad de establecer valores fijos al momento de realizar las respectivas comparaciones con la carga y descarga de los desechos sólidos.

#### Condiciones de ejecución

El usuario debe tener los permisos para ingresar los datos básicos que apoyan al monitoreo, además el vehículo recolector debe tener calculado sus valores fijos de carga en unidad de medida de kilogramos y los parámetros del contenedor.

#### Entrada

Ingreso de parámetros de cálculo para determinar porcentajes de carga y detalle del estado actual del contenedor y del vehículo recolector.

#### Resultado esperado

Una vez ingresados los datos de valores estándar de carga en kilogramos, el usuario podrá visualizar en la pantalla principal del vehículo las diferentes rutas en función al estado actual de cada contenedor.

## 2.5.4. Generación de reportes y registro de seguimiento

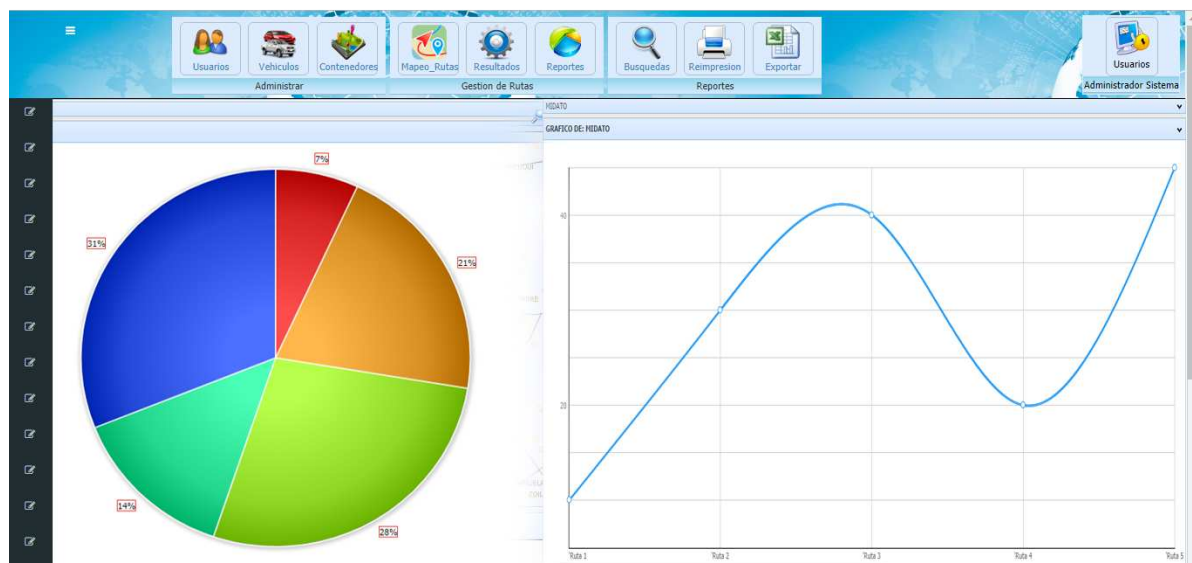


Figura 31: Reportes y estadísticas Gráficos.

*Fuente: Propia*

### Descripción

El usuario podrá visualizar dos reportes gráficos: por cada ruta creada, carga y descarga de desechos sólidos, por vehículo recolector, además podrá registrar la actividad de recolección diaria, mensual y anual.

### Condiciones de ejecución

El usuario debe tener permisos de visualización de reportes y la generación de zonas definidas para la recolección permanente.

### Entrada

Actividad del vehículo recolector, filtro de fechas para el reporte del seguimiento.

### Resultado esperado

En la pantalla principal del vehículo recolector, el usuario podrá ver y exportar los gráficos de rutas creadas, al acceder podrá visualizar las zonas creadas para la gestión de desechos sólidos, podrá visualizar un reporte de actividades en la sección “Seguimiento en línea”.

## CAPÍTULO III

### 3. PRESENTACION DE RESULTADOS

#### 3.1. Geolocalización de Contenedores

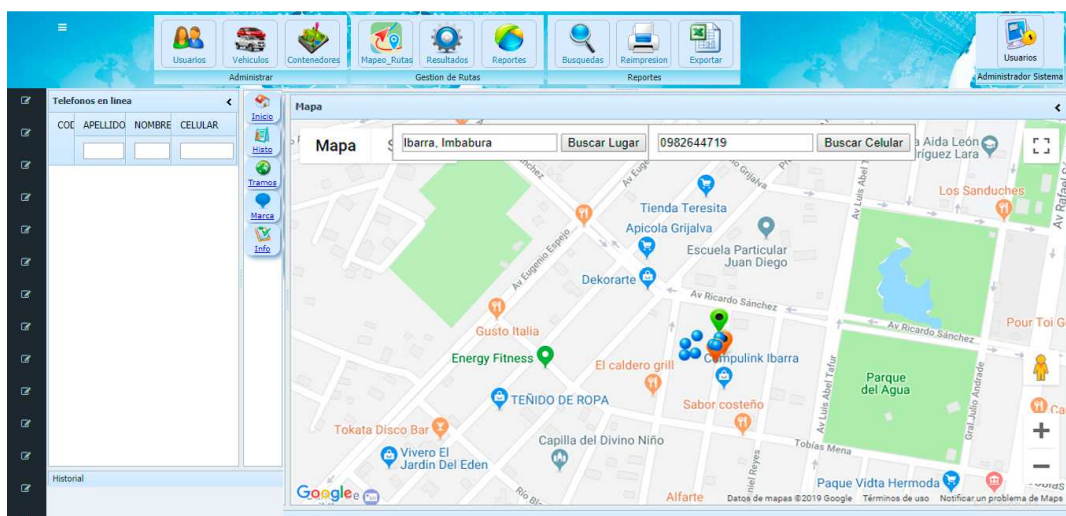


Figura 32: Visor de monitoreo de Contenedores

*Fuente: Propia*

La geolocalización de los contenedores de residuos de la Ciudad de Ibarra permite, brindar información con datos en tiempo real sobre el volumen de llenado de los contenedores, además de otras muchas posibles incidencias.

Disponer de los datos del volumen de llenado de los contenedores a tiempo real, así como la geolocalización del mismo, permite optimizar las rutas de recogida, lanzando la flota de camiones y activando la logística de recogida solamente cuando es estrictamente necesario. De este modo se reducen considerablemente las emisiones de CO<sub>2</sub> emitidas a la atmósfera.

Otra gran ventaja de este proceso es que el stock de residuos que las empresas de recogida necesitan almacenar, previo transporte al reciclador, se ve muy reducido, gracias a que los datos de llenado nos permiten conocer el volumen de residuos en Toneladas, que se van a reciclar.

#### 3.2. Validación de Datos

En la ciudad de Ibarra, se ha localizado los contenedores en varios sectores de la Ciudad georeferenciando y caracterizando cada uno de ellos. El mapa situará de manera instantánea los contenedores más cercanos al usuario para la recolección de desechos.

En el proceso de validación se adaptan las actividades de la recogida de residuos a través de las nuevas tecnologías existentes de monitoreo continuo con equipos de posicionamiento global y sistemas de información geográfica.

La aplicación incluye una actualización que permitirá al vehículo recolector conocer los distintos estados de cada contenedor. De esta manera, se pretende evitar la redundancia en el ciclo de rutas de recolección optimizando de esta manera los recursos destinados a esta labor diaria.

A partir del análisis bibliográfico, se decidió aplicar el modelo CVRP, ya que se cumplieron los requisitos observados en el entorno local. Los datos específicos para determinar el número óptimo de subzonas se muestran en la Tabla:

Tabla 19: *Características de las subzonas de la ciudad de Ibarra.*

<b>Factores</b>	<b>Valores</b>	<b>Unidades</b>
Área de zona Residencial	31,97	km <sup>2</sup>
Área de zona Comercial	3,64	km <sup>2</sup>
<b>Áreas Residencial y comercial</b>	<b>35,62</b>	<b>km<sup>2</sup></b>
PPC (producción de residuos por persona)	0,685	kg/habitantes-día
Capacidad del camión colector (Kc)	10	Ton
Frecuencia de recogida (Fr)	2 - 7	Días/semana
Densidad del área residencial	4282,80	Habitantes/km <sup>2</sup>
Densidad de Zona Comercial	8123,94	Habitantes /km <sup>2</sup>
Número de viajes por distrito (Vss)	2 - 3	Viajes / colección del primer distrito
Días laborables	3 - 7	Días/semana

Fuente: (Propia, 2018)

Respecto al Modelo aplicado se tomaron en cuenta tres escenarios: en el primero, una frecuencia de recolección de dos veces por semana. En el segundo, una frecuencia de recolección cada tercer día. Además, en el tercer escenario, diariamente, el inventario máximo. Es importante mencionar que para este cálculo; solo se consideraron las parroquias residenciales de Ibarra (Alpachaca, Priorato, Sagrario, San Francisco, Caranqui y San Antonio); como se muestra en la Tabla:

Tabla 20: *Características de las subzonas de la ciudad de Ibarra.*

Descripción	Valor	Unidad
Población total	166528	habitantes
Producción diaria per cápita.	0,685	kg / habitante-día
Generación de otras fuentes.	21588,7 8	kg / día
Generación diaria de residuos.	135660, 46	kg / día
Frecuencia de recogida dos veces por semana.	180880, 61	kg / día
Frecuencia de recogida cada tres días.	203490, 69	kg / día
Frecuencia de recolección diaria	271320, 92	kg / día
Capacidad del camión en kg.	10000	kg
Número de viajes por día del camión	2	viajes diarios
Eficiencia de llenado de camiones	0,95	%

Fuente: (Propia, 2018)

### 3.3. Resultados

La información recopilada en campo se la valida en la ciudad de Ibarra para determinar rutas óptimas de recolección de desechos sólidos.

Este trabajo propone un modelo de distribución territorial para la optimización de la distribución de las rutas de los vehículos colectores en los sectores comercial y residencial del Cantón Ibarra. Se basa también en datos de población y generación de residuos.



Además, se propone la división de las áreas residenciales de manera homogénea e indica las mediciones realizadas en las subzonas propuestas lo que demuestra la homogeneización entre las mismas, así como lo que implica un balance en la carga de trabajo para los equipos de recolección. Resultado de la evaluación del modelo se muestra en las siguientes tablas:

Tabla 21: *Características de los distritos propuestos*

<b>Zona</b>	<b>Nodos</b>	<b>Personas para cada zona</b>	<b>Basura estimada por distritos (Tonelada / día)</b>
Distrito residencial1	3172	15296,70	10,48
Distrito residencial2	3160	15238,83	10,44
Distrito residencial3	3167	15272,59	10,46
Distrito residencial4	3166	15267,76	10,46
Distrito residencial5	3178	15325,63	10,50
Distrito residencial6	3162	15248,48	10,45
Distrito residencial7	3176	15315,99	10,49
Distrito residencial8	3161	15243,65	10,44
Distrito residencial9	3123	15060,40	10,32
<b>Total</b>	<b>28465</b>	<b>137.270,03</b>	<b>94,03</b>

Fuente: (Propia, 2018)

Tabla 22: *Especificaciones de zonas residenciales y comerciales.*

<b>Especificaciones</b>	<b>Zona Residencial</b>	<b>Zona Comercial</b>
Habitantes	137.270,03	29.247,94
PPC	0,685	0,685
kg / día	94.029,97	20.034,84
tonelada / día	94,03	20,03
tonelada / camión	10	10
camiones / día	9,40	2,00
Número de distritos	9,38	1,01
Habitantes / distrito	14.633,70	-
kg / distrito-día	10.024,08	-

Fuente: (Propia, 2018)

De esta manera tenemos como resultado geográfico la división de distritos propuestos para la gestión de residuos sólidos:

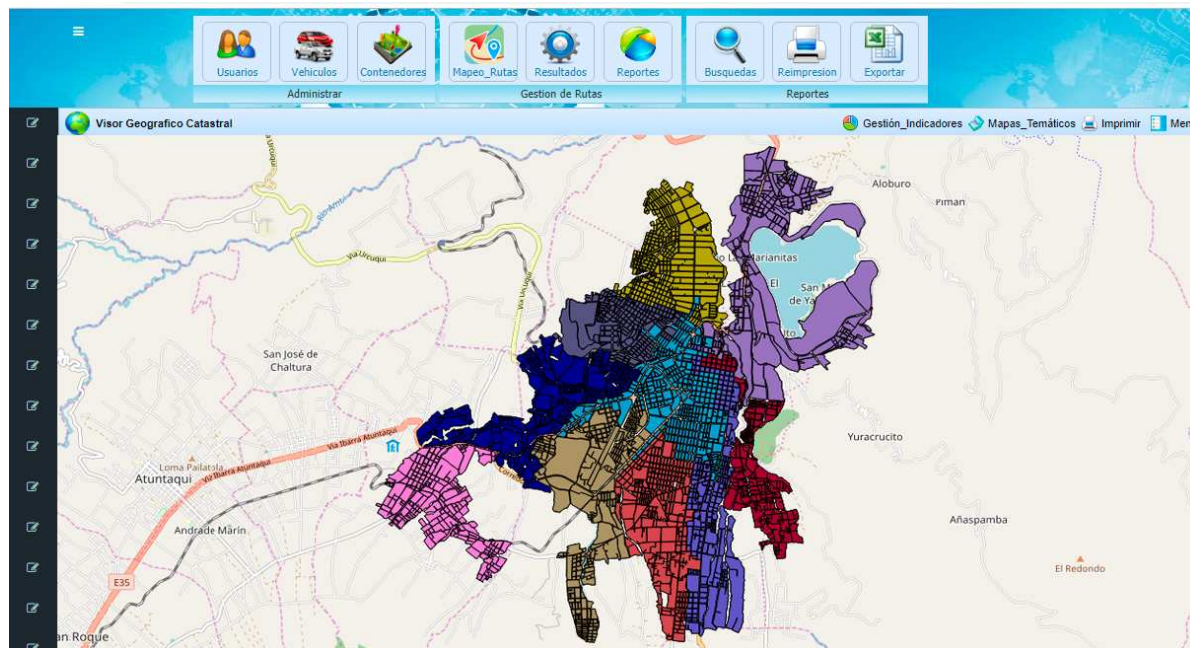


Figura 33: Mapeo de Zonas propuestas para la Gestión de Residuos Sólidos

*Fuente: Elaboración Propia*

### 3.3.1. Impactos generados con el proyecto

Tabla 23: *Impactos generados*

IMPACTO	BENEFICIOS
<b>Económico</b>	Permitir la optimización de los recursos en las instituciones públicas que tienen como competencia la gestión de desechos sólidos
<b>Social</b>	El aporte primordial que se dará con el presente proyecto es la de resolver un problema social y ambiental de la ciudad de Ibarra así como también mejorar la planificación de recolección de desechos de manera óptima y organizada
<b>Tecnológico</b>	En el aspecto tecnológico generar un mecanismo que permita determinar las rutas efectivas de recolección de desechos sólidos, además de las características del estado actual de cada uno de los

---

contenedores ubicados en la ciudad.

<b>Educativo</b>	En el aspecto Educativo brindar un aporte esencial en el uso de herramientas de Sistemas de Información Geográfica y de algoritmos de ruteo que permitan generar información investigativa y a la vez llevar a situaciones reales que suceden en nuestro medio.
<b>Ambiental</b>	Brindar un aporte significativo en la gestión de recolección segura de los desechos sólidos, a través de un aplicativo Web que permita llevar un control adecuado de contenedores y camiones de recolección, con tecnología GPRS y GSM; aportando de esta manera a la reducción de la contaminación ambiental

---

*Fuente: Propia*

## **CAPÍTULO IV**

### **4. CONCLUSIONES Y RECOMENDACIONES**

#### **4.1. Conclusiones**

- En el desarrollo del aplicativo Web para la solución de ruteo de vehículos se concluye que la metodología XP fue la más acertada debido a su flexibilidad en la documentación, además el framework Laravel permitió agilizar la programación debido a sus múltiples funciones y componentes.
- El modelo CVRP permitió conseguir una solución realizable al problema de ruteo de vehículos con capacidad considerando una demanda compartida.
- Con la geolocalización de contenedores se logró identificar de manera clara su estado actual, además de identificar el grado de contaminación que genera el momento de llenarse el contenedor.
- Al validar la información recopilada en campo y procesada en un sistema de información geográfica se obtiene como resultado la optimización de las rutas de recolección generando de esta manera un componente importante en el área social y ambiental generan un aporte significativo en la gestión de recolección segura de los desechos sólidos.
- De la misma forma mediante la ejecución del aplicativo se podría concluir que, si se desea una frecuencia de recolección de dos veces por semana o cada tercer día, debería ser suficiente con el grupo actual de vehículos recolectores de la municipalidad; sin embargo, si la oferta del servicio desea ser alcanzada en una recolección diaria, es necesario la adquisición de un nuevo vehículo tipo camión compactador de 16 yardas.

#### **4.2. Recomendaciones**

- Una vez que se investigó el ruteo de vehículos recolectores en la ciudad de Ibarra, es recomendable investigar la aplicación que este tiene en el sector de tránsito y transporte para generar sistemas inteligentes que permitan optimizar rutas para la planificación de buses de transporte público en la ciudad de Ibarra en donde la capacidad es limitada y sus rutas no son tan eficientes dejando ciertas zonas desatendidas.

- Como proyecto a futuro se investigará más variables para el ruteo de vehículos con capacidad, en particular variables de: tiempo y múltiples depósitos.
- En la implementación de aplicativos para el ruteo someterlos a pruebas numéricas y ver cuál de ellos brinda mejores soluciones, con sus respectivas variables.
- Después de lo que se ha desarrollado, se recomienda llevar a cabo otros estudios relacionados, tales como: zonificación de micro zonas teniendo en cuenta otro tipo de restricciones, como la asignación de frecuencias y programas de operación, el levantamiento de la recolección, el programa y el proceso de movimientos, así como la medición del ambiente e impacto causado.

### 4.3. Bibliografía

- Archetti, C. M. (2001). The Vehicle Routing Problem. *General Distances and Multiple Customer Visits*, 102-112.
- Beck, K. (1999). Extreme Programming Explained. En *Extreme Programming Explained: Embrace Change*. Medford: Universidad de Oregon.
- Beck, K. (2005). Aceptando el cambio. En *2.ª Edición Extreme Programming Explained: Embrace Change*. Medford.
- Bustamante Dayana, R. J. (Marzo del 2014). Metodología XP. *Metodología Actual*.
- BustamanteDayana C.C.I: 22.983.709 ,Rodríguez Jean C C.I: 21.169.047. (2014). *Metodología Actual, Metodología XP*. Barinas.
- CONSTITUYENTE, A. (2008). CONSTITUCION DE LA REPUBLICA DEL ECUADOR. MONTECRISTE.
- Contardo Vera, C. (2005). *Formulación y solución de un problema de ruteo de vehículos con demanda variable en tiempo real, trasbordos y ventanas de tiempo*. Chile: Universidad de Chile.
- David de la Fuente, J. L. (2011). Estado del arte de algoritmos basados en colonias de hormigas. *Congreso de Ingeniería de Organización* .
- Dorado Cerón, J., & Muñoz, E. (2012). JSF. Obtenido de <http://osl2.uca.es/wikiCE/index.php?title=JSF&oldid=1277>
- Ecuador. (2017). *Plan Nacional de Desarrollo*.
- Ecuador, R. d. (2017). Plan Nacional de Desarrollo 2017-2021.
- Gallart Suárez, J. (2009). *Análisis diseño e implementación de un Algoritmo Meta Heurístico*. Peru: Universidad Católica del Perú.
- Guimarans Serrano, D. (2012). *Hybrid algorithms for solving routing problems*. Barcelona: Tesis PHD.

- ISAIAS, C. G. (2017). *Optimización de las rutas de recolección y tratamiento anaerobio para la industrialización energética de los desechos sólidos municipales*. Ibarra.
- Jacobson, I. (1992). Object-Oriented Software Engineering. En *A Use Case Driven Approach*.
- jbarrios. (30 de 05 de 2003 ). *Enterprise JavaBeans - DCC*. Obtenido de Enterprise JavaBeans - DCC: <https://users.dcc.uchile.cl/~jbarrios/J2EE/node44.html>
- M. A. H. Akhand. (2017). *Constructive and Clustering Methods to Solve Capacitated Vehicle Routing Problem*. Obtenido de <http://www.computerscijournal.org/vol10no3/constructive-and-clustering-methods-to-solve-capacitated-vehicle-routing-problem/>
- MAE. (2002). *Programa Nacional para la Gestión Integral de Desechos Sólidos – PNGIDS ECUADOR*. QUITO. Obtenido de <http://www.ambiente.gob.ec/programa-pngids-ecuador/>
- Medical, H. H. (2018). *PostGIS*. Obtenido de PostGIS: <https://postgis.net/>
- Olivera, A. (2005). *Tesis de Maestría en Informática PEDECIBA*. Uruguay.
- PDOT. (2012). *Plan de Desarrollo y Ordenamiento Territorial*.
- (2017). *PNBV*.
- PNBV. (2017). *Plan Nacional de Desarrollo para el Buen Vivir*.
- PNGIDS. (2010). *Programa Nacional para la Gestión Integral de Desechos Sólidos*.
- PostgreSQL. (2018). *PostgreSQL*. Obtenido de PostgreSQL: <https://www.postgresql.org/>
- Propia, W. G.-P.-C.-G.-J.-L.-S.-C. (2018). Macro-Routing. *Macro-Routing Proposal to Optimize the Municipal Solid Waste* 26-8.
- Ramos, J. (2011). *Combining probabilistic algorithms, Constraint Programming and Lagrangian Relaxation to solve the Vehicle Routing Problem*. Barcelona.
- Ramser, G. B. (1964). Problema de Enrutamiento de Vehículos. *The Truck Dispatching Problem*, 15–64.
- RoseIndia. (2015). *Java Persistence API*. Obtenido de <http://www.roseindia.net/ejb/JavaPersistenceAPI.shtml>
- Savelsberg, M. (1995). Local search in RoutingProblem with Time Windows. *Annals of Operations Research*, 1-7.
- Suresh Nanda Kumar, R. P. (2015). Estudio comparativo de la solución basada en algoritmo genético. *Journal of Service Science and Management*.
- Toth, P., & Vigo, D. (2002). *The vehicle routing problem*. Philadelphia : Society for Industrial and Applied Mathematics, ©2002.
- U.MÁLAGA. (2007). *Metaheurísticas e Ingeniería del Software*. Malaga.
- Villalobos, P. D. (12 de 07 de 2017). Obtenido de problema de rutas con vehículos capacitados - CVRP: [http://arodrigu.webs.upv.es/grafos/doku.php?id=problema\\_cvrp](http://arodrigu.webs.upv.es/grafos/doku.php?id=problema_cvrp)
- Web, P. (2002). Obtenido de <http://www.ambiente.gob.ec/programa-pngids-ecuador/>