



PONTIFICIA **UNIVERSIDAD CATÓLICA** DEL PERÚ

Esta obra ha sido publicada bajo la licencia Creative Commons
Reconocimiento-No comercial-Compartir bajo la misma licencia 2.5 Perú.

Para ver una copia de dicha licencia, visite
<http://creativecommons.org/licenses/by-nc-sa/2.5/pe/>



PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA



PONTIFICIA
**UNIVERSIDAD
CATÓLICA**
DEL PERÚ

**DESARROLLO DE UN SISTEMA DE GESTIÓN DE SERVICIO DE
TAXIS QUE OPTIMIZA RUTAS DE TRANSPORTE**

Tesis para optar por el Título de Ingeniero Informático, que presenta el bachiller:

Miguel Ángel Ortega Bulnes

ASESOR: Dra. Nora Bertha La Serna Palomino

Lima, Noviembre del 2010

Resumen

El presente proyecto de fin de carrera se dedica a desarrollar un sistema de información de gestión de servicio de taxis que optimice las rutas de transporte, el cual comprenderá los módulos de seguridad, mantenimiento, recepción y asignación de servicios, y apoyo a la gestión.

Básicamente, el sistema permitirá registrar las solicitudes de servicios de taxi y atenderlos eficientemente asignando la mejor ruta.

De esta manera las empresas de taxi podrán ahorrar tiempo y dinero a la hora de ofrecer un servicio desde un lugar a cualquier otro punto de la ciudad, mejorando la atención al cliente mediante una respuesta rápida.

Para llevar a cabo el proyecto se ha hecho uso de un algoritmo de optimización apropiado al contexto del problema, además de haberse utilizado software libre para la elaboración del sistema. Por otro lado, se ha aplicado una metodología de desarrollo que toma como base el Proceso Unificado Racional (Rational Unified Process – RUP), así como también se ha empleado la guía estándar para la gestión de proyectos (Project Management Body of Knowledge – PMBOK).

Finalmente, resulta conveniente mencionar que durante el proceso de desarrollo de la tesis, se aplicaron en gran parte los conocimientos adquiridos durante los años de estudio de la carrera, por lo que el proyecto informático desarrollado es el resultado de competencias y habilidades del aspirante al título de ingeniero.



Dedicatoria

A Dios por darme salud
y a mi familia por darme fuerzas para seguir adelante.



Agradecimientos

A mi asesor por su disponibilidad y consejos,
y a las personas que hicieron posible este trabajo.

Tabla de Contenido

Introducción	7
Capítulo I: Generalidades	9
1.1. Definición del Problema	9
1.2. Marco Conceptual	10
1.2.1. Sistema de Información (SI)	10
1.2.2. Tipos de Sistema de Información	11
1.2.3. Optimización Combinatoria	14
1.2.4. Algoritmos Exactos y Aproximados	15
1.2.5. Algoritmos Meta heurísticos	15
1.2.6. Algoritmos Bio-inspirados	17
1.2.7. Optimización Colonia de Hormigas	19
1.3. Estado del arte	25
1.3.1. Problema del camino más corto (Shortest Path Problem)	25
1.3.2. Objeto de estudio: Problema del camino más corto s-t	25
1.3.3. Algoritmos que resuelven el problema	26
1.3.4. Soluciones existentes y estudios relacionados	28
1.4. Plan del proyecto	30
1.4.1. Metodología de gestión	30
1.4.2. Etapas del proyecto	36
1.5. Descripción y sustentación de la solución	38
1.5.1. Componentes	38
1.5.2. Ventajas	39
Capítulo II: Análisis	40
2.1. Metodología de desarrollo del proyecto	40
2.2. Identificación de requerimientos	44
2.2.1. Diagramas de caso de uso	48
I. Catálogo de actores	48
II. Casos de uso	49
2.3. Análisis de la solución	56
2.3.1. Viabilidad	56
2.3.2. Análisis Técnico y Económico	56
2.3.3. Asignación de Funciones	57
2.3.4. Restricciones de Costo y Tiempo	58
2.3.5. Diagrama de clases de análisis	59
2.3.6. Diagrama de paquetes	59
Capítulo III: Diseño	60
3.1. Arquitectura de la solución	60
3.1.1. Tecnología Requerida	62
3.1.2. Diagrama de Despliegue	62
3.1.3. Diagrama de Componentes	63
3.1.4. Modelo de base de datos	64
3.2. Diseño de Interfaz Gráfica	67
3.2.1. Principios Básicos	67
3.2.2. Sobre el diseño en general	67
3.2.3. Diseño de pantallas	68
Capítulo IV: Construcción	72
4.1. Construcción	72
4.1.1. Tecnología usada	72
4.1.2. Algoritmo de optimización propuesto	75
4.2. Pruebas	84
4.2.1. Pruebas de aceptación	84
4.2.2. Casos de pruebas	84
4.2.3. Catálogo de pruebas	85
Capítulo V: Observaciones, Conclusiones y Recomendaciones	87
5.1. Observaciones	87
5.2. Conclusiones	88
5.3. Recomendaciones	88
Bibliografía	89
Anexos	92

Índice de Figuras

Figura 1.1 Tipos de Sistemas de Información.	11
Figura 1.2 Plan de Proyecto: Administración de la Integración del Proyecto.	32
Figura 1.3 Plan de Proyecto: Administración del Alcance del Proyecto.	33
Figura 1.4 Plan de Proyecto: Administración del Tiempo del Proyecto.	34
Figura 1.5 Plan de Proyecto: Administración de la Calidad del Proyecto.	35
Figura 1.6 Estructura de descomposición del trabajo: EDT.	37
Figura 2.1 Metodología base para el proyecto: Estructura del RUP.	41
Figura 2.2 Catálogo de actores.	48
Figura 2.3 Diagrama de casos de uso del módulo de Seguridad.	50
Figura 2.4 Diagrama de casos de uso del módulo de Mantenimiento.	51
Figura 2.5 Diagrama de casos de uso del módulo de Mantenimiento.	51
Figura 2.6 Diagrama de casos de uso del módulo de Mantenimiento.	52
Figura 2.7 Diagrama de casos de uso del módulo de Recepción y Asignación.	53
Figura 2.8 Diagrama de casos de uso del módulo de Recepción y Asignación.	53
Figura 2.9 Diagrama de casos de uso del módulo de Recepción y Asignación.	54
Figura 2.10 Diagrama de casos de uso del módulo de Apoyo a la Gestión.	55
Figura 2.11 Diagrama de Paquetes.	59
Figura 3.1 Arquitectura de la solución.	61
Figura 3.2 Diagrama de Despliegue.	63
Figura 3.3 Diagrama de Componentes.	64
Figura 3.4 Diagrama Entidad Relación: Esquema Negocio.	65
Figura 3.5 Diagrama Entidad Relación: Esquema Grafo.	66
Figura 3.6 Ventana Autenticación de Usuario.	69
Figura 3.7 Ventana Principal del sistema.	69
Figura 3.8 Ventana de Mantenimiento Estándar.	70
Figura 3.9 Ventana de Registro Estándar.	70
Figura 3.10 Ventana de Reporte Informativo de Solicitudes.	71
Figura 3.11 Ventana de Reporte Estadístico de Solicitudes.	71
Figura 4.1 Acotación del espacio de búsqueda.	76

Índice de Tablas

Tabla 1.1 Tabla de aplicaciones de algoritmos meta heurísticos.	17
Tabla 1.2 Tabla comparativa de aplicaciones y trabajos informáticos existentes.	29
Tabla 2.1 Módulo de Seguridad.	44
Tabla 2.2 Módulo de Mantenimiento.	45
Tabla 2.3 Módulo de Recepción y Asignación de servicios.	46
Tabla 2.4 Módulo de Apoyo a la Gestión de servicios.	47
Tabla 2.5 Requerimientos No Funcionales.	47
Tabla 2.6 Presupuesto para el desarrollo.	57
Tabla 3.1 Arquitectura de la solución.	62
Tabla 3.2 Descripción de los nodos.	63
Tabla 4.1 Clases equivalentes – Actualizar Contraseña.	85
Tabla 4.2 Casos de Pruebas – Actualizar Contraseña.	86

Introducción

El presente proyecto de fin de carrera o tesis de pregrado se centra en desarrollar un sistema de información que apoye la gestión de servicio de taxis, dicho sistema busca mejorar el servicio de taxi que se les ofrece a los clientes empleando determinadas tecnologías informáticas.

Para llevar a cabo el desarrollo del sistema mencionado se empezó por identificar la problemática que le da origen, siendo ésta el tiempo que se pierde en la trayectoria del servicio de taxi y que por consiguiente deja a un cliente insatisfecho con el servicio proporcionado. A partir de este planteamiento se propuso el desarrollo de un sistema de información que dé solución a la necesidad de brindar un servicio de mejor calidad para los clientes. Es por ello que se ha realizado el análisis, diseño e implementación del sistema de gestión de servicio de taxis, que soporte los siguientes procesos: el registro de solicitudes, seguimiento y control de las unidades, asignación eficiente del servicio, obtención de la ruta óptima del servicio y generación de reportes con datos confiables y actualizados.

A la vez, es importante mencionar que para el desarrollo del presente proyecto se ha contado con una empresa usuaria que proporcione la información necesaria de la lógica del negocio, con el propósito de realizar un trabajo objetivo.

Asimismo, cabe indicar que en este trabajo no sólo se implementó un algoritmo de optimización para la generación de rutas de transporte, precisamente la meta heurística Optimización Colonia de Hormigas (Ant Colony Optimization – ACO), sino que también se realizaron aportes matemáticos enfocados al problema en específico.

Finalmente, este proyecto de tesis está dividido en los 5 capítulos siguientes: Generalidades, Análisis, Diseño, Construcción y, por último, Observaciones, Conclusiones y Recomendaciones.

En el capítulo 1, Generalidades, se presentan las secciones en donde se plantea el problema a resolver, los fundamentos teóricos, el plan del proyecto, los trabajos relacionados y un sustento de la solución.

En el capítulo 2, Análisis, se describen las secciones en las cuales se evalúa la viabilidad del sistema, se realiza un análisis técnico y económico, se asigna funciones, se establecen restricciones y una definición detallada del sistema.

En el capítulo 3, Diseño, se detallan las secciones en las que se definen la arquitectura y el diseño de la interfaz gráfica de la solución.

En el capítulo 4, Construcción, se exponen las secciones en donde se justifican las tecnologías utilizadas y se presentan las pruebas de aceptación.

En el capítulo 5, Observaciones, Conclusiones y Recomendaciones, se muestran las secciones en donde se dan las observaciones, comentarios, conclusiones finales al proyecto y recomendaciones futuras.



Capítulo I: Generalidades

En el presente capítulo se presentan la definición del problema con la explicación de lo que se desea resolver en el proyecto, luego el marco conceptual con la teoría fundamental o básica para entender el problema, seguidamente el estado del arte con la explicación de cómo se resuelve actualmente el problema; también se incluye el plan del proyecto llevado a cabo con las metodologías y etapas, y finalmente una explicación detallada de la solución a desarrollarse.

1.1. Definición del Problema

En la actualidad, para las empresas de servicio de taxis, es cada vez mayor la necesidad de afrontar el costo que origina no estar a la altura de las expectativas del cliente, en cuanto a ofrecer un mejor servicio. Además, estas empresas se encuentran en un mercado cada vez más competitivo, debido a una creciente demanda de brindar calidad en los servicios, y afrontan los problemas de tránsito que se presentan, generalmente, en las ciudades más representativas.

En tal sentido existen problemas para la adecuada gestión de servicios de taxis al momento de querer ofrecer al cliente una mejor atención, no sólo en términos del trato que se le da, sino más bien en atender de manera rápida la solicitud del servicio, asignar de manera eficiente un taxi y demorar menos tiempo en lo que implica la trayectoria del servicio.

En países subdesarrollados, si bien no se aprecia un claro interés por brindar un mejor servicio de transporte al usuario en el sector público, sí es tomado con seriedad por las empresas privadas al querer brindar un servicio de calidad y buscar cubrir esta necesidad con herramientas adecuadas.

El Perú, un país con considerable demanda turística, presenta una serie de dificultades que no permite una solución sencilla, particularmente en las capitales de las provincias o en lugares con gran concentración de habitantes; se tiene, por ejemplo, una inadecuada infraestructura vial con frecuentes cambios (construcción y reparación de vías), frecuente congestión (especialmente en horas punta) y carencia de control en determinadas zonas.

Por lo expuesto anteriormente, las empresas de servicios de taxis afrontan un reto al tratar este problema y es evidente la necesidad de recurrir a las tecnologías de información con el fin de aumentar la calidad del servicio al cliente. Esto supone una mejora indudable en este tipo de servicio en el Perú.

1.2. Marco Conceptual

1.2.1. Sistema de Información (SI)

Un sistema de información es un conjunto organizado de elementos que interactúan entre sí para procesar los datos y la información (incluyendo procesos manuales y automáticos), y distribuirla de la manera más adecuada posible en una determinada organización en función de sus objetivos [1]. Normalmente el término es usado de manera errónea como sinónimo de sistema de información informático, estos son el campo de estudio de la tecnología de la información (IT), y aunque puedan formar parte de un sistema de información (como recurso material), por sí solos no se pueden considerar como sistemas de información, este concepto es más amplio que el de sistema de información informático.

Entre los elementos que la conforman tenemos:

- a) Personas.
- b) Datos.
- c) Actividades o técnicas de trabajo y
- d) Recursos Materiales (típicamente recursos informáticos y de comunicación).

1.2.2. Tipos de Sistema de Información

La mayor parte de los sistemas están diseñados para generar información y respaldar la toma de decisiones a diferentes niveles de gerencia, así como para realizar el mantenimiento de registros y trabajos rutinarios de procesamiento de transacciones.

La clasificación de los sistemas de información puede verse en la figura 1.1.

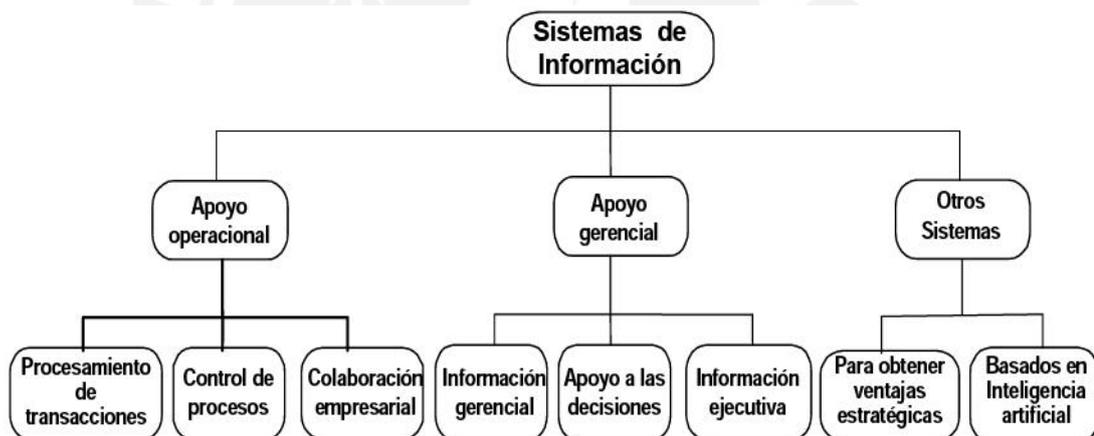


Figura 1.1 Tipos de Sistemas de Información [14].

Sistemas de apoyo a las operaciones

El papel de los sistemas de respaldo a las operaciones consiste en procesar eficientemente las transacciones, controlar los procesos, respaldar las comunicaciones y la colaboración en la empresa y actualizar las bases de datos de la empresa, pero no hacen énfasis en la generación de productos específicos que puedan ser utilizados de manera óptima por los gerentes, ya que éstos requieren de información procesada [14].

Dentro de los sistemas de apoyo a las transacciones se encuentran:

- **Sistemas de procesamiento de transacciones**

Estos sistemas registran y procesan datos resultantes de las transacciones comerciales, tales como ventas, compras, facturación y cobranzas, gestión de compras, pagos a proveedores, etc.

Los resultados de este proceso se usan para actualizar las bases de datos de clientes, de inventarios, personal y de otras bases organizacionales. Posteriormente, estas bases de datos proporcionan los recursos de datos que pueden ser procesados y utilizados por los sistemas de apoyo gerencial.

- **Sistemas de control de procesos**

Son los que se utilizan para tomar decisiones que controlan procesos, en los cuales las computadoras toman decisiones automáticamente sobre un proceso. Por ejemplo, una planta de potabilización puede utilizar sensores electrónicos enlazados a computadoras con el fin de supervisar el proceso de tratamiento de agua.

- **Sistemas de colaboración empresarial**

Son sistemas que utilizan una serie de tecnologías de la información para ayudar que las personas trabajen en forma conjunta. Este tipo de sistemas ayuda a colaborar, comunicar ideas, compartir recursos y coordinar esfuerzos de trabajo de los miembros de los muchos equipos de procesos, proyectos y de otros grupos de trabajo de las organizaciones. La meta de estos sistemas es aumentar la productividad y la creatividad de los equipos de trabajo.

Sistemas de apoyo gerencial

Son los sistemas de información que se centran en el suministro de información y respaldo para la toma de decisiones efectivas por parte de los gerentes. Las dos características principales de los sistemas de apoyo gerencial son: La orientación gerencial de la tecnología de la información en la empresa y la visión de las aplicaciones como sistemas interrelacionados e integrados [14].

Dentro de los sistemas de apoyo gerencial se encuentran:

- **Sistemas de información gerencial**

Son la forma más común de sistema de apoyo gerencial, suministran a los usuarios finales productos de información que respaldan gran parte de los procesos cotidianos de toma de decisiones, mediante la provisión de informes y presentaciones a la gerencia. Estos sistemas de información gerencial recuperan de las bases de datos información actualizadas sobre operaciones de las empresas y también obtienen a partir de fuentes externas datos del entorno empresarial.

- **Sistemas de apoyo a las decisiones**

Son sistemas de información interactivos que se basan en el computador y que utilizan modelos de decisión y bases de datos especializadas para apoyar los procesos de toma de decisiones de los usuarios finales. En esta clase de sistemas los gerentes simulan y exploran nuevas alternativas y reciben información tentativa basada en un conjunto de supuestos, y por lo tanto no es necesario especificar de antemano sus necesidades de información, sino que los sistemas de apoyo a las decisiones ayudan a encontrar de manera interactiva la información que se necesita.

- **Sistemas de información ejecutiva**

Son sistemas de información gerencial adaptados a las necesidades estratégicas de información de la alta gerencia. El objetivo consiste en proporcionar a la alta gerencia un acceso inmediato y fácil a la información selectiva sobre factores claves que son fundamentales para el logro de los objetivos estratégicos de la organización.

Otros sistemas

Existen adicionalmente otros tipos de sistemas de información que son de mayor aplicación en otro tipo de industrias, pero que son mencionados para mostrar la tendencia de la tecnología de la información en esos campos [14].

Dentro de estos sistemas se encuentran:

- **Sistemas basados en Inteligencia Artificial**

Son sistemas de información que se basan en el conocimiento, es decir que mediante la aplicación de la inteligencia artificial utilizan su conocimiento sobre un área específica para actuar como un consultor experto para los usuarios.

- **Sistemas para obtener ventajas estratégicas**

Estos sistemas comprenden el uso de la tecnología de la información para desarrollar productos, servicios y capacidades que dan a una empresa ventajas estratégicas sobre las fuerzas competitivas que enfrenta en el mercado global.

1.2.3. Optimización Combinatoria

La Optimización Combinatoria es una rama de la optimización en las matemáticas aplicadas y ciencias de la computación [17], relacionada con la investigación operativa, teoría de algoritmos y teoría de la complejidad computacional.

Un problema de optimización combinatoria se puede expresar como un sistema de ecuaciones y expresiones matemáticas relacionadas que describen la esencia del problema.

Este tipo de problema presenta las siguientes características:

- A veces es difícil encontrar una solución posible.
- Paralelamente existe un número muy elevado de soluciones posibles.
- De todas las soluciones alguna es la óptima.

En este mismo contexto aparecen los algoritmos de optimización combinatoria, quienes resuelven instancias de problemas del tipo NP difíciles, reduciendo el tamaño efectivo del espacio y explorando el espacio de búsqueda eficientemente.

El modelo general que representa un problema de optimización combinatoria es de la siguiente manera [16]:

Optimizar $f(x)$.

Sujeto a:

$$g_i(x) > 0 \text{ para } i: 1..m$$

$$h_j(x) > 0 \text{ para } j: 1..n$$

Donde:

- $f(x)$: es conocida como la función objetivo y representa un valor que debe ser optimizado en su máxima o mínima expresión.
- $g(x)$ y $h(x)$: son denominadas las restricciones del problema y especifican las condiciones que debe poseer toda solución viable.

1.2.4. Algoritmos Exactos y Aproximados

Los algoritmos existentes que permiten resolver problemas de optimización combinatoria pueden ser clasificados en algoritmos exactos y algoritmos aproximados [18].

Algoritmos Exactos

Intentan encontrar una solución óptima y asegurar que la solución obtenida sea la única óptima, estos algoritmos incluyen técnicas tales como retroceso o vuelta atrás (backtracking), ramificación y acotación (branch and bound), programación dinámica (dynamic programming), etc.

Algoritmos Aproximados

Debido a que los algoritmos exactos muestran pobre rendimiento para muchos problemas, se desarrollaron los algoritmos aproximados que proveen alta calidad de soluciones a problemas de optimización combinatoria complejos en un tiempo computacional corto.

Los algoritmos aproximados pueden ser clasificados en dos tipos principales:

- Algoritmos de construcción: basado en ir generando soluciones e ir agregando componentes paso a paso.
Ejemplo: heurísticas de construcción ávidas (greedy construction heuristics).
- Algoritmos de búsqueda local: intentan repetidamente mejorar las soluciones retornadas por una búsqueda local.
Ejemplo: algoritmos de mejora iterativa (iterative improvement algorithms).

1.2.5. Algoritmos Meta heurísticos

Son una clase particular de algoritmos aproximados, proveen una base general para permitir crear nuevos procedimientos derivados de heurísticas clásicas e incluyen diferentes conceptos tales como genética, biología, inteligencia artificial, matemáticas, físicas, sistemas neuronales, mecanismos estadísticos, etc. [18].

Tienen como objetivo resolver problemas de optimización combinatoria donde técnicas heurísticas han fallado.

A continuación presentaremos los algoritmos meta heurísticos más relevantes [19]:

A. Búsqueda Tabú (Tabu Search - TS)

Es un procedimiento de búsqueda meta heurístico iterativo para resolver problemas de optimización combinatoria basándose en principios generales de resolución de problemas inteligentes.

B. Algoritmos Genéticos (Genetic Algorithms - GA)

Son una clase de métodos de búsqueda adaptativos en un modelo abstracto de evolución natural, cuya idea básica es mantener una población de soluciones candidatas que evolucionen bajo una presión selectiva y favorezcan a las mejores soluciones.

C. Procedimientos de Búsqueda Ávidos Aleatorios Adaptativos (Greedy Randomized Adaptive Search Procedures - GRASP)

Es un algoritmo meta heurístico que se basa en la ejecución repetida de un procedimiento de construcción de soluciones, en combinación con una búsqueda local que intenta mejorar la calidad de las mismas [8].

D. Optimización Colonia de Hormigas (Ant Colony Optimization - ACO)

Es un algoritmo inspirado del comportamiento de colonias de hormigas reales para resolver problemas de optimización combinatoria. Son esencialmente algoritmos de construcción.

E. Templado Simulado (Simulated Annealing - SA)

Es un algoritmo de búsqueda local basado en el proceso de recocido de sólidos, capaz de evitar hallar el óptimo local.

F. Redes Neuronales Artificiales (Artificial Neural Networks - ANN)

Son modelos extremadamente simplificados de su contraparte biológica, capaces de desarrollar internamente capacidades de procesamiento de la información para resolver un problema en específico.

Entre los problemas reales que resuelven los algoritmos anteriormente descritos tenemos:

Algoritmos Meta Heurísticos	Aplicaciones
Búsqueda Tabú	Finanzas, Manufactura, Planeación y programación, Telecomunicaciones, Transporte, ruteo y diseño de redes, Optimización en estructuras, Manufactura, entre otros. [20]
Algoritmos Genéticos	Acústica, Ingeniería aeroespacial, Astronomía y astrofísica, Química, Ingeniería eléctrica, Finanzas, Juegos, Robótica, Diseño de rutas y horarios, entre otros. [39]
GRASP	Manufactura, Transporte, Telecomunicaciones, Sistemas eléctricos, Biología, entre otros. [21]
Colonia de Hormigas	Ruteo de vehículos, Aprendizaje, Bio-Informática, Problemas de asignación y programación de tareas, Aprendizaje automático, entre otros. [22]
Templado Simulado	Telecomunicaciones, Cibernética, Diseño de horarios, entre otros. [18]
Redes Neuronales Artificiales	Biología, Minería de datos, Ingeniería civil, Análisis financiero, Música, entre otros. [23]

Tabla 1.1 Tabla de aplicaciones de algoritmos meta heurísticos.

1.2.6. Algoritmos Bio-inspirados

Son algoritmos diseñados a partir de la inspiración de procesos observados en la naturaleza, tales algoritmos pertenecen al campo de la computación bio-inspirada y están relacionados con temas de comportamiento social y emergente [24].

Su objetivo es resolver problemas complejos de ingeniería, en donde la estructura del problema no es a menudo conocida, y problemas de optimización combinatoria.

Entre las principales características de estos algoritmos tenemos:

- Son capaces de resolver problemas complejos sin conocer mucho el espacio de búsqueda.
- Hacen uso de muchas decisiones aleatorias.
- Consideran problemas donde la función a ser optimizada es propuesta analíticamente.

A continuación se presenta los algoritmos más conocidos:

A. Algoritmos Evolutivos (Evolutionary Algorithms)

Intentan reproducir los principios de la naturaleza con la más alta exactitud posible, para ello se basa en procesos de evolución, estrategias reproductivas (mutación y recombinación) y un mecanismo de selección natural.

Entre los principales problemas que resuelven se encuentran:

- Problema del agente viajero (Travelling Salesman Problem - TSP).
- Problema de programación de horarios (Timetabling problem).
- Problema de ruteo de vehículos (Vehicle Routing Problem - VRP).
- Entre otros.

B. Inteligencia de Enjambre (Swarm Intelligence)

Son algoritmos que pretenden resolver problemas de distribución inspirados de ejemplos biológicos provistos por insectos sociales como hormigas, termitas, abejas, etc. En otras palabras, está basado en principios de sistemas naturales que consisten de muchos agentes.

Entre los principales problemas que resuelven se encuentran:

- Problema del agente viajero (Travelling Salesman Problem - TSP).
- Problema de asignación de tareas (Task Assignment Problem).
- Ruteo en redes de telecomunicaciones.
- Entre otros.

C. Modelo de Redes Neuronales (Neural Networks Model)

Véase en la sección 1.2.5.

D. Optimización Colonia de hormigas (Ant Colony Optimization - ACO)

Véase en la sección 1.2.7.

1.2.7. Optimización Colonia de Hormigas

Básicamente, la meta heurística Optimización Colonia de Hormigas (OCH) consiste de un conjunto de hormigas artificiales quienes cooperan entre sí para encontrar buenas soluciones a problemas de optimización difíciles.

Para explicar este algoritmo empezaremos por sus inicios hasta describir dos de sus principales variantes.

El comportamiento entre las hormigas de una colonia ha sido parte de numerosos estudios con el propósito de determinar cómo trabajan estos individuos para lograr una meta en común, que es la búsqueda de una fuente de alimento. Un aspecto importante es la habilidad que estos insectos presentan para buscar caminos más cortos del nido hacia la fuente de alimento, a pesar de no tener una visión que les permita tomar un determinado camino.

La clave para conocer el por qué sucede este comportamiento es que las hormigas reaccionan ante una sustancia química depositada durante su recorrido llamada feromona, que resulta ser un tipo de comunicación indirecta entre la colonia y para las hormigas esta presencia influye en la elección de su camino.

Para comprobar que esto sucede así se realizaron experimentos [2], que dieron cierta información sobre la conducta de estos individuos, los cuales comentaremos a continuación:

En un primer experimento se tenía dos puentes con la misma longitud y las hormigas iban del nido a la fuente y viceversa por las dos vías, dejando depósitos de feromona durante su camino. En un principio cada hormiga escogía ir por un puente de manera aleatoria, pero a medida que pasaba el tiempo y se depositaba feromona, estas preferían seguir el rastro con mayor presencia de feromona de manera probabilística.

Por otro lado, un segundo experimento demostró que al tener dos caminos con longitudes diferentes las hormigas se quedaban con el camino más corto pasado un tiempo, lo cual sucedía por una preferencia en regresar al nido también por el camino más corto y por el hecho de que los caminos más cortos son completados más temprano que los largos.

Por tanto se pueden manejar 3 principios que dan a OCH una ventaja frente a otros algoritmos [2]:

- **Estigmergía (Stigmergy):** realizado durante la comunicación a través de la feromona.
- **Comportamiento auto-catalítico (Autocatalytic):** realizado mediante una retroalimentación de lo recorrido.
- **Evaluación implícita (Implicit evaluation):** realizado cuando la feromona es depositada con mayor frecuencia en caminos cortos.

El algoritmo OCH adquiere este comportamiento de las hormigas reales para poder resolver problemas de optimización combinatoria y se basa en hormigas artificiales como agentes computacionales que trabajan en cooperación a través de rastros de feromona.

En OCH, las hormigas artificiales construyen una solución a un problema de optimización a través de un grafo, cuya notación es $G(V, E)$, constituido por un conjunto de vértices V y arcos E . Las hormigas se movilizan en este grafo, construyendo soluciones parciales, dejando adicionalmente depósitos de feromona dependiendo de la calidad de las soluciones encontradas.

A continuación se presenta la estructura general de un algoritmo OCH [4], para más detalle de sus procedimientos véase la sección 4.1.2:

Procedimiento Meta heurística OCH

Inicialización de parámetros.

Mientras Condiciones no satisfechas **hacer**

Actividades Programadas

Construcción de soluciones

Búsqueda local (Opcional)

Actualización de feromonas

Fin Actividades Programadas

Fin Mientras

Fin Procedimiento

La meta heurística OCH mostrada está compuesta básicamente por un paso de inicialización y tres fases dentro una simple iteración, las cuales se explica a continuación:

- **Inicialización de parámetros**

Comprende el valor numérico constante de feromona que se deposita en cada arco, el número de hormigas, los pesos entre la información heurística y probabilística, entre otros.

- **Construcción de soluciones**

Procedimiento donde las hormigas artificiales construyen soluciones de elementos de un conjunto finito de componente de soluciones disponibles $C = \{C_{ij}\}$, $i=1, \dots, n$; $j=1, \dots, |D_i|$. Una solución empieza a partir de una solución parcial vacía, $s^p = \emptyset$ y luego se extiende agregando un componente del conjunto de vecinos factibles $N(s^p) \subseteq C$. La elección de un componente de $N(s^p)$ es realizado de manera probabilística en cada paso de la construcción.

- **Búsqueda local**

Procedimiento para cuando se decida realizar algunas otras acciones que sean requeridas una vez que las soluciones hayan sido construidas. Puede ser usada para implementar un problema en específico.

- **Actualización de feromonas**

Procedimiento necesario para evitar una muy rápida convergencia del algoritmo, además de favorecer la exploración de nuevos espacios incrementando valores de feromonas asociados a buenas soluciones y disminuyendo aquellos rastros de feromona que generan malas soluciones.

Agregando a lo anterior, se identifican 5 pasos necesarios para resolver un problema por la meta heurística OCH, los cuales se describen a continuación:

- El primer paso consiste en representar el problema planteado en forma de un grafo con arcos y nodos, por el cual las hormigas artificiales viajarán y construirán soluciones.

- El segundo paso consiste en definir apropiadamente el rastro de feromona y el grado que ella influye. Este es un paso crucial en la implementación de un algoritmo OCH.
- El tercer paso consiste en definir apropiadamente la preferencia heurística para cada decisión que una hormiga ha de tomar mientras construye una solución.
- El cuarto paso consiste en escoger un algoritmo en específico, variante de OCH, y aplicarlo al problema que se quiere resolver. La elección dependerá del desarrollador y el contexto del problema.
- El quinto y último paso consiste en ajustar los parámetros del algoritmo OCH hasta que sean encontrados buenos valores para el problema. Dentro de este paso se realiza el proceso de experimentación, fundamental para hallar un buen nivel de exactitud en las soluciones.

A continuación se presentan con detenimiento dos variantes de la meta heurística OCH, siendo una de ellas el algoritmo propuesto en este trabajo:

A. Sistema de Hormigas (Ant System - AS)

Es una de las primeras variantes de OCH, desarrollada por Dorigo, Maniezzo y Colorni en 1991 [3], donde su principal característica es que los valores de la feromona son actualizados por todas las hormigas una vez que hayan completado su recorrido.

Primero, todos los rastros de feromona (τ_{rs}) son reducidos por una constante, implementando de esta manera la evaporación.

$$\tau_{rs} \leftarrow (1 - \rho) \cdot \tau_{rs},$$

Donde ρ es el ratio de evaporación entre 0 y 1.

Segundo, cada hormiga de la colonia deposita una cantidad de feromona acorde a la calidad de su solución (ΔT_{rs}^k).

$$\tau_{rs} \leftarrow \tau_{rs} + \Delta T_{rs}^k, \quad \forall a_{rs} \in S_k,$$

Donde r y s son dos nodos que forman el arco a , quien pertenece a la solución de la hormiga k .

Durante la construcción de soluciones, una hormiga k elige ir al siguiente nodo en base a la decisión probabilística (p_{rs}^k), por tanto tenemos:

$$p_{rs}^k = \begin{cases} \frac{[\tau_{rs}]^\alpha \cdot [\eta_{rs}]^\beta}{\sum_{s \in \mathcal{N}_r^k} [\tau_{rs}]^\alpha \cdot [\eta_{rs}]^\beta}, & \text{Si } s \in \mathcal{N}_k(r) \\ 0, & \text{Caso Contrario} \end{cases},$$

Donde α y β son dos parámetros que miden la importancia relativa entre la información heurística (η_{rs}) y el rastro de feromona (τ_{rs}). Además S es un nodo que pertenece al conjunto de vecinos factibles $\mathcal{N}_k(r)$ del nodo r .

B. Sistema Colonia de Hormigas (Ant Colony System - ACS)

Es uno de los primeros sucesores del Sistema de Hormigas y es el algoritmo seleccionado para este proyecto de fin de carrera. Esta variante presenta 3 principales modificaciones [4]:

I. Regla proporcional semi-aleatoria (pseudo-random proportional rule)

Esta vez se usa una regla diferente de transición, es decir, dada la hormiga k ubicada en el nodo r , q_0 es un parámetros en $[0,1]$ y q un valor aleatorio en $[0,1]$, el próximo nodo S es aleatoriamente escogido acorde a la siguiente distribución probabilística (p_{rs}^k).

Si $q \leq q_0$:

$$p_{rs}^k = \begin{cases} 1, & \text{Si } s = \arg \max_{s \in \mathcal{N}_k(r)} \{\tau_{rs} \cdot \eta_{rs}^\beta\} \\ 0, & \text{Caso Contrario} \end{cases},$$

C.C ($q > q_0$):

$$p_{rs}^k = \begin{cases} \frac{[\tau_{rs}]^\alpha \cdot [\eta_{rs}]^\beta}{\sum_{s \in \mathcal{N}_r^k} [\tau_{rs}]^\alpha \cdot [\eta_{rs}]^\beta}, & \text{Si } s \in \mathcal{N}_k(r) \\ 0, & \text{Caso Contrario} \end{cases}$$

Donde además α y β son dos parámetros que miden la importancia relativa entre la información heurística (η_{rs}) y el rastro de feromona (τ_{rs}). Además S es un nodo que pertenece al conjunto de vecinos factibles $\mathcal{N}_k(r)$ del nodo r .

De esta manera se consigue tener un doble propósito, cuando $Q \leq Q_0$ se explota el conocimiento disponible, escogiendo la mejor opción con respecto a la información heurística y el rastro de feromona. Sin embargo, si $Q > Q_0$, se aplica una exploración controlada, como la realizada en AS.

II. Actualización de feromona fuera de línea (offline pheromone update)

Sólo se considera una sola hormiga, la única quien ha generado la mejor solución global para realizar la actualización de la feromona.

La fórmula de actualización del rastro de feromona (τ_{rs}) es la siguiente:

$$\tau_{rs} \leftarrow (1 - \rho) \cdot \tau_{rs} + \rho \cdot f(C(S_{global-best})), \quad \forall a_{rs} \in S_{global-best}$$

Donde ρ es el ratio de evaporación en $(0, 1]$; r y S son dos nodos que forman el arco a , quien pertenece a la mejor solución encontrada.

III. Actualización de feromona paso a paso (online step-by-step pheromone trail update)

Se aplica una actualización en línea al momento de cada paso de la construcción de cada hormiga, es decir, realiza el proceso de evaporación y depósito de la feromona con el objetivo de generar diferentes soluciones aún por encontrar.

Cada vez que una hormiga recorre un arco a_{rs} , se aplica la regla:

$$\tau_{rs} \leftarrow (1 - \varphi) \cdot \tau_{rs} + \varphi \cdot \tau_0$$

Donde φ es un segundo coeficiente de evaporación en $(0,1]$.

1.3. Estado del arte

En esta sección se explica cómo se resuelve el problema planteado actualmente, para ello se revisa el problema de optimización a ser resuelto, se mencionan los procedimientos computacionales diseñados y se describen soluciones existentes, así como trabajos relacionados con el problema.

1.3.1. Problema del camino más corto (Shortest Path Problem)

Es el problema más conocido de una red de caminos donde el objetivo es encontrar el camino más corto entre dos pares de vértices [25].

Hay muchas variantes relacionadas al problema del camino más corto. Entre ellas tenemos las siguientes:

- El problema del camino más corto s-t (s-t shortest path problem): donde se busca el camino más corto desde un vértice s hasta un t dado.
- El problema del camino más corto desde/a una fuente (single-source shortest path problem): donde se busca los caminos más cortos desde un vértice s a todos los otros vértices en el grafo.
- El problema del camino más corto de todos los pares (all-pairs shortest path problem): donde se busca los caminos más cortos de cada vértice a otro.

1.3.2. Objeto de estudio: Problema del camino más corto s-t

Como se explica el problema del camino más corto dado dos vértices s y t consiste en hallar el camino más corto entre estos vértices en un grafo [21].

Objetivo

El objetivo es hallar la trayectoria óptima que deberá ser recorrida por la unidad encargada de realizar la solicitud de servicio de taxi.

Viabilidad

La solución es factible siempre y cuando se tenga un grafo con sus vértices o nodos conectados y no se presenten valores negativos en los arcos del grafo.

Formulación

Dado un grafo dirigido $G (V, E)$ y una función de variable real ponderada, el camino más corto p del vértice S al vértice t está dado por:

$$\text{Mín. } \sum_{p \in P} f(p)$$

Donde:

- V es un conjunto de vértices de 1 a n .
- E es un conjunto de aristas $\{(i, j): i, j \text{ pertenecen a } V\}$.

1.3.3. Algoritmos que resuelven el problema

Algoritmos Exactos

Para resolver este tipo de problemas se han desarrollado principalmente los siguientes algoritmos:

A. Algoritmo Dijkstra

El algoritmo de Dijkstra [5], también llamado algoritmo de caminos mínimos, es un algoritmo para la determinación del camino más corto dado un vértice origen al resto de vértices en un grafo dirigido y con pesos en cada arista.

La idea subyacente en este algoritmo consiste en ir explorando todos los caminos más cortos que parten del vértice origen y que llevan a todos los demás vértices; cuando se obtiene el camino más corto desde el vértice origen al resto de vértices que componen el grafo, el algoritmo se detiene.

El algoritmo es una especialización de la búsqueda de costo uniforme, y como tal, no funciona en grafos con aristas de costo negativo (al elegir siempre el nodo con distancia menor, pueden quedar excluidos de la búsqueda nodos que en próximas iteraciones bajarían el costo general del camino al pasar por una arista con costo negativo).

B. Algoritmo de Bellman-Ford

Este algoritmo genera los caminos mínimos desde un solo origen de un grafo ponderado (alguno de ellos podrán ser negativos). El algoritmo de Dijkstra logra esta misma tarea con un coste de tiempo menor, pero requiere que los pesos de las aristas no sean negativos.

Es por eso, que el algoritmo de Bellman-Ford [6], se utiliza únicamente cuando hay aristas negativas presentes en el grafo.

C. Algoritmo de Floyd-Warshall

Este algoritmo obtiene la ruta más corta entre todo par de nodos (todos contra todos). Itera sobre el conjunto de nodos que se permiten como nodos intermedios en los caminos [7].

Empieza con caminos de un solo salto y en cada paso ve si es mejor la ruta de la iteración anterior o si por el contrario conviene ir por otro nodo intermedio más.

Este algoritmo se puede aplicar a multitud de problemas, incluyendo el diseño de circuitos, el diseño de rutas de transporte, aproximaciones al problema del viajante de comercio, o como base de otros algoritmos más complejos.

Algoritmos Bio-inpirados

Como uno de los objetivos de este proyecto es aplicar un algoritmo bio-inpirado (Algoritmo Sistema Colonia de Hormigas), a continuación se presentarán algunos trabajos relacionados que dan solución al problema planteado:

- “Un algoritmo genético para resolver el problema del camino más corto para mapas de carreteras” [40].
- “Optimización de enjambre de partículas con meta heurísticas ruidosas para resolver el problema del camino más corto en una red” [41].
- “Algoritmos Optimización Colonia de hormigas para problemas del camino más corto” [42].
- “Resolviendo el problema del camino más corto por el algoritmo Colonia de Hormigas en vehículos con sistemas de navegación” [43].

1.3.4. Soluciones existentes y estudios relacionados

A continuación se detalla de manera breve algunas aplicaciones y trabajos relacionados:

1) Sistema digital de La Albericia

Sistema informático en la centralita de La Albericia ubicado en la ciudad de Santander, España, que busca atender los servicios y enviar la orden al terminal del taxi, este centro tiene a tiempo real la ubicación de cada vehículo, las llamadas de los clientes y dónde son requeridos los servicios [28].

Este sistema informático es capaz de asignar en 2 o 3 segundos el taxi más óptimo (más próximo).

2) Sistema Integral de Gestión del Transporte Público Regular de Viajeros

Sistema desarrollado con el fin de unificar la gestión de transporte en toda la flota de taxis en la ciudad de La Palma, España, busca mediante unidad de gestión, control y vigilancia controlar una flota de 120 taxis y distribuir el trabajo de forma adecuada, permitiendo que cualquier taxista pueda desplazarse a cualquier punto, en función de su posición [29].

3) Sistema Inteligente de Búsqueda de Rutas para el Recojo de Basura en Municipios

Sistema desarrollado por alumnos que pertenecen al grupo de inteligencia artificial de la Pontificia Universidad Católica del Perú y cuyo trabajo ocupó el tercer puesto en el concurso “Generando conocimiento” de EQUIPU, categoría perfiles [30].

4) Análisis, diseño e implementación de un algoritmo meta heurístico GRASP que permita resolver el problema de rutas de vehículos con capacidad

Tesis que implementa dos algoritmos meta heurísticos GRASP que permita resolver el problema de ruteo de vehículos con capacidad uniforme considerando la demanda compartida en caso el cliente tenga mayor demanda que la capacidad del vehículo de reparto [26].

En la tabla 1.2 se presenta un cuadro comparativo de ciertas características de las aplicaciones y trabajos presentados, así como la solución que se propone desarrollar.

	Sistema digital de la Albericia	Sistema Integral de Gestión del Transporte Público Regular de Viajeros,	Sistema Inteligente de Búsqueda de Rutas para el Recojo de Basura en Municipios	Análisis, diseño e implementación de un algoritmo meta heurístico GRASP que permita resolver el problema de vehículos con capacidad	Sistema de gestión de servicio de taxis, que optimice rutas de transporte (*)
Objetivo	Atender los servicios de taxi y ofrecer un servicio rápido y eficiente.	Gestionar el transporte por carretera de taxis mediante un control cartográfico pionero.	Proveer rutas óptimas como solución para el recojo de basura en municipios.	Proveer rutas óptimas de reparto que permitan satisfacer la demanda de clientes.	Proveer la mejor trayectoria para la solicitud de servicio de taxi.
Algoritmo	No especifica	No especifica	Algoritmo Meméticos	Algoritmo GRASP con mejoría 2-OPT	Algoritmo Sistema Colonia de Hormigas
Descripción Algorítmica	No especifica	No especifica	-Problema de ruteo de arcos capacitados en un grafo mixto. -Empleo de una matriz de caminos más cortos entre arcos. -Uso de una estructura de tipo cromosomática sin delimitadores de viaje.	-Problema de ruteo de vehículos con capacidad. -Problema de embalaje de paquetes en 3 dimensiones. -Empleo de una constante de tráfico.	-Problema del camino más corto. -Acotación del grafo. -Empleo de una variable empírica para el problema del tráfico (λ).
Lugar	Santander – España	La Palma – España	Lima – Perú	Lima – Perú	Lima – Perú
Año	2006	2008	2008	2009	2010

Tabla 1.2 Tabla comparativa de aplicaciones y trabajos informáticos existentes.

(*) Sistema propuesto en el presente proyecto de fin de carrera.

1.4. Plan del proyecto

1.4.1. Metodología de gestión

La planificación inicial del proyecto se realizará bajo la metodología PMBOK, documento usado por el PMI, relacionado a la gerencia de proyectos [10].

Según el PMI, todo proyecto se compone de cinco procesos, a saber: Inicio, Planeamiento, Ejecución, Control y Cierre. Estos procesos son temporales y en cada uno de ellos la disciplina propone acciones que giran en torno a nueve áreas de conocimiento.

La mencionada disciplina establece detalladamente todos los procesos incluidos en cada área de conocimiento: qué hacer, qué información utilizar como entrada (input) y qué salida (output) generar.

La razón por la que se consideró esta metodología es que permitirá explicar de manera sencilla y práctica como se gestionará el proyecto en curso y cuáles serán los principales procesos que se realizarán durante la gestión.

Adicionalmente se comentará como se vinculan dichos procesos.

Este proyecto abarcará las siguientes áreas del PMBOK:

- Administración de la Integración del Proyecto.
- Administración del Alcance del Proyecto.
- Administración del Tiempo del Proyecto.
- Administración de la Calidad del Proyecto.

Además, los entregables que serán realizados comprenderán lo siguiente:

- Estructura de descomposición del trabajo (EDT).
- Requerimientos del sistema.
- Presupuesto del proyecto.
- Medidas de control de calidad.

Administración de la Integración del Proyecto

a. Desarrollo de la Definición del Proyecto

Es el paso inicial para el presente proyecto. Para esto se toma en cuenta el problema observado, se hace un análisis del contexto del problema y se presenta una solución.

La definición del proyecto es crítica durante toda su vida, ya que se debe especificar el problema a resolver, así como un alcance preliminar, para evitar tener que retroceder fases del proyecto y poder cubrir nuevos alcances que no fueron definidos en esta etapa.

b. Monitoreo y Control del Proyecto

Durante el desarrollo del proyecto es importante realizar un monitoreo y control. Esto permitirá detectar errores, excesos en las estimaciones y aplicar las correcciones necesarias para solucionar estas deficiencias.

c. Control Integral del Cambio

Debido a que el proyecto realiza diferentes procesos en paralelo o en secuencia, los cambios realizados en algún proceso podrían afectar a uno o más procesos diferentes. Los cambios deben ser controlados para evitar futuros errores y proporcionar información suficiente para adaptarlos.

Además el proceso cumple la función de actualizar los entregables luego de los cambios realizados. Este proceso es incluido debido a las frecuentes observaciones que se darán por el asesor para mejorar el proyecto, así como la del jurado en la etapa previa a la sustentación, el cual no debe de afectar el alcance dado por el presente proyecto.

d. Cerrar el Proyecto

El proyecto, después de conseguir sus objetivos o al ser terminado por otras razones, requiere un cierre. El cierre consiste en verificar y documentar los resultados para formalizar la aceptación del producto del proyecto. Esto incluye la colección de archivos del proyecto, asegurándose que éstos reflejan las especificaciones finales, el análisis de éxito y efectividad del proyecto, archivando tal información para uso futuro.

En la figura 1.2 se muestra la relación de procesos dentro de la administración del presente proyecto.

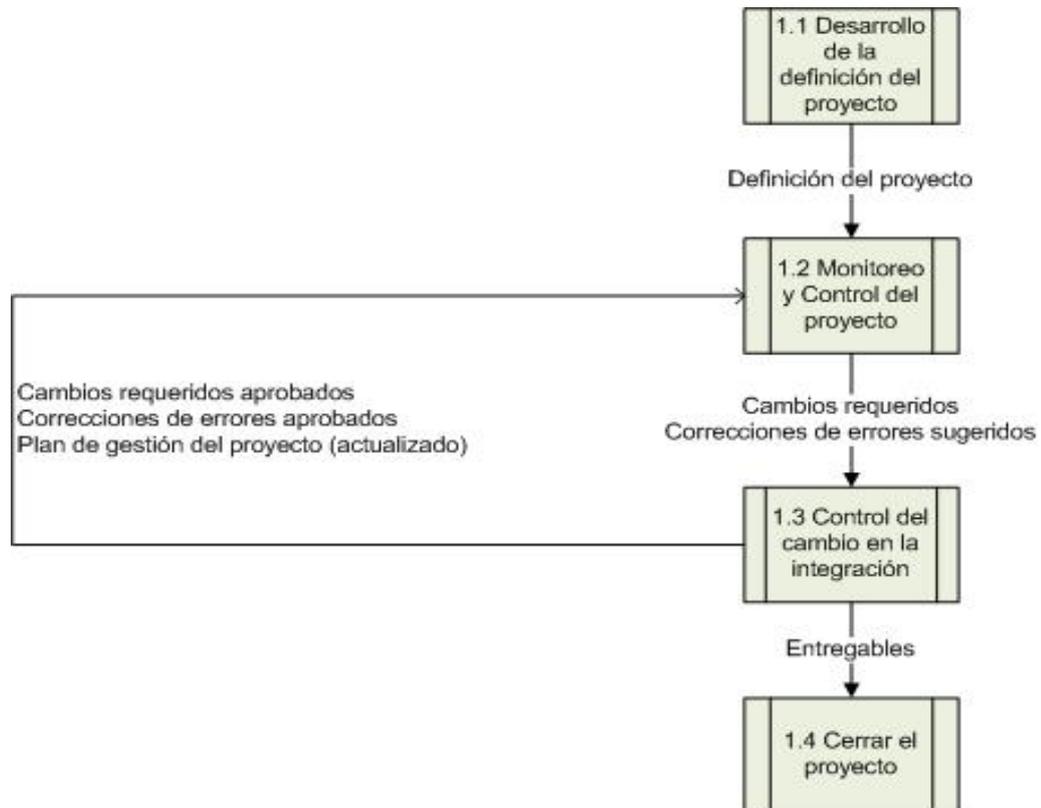


Figura 1.2 Plan de Proyecto: Administración de la Integración del Proyecto.

Administración del Alcance del Proyecto

a. Definición del Alcance

En esta etapa estarán plasmados los acuerdos tomados con el fin de cubrir con todo lo requerido, así como también se identificarán los objetivos del proyecto y los productos a entregar.

La administración del alcance del proyecto incluirá todos los procesos que se requieren para asegurar que el proyecto contenga únicamente el trabajo definido.

b. Creación del EDT

Se identificará y definirá las tareas a realizar (alcances). Todo trabajo o tarea que no esté incluida dentro del EDT (Estructura de Descomposición del Trabajo) está fuera del alcance del proyecto. Cada nivel descendiente representa una descripción más detallada de los elementos del proyecto.

En la figura 1.3 se muestra la relación de procesos dentro de la administración del alcance del presente proyecto.

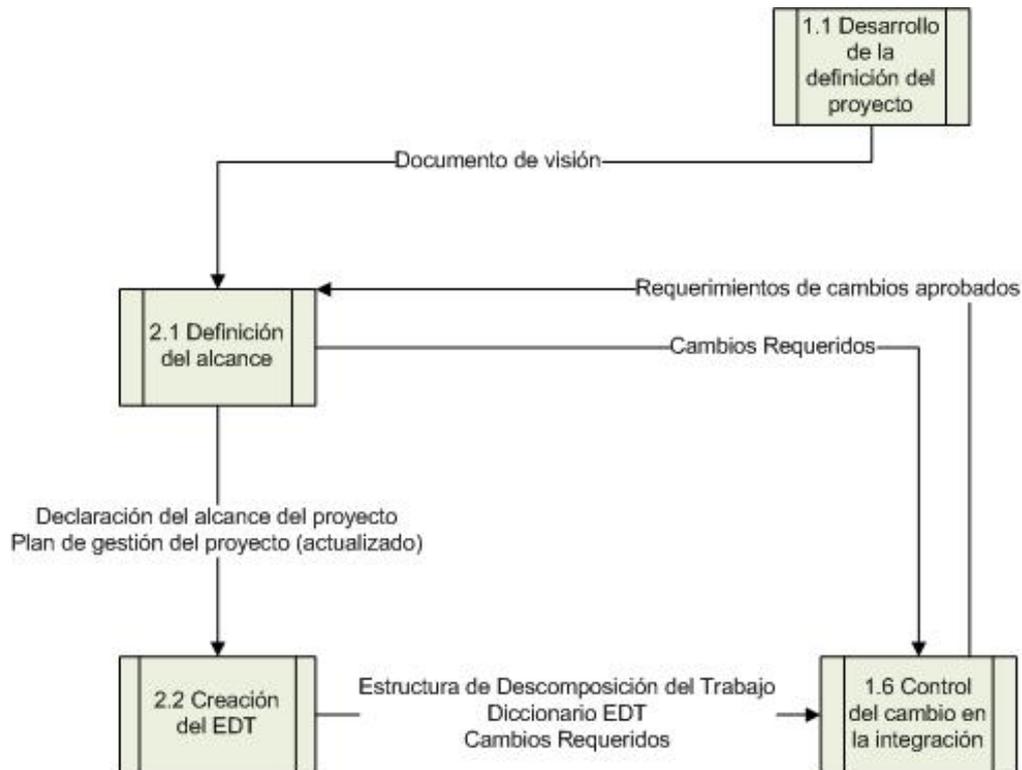


Figura 1.3 Plan de Proyecto: Administración del Alcance del Proyecto.

Administración del Tiempo del Proyecto

a. Definición de las actividades

Consiste en la identificación y documentación de las actividades que se van a realizar en el proyecto. Cada una de estas actividades debe tener un objetivo específico que ayude a terminar correctamente el proyecto.

b. Definición de secuencia de actividades

La definición de la secuencia de las actividades consiste en la documentación de las dependencias y relaciones entre diferentes actividades. Estas deben de colocarse en el orden correcto para facilitar la creación de un calendario realista y alcanzable.

c. Estimación de duración de actividades

La estimación de duración de actividades consiste en determinar el número de períodos necesarios para completar cada una de las actividades previamente identificadas.

Para calcular este número es necesario haber determinado la complejidad de la actividad a tratar.

En la figura 1.4 se muestra la relación de procesos dentro de la administración del tiempo del presente proyecto.

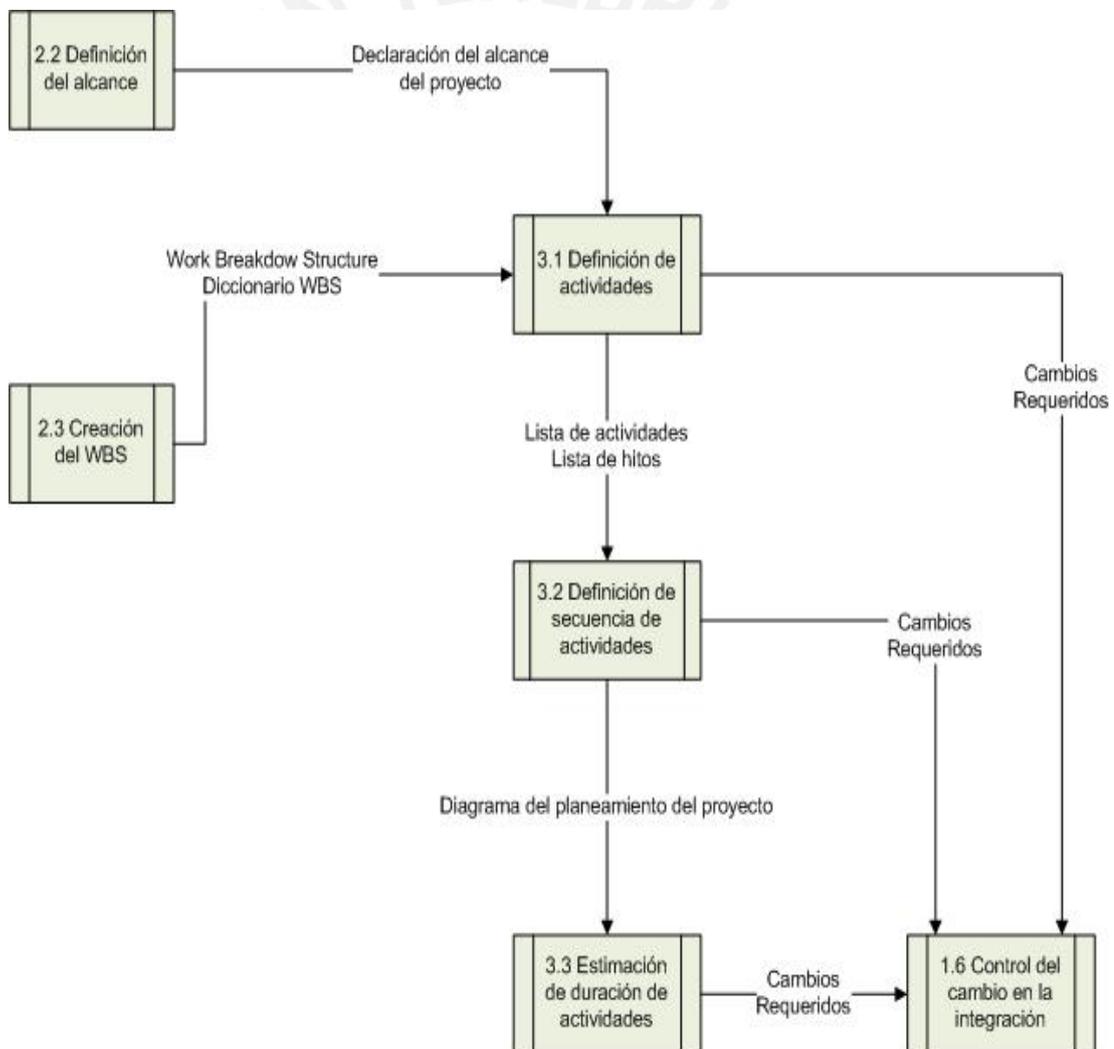


Figura 1.4 Plan de Proyecto: Administración del Tiempo del Proyecto.

Administración de la Calidad del Proyecto

a. Planeación de la Calidad

Identificar que estándares de calidad son relevantes al proyecto y determinar cómo satisfacerlos.

Para ello se planificará cuales serán las pruebas de calidad para tener mejores resultados de la solución y se tendrá en cuenta el tiempo necesario para desarrollar un producto de calidad.

b. Llevar a cabo el control de la calidad

Evaluar el desempeño general del proyecto de manera regular para así proveer la confianza de que el proyecto va a satisfacer los estándares de calidad relevantes.

Se realizará el control de la calidad del proyecto haciendo las modificaciones correspondientes a los documentos o actividades que no pasen este control.

En la figura 1.5 se muestra la relación de procesos dentro de la administración de la calidad del presente proyecto.

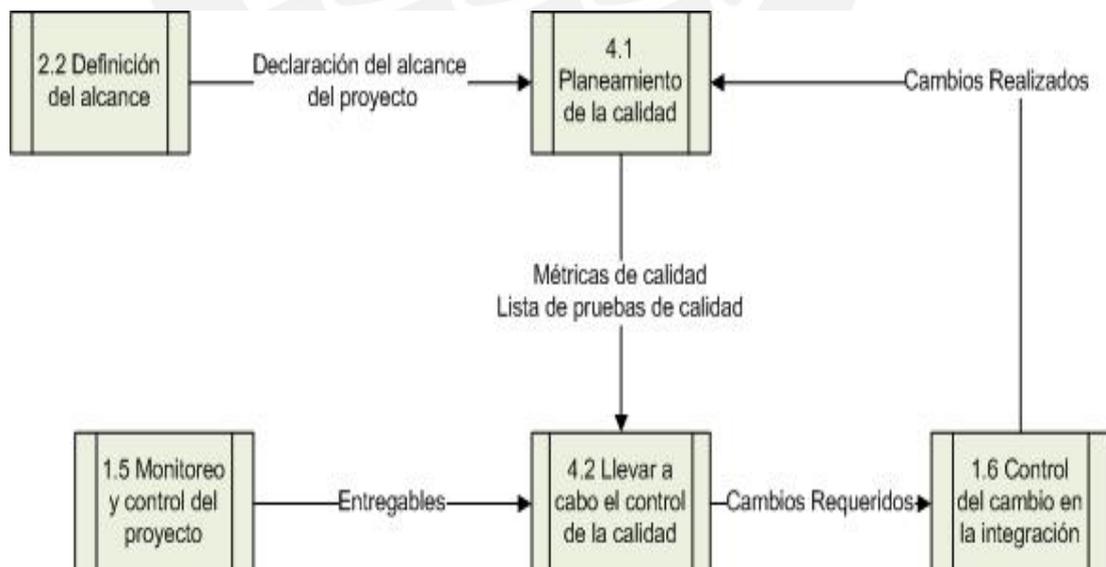


Figura 1.5 Plan de Proyecto: Administración de la Calidad del Proyecto.

1.4.2. Etapas del proyecto

Para poder realizar cada una de las tareas que se necesitan en el presente proyecto se ha elegido como base la metodología Proceso Unificado Racional (Rational Unified Process - RUP).

a. Elaboración del plan de tesis

Esta etapa considera la elaboración de este documento, el cual servirá de base para las siguientes etapas.

b. Etapa de concepción

Esta etapa considera el proceso de investigación que se realizó, incluyendo las entrevistas, levantamiento de información entre otras actividades con el fin de tener una idea clara del negocio y poder cumplir así con los objetivos del proyecto. Posteriormente se procede a realizar un análisis, donde se construye el modelo del negocio mostrando sus propiedades importantes, durante esta de análisis se abstrae de manera precisa lo que debe hacer el sistema deseado.

Finalmente se prosigue a realizar los documentos respectivos siendo uno de los más importantes la especificación de requerimientos.

c. Etapa de elaboración

La etapa de Elaboración analizará los requisitos y desarrollará el prototipo arquitectónico, que constituirá el diseño a alto nivel. Adicionalmente se realizará el diseño de la interfaz gráfica

d. Etapa de construcción

En esta etapa se realiza la implementación de los casos de uso identificados utilizando la herramienta computacional previamente definida. Se desarrolla la interfaz gráfica, el motor lógico y un plan de pruebas definidas para el software.

A continuación en la figura 1.6 se presenta la estructura de descomposición del trabajo (ETD) mediante una visión general de las etapas y principales entregables del proyecto.

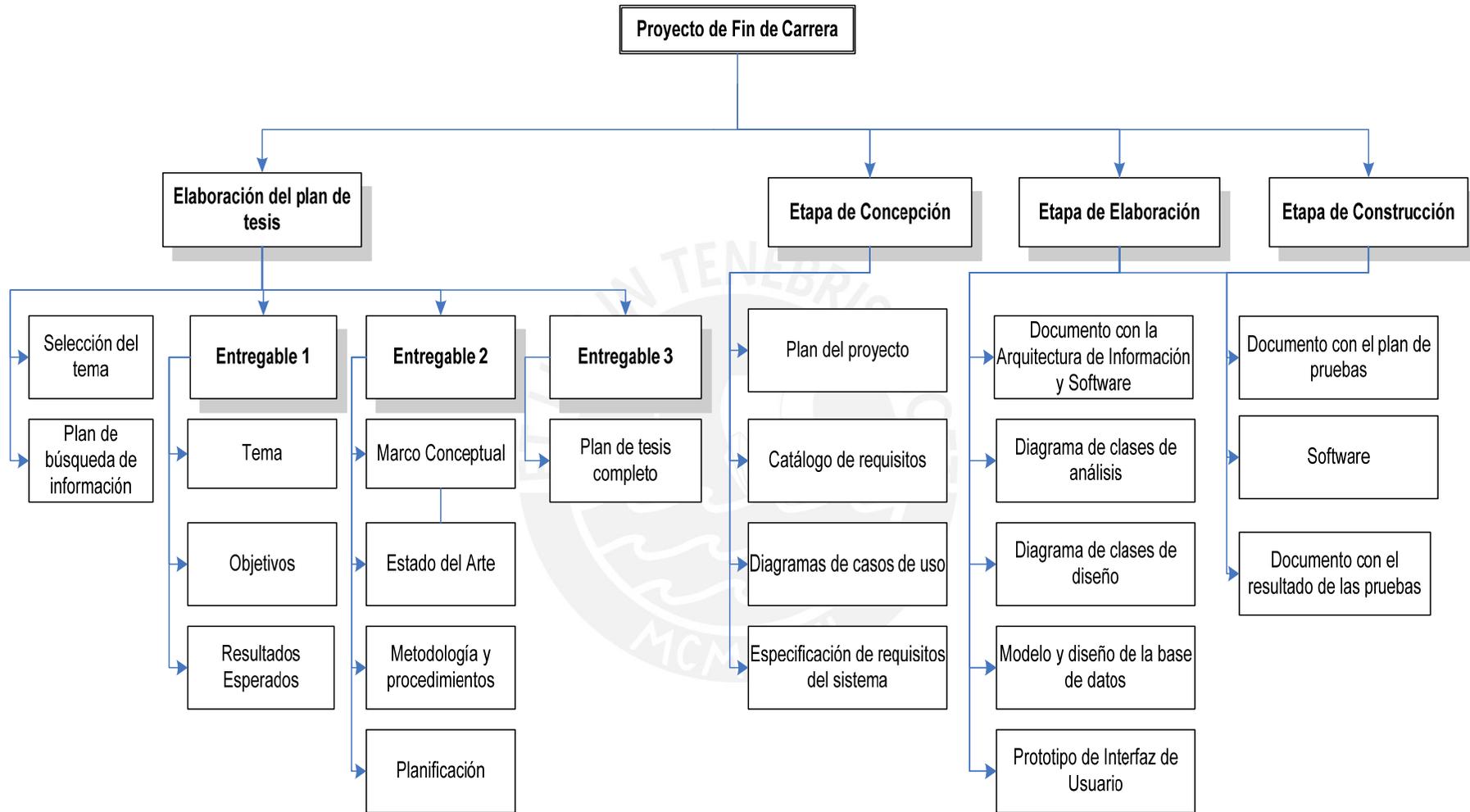


Figura 1.6 Estructura de descomposición del trabajo: EDT.

1.5. Descripción y sustentación de la solución

El objetivo del presente proyecto de fin de carrera es implementar un sistema de gestión de servicio de taxis que permita facilitar las siguientes actividades:

- a) Registrar las vías de las zonas que se tenga cobertura, obtenidos a partir de un modelo representativo de la ciudad.
- b) Registrar clientes.
- c) Registrar conductores.
- d) Registrar vehículos.
- e) Registrar tarifas.
- f) Atender y registrar de solicitudes de servicio.
- g) Anular solicitudes de servicio.
- h) Obtener las unidades disponibles y elegir la mejor disponible para brindar el servicio.
- i) Generar la ruta óptima para ofrecer el servicio.
- j) Generar reportes informativos y estadísticos de las operaciones.
- k) Generar gráficos estadísticos.

1.5.1. Componentes

El sistema se compone de 4 módulos claramente definidos:

- Módulo de Seguridad
Permite administrar a los usuarios asignándoles roles y permisos.
- Módulo de Mantenimiento
Permite realizar las operaciones de carga y registro de datos maestros.
- Módulo de recepción y asignación de servicios
Permite realizar las operaciones de atención de solicitudes de servicios de taxi.

- Módulo de apoyo a la gestión del servicio

Permite generar reportes informativos con gráficos estadísticos de los servicios realizados.

En la sección 2.2.1 se presenta un mayor detalle del contenido de cada módulo.

1.5.2. Ventajas

El sistema desarrollado permite las siguientes ventajas:

- a) Al llevar un registro y control de las unidades de la compañía, así mismo de los clientes, permite poder gestionar de mejor forma las solicitudes de servicio.
- b) La asignación del taxista mejor disponible y la generación de rutas óptimas propone reducir los costos de gasolina, disminuir el tiempo libre del taxista y eliminar las cancelaciones o anulaciones de los servicios, también llamados plantones.
- c) La utilización del algoritmo de optimización permitirá minimizar el tiempo que dure la trayectoria del servicio, condicionado principalmente por factores como el nivel de tráfico y distancia.
- d) El sistema busca ser una solución de tipo on-line (en línea), es decir, busca tener información real para brindar un servicio adecuado a las circunstancias de las zonas o calles.
- e) Es un sistema que brinda información útil, de apoyo a la gestión, que permite realizar un seguimiento de cómo se está llevando la gestión de servicios y proveer información exacta para los siguientes servicios.

Capítulo II: Análisis

En este capítulo se presentan primero la metodología de desarrollo del proyecto que se utiliza, luego se indican los requerimientos del sistema, posteriormente se analiza la viabilidad del proyecto y finalmente se crea una definición del sistema que constituya la base para todo el trabajo que se va a realizar.

2.1. Metodología de desarrollo del proyecto

Para el desarrollo del sistema de información de gestión de servicios de taxis que optimiza rutas de transporte se usará como base la metodología Proceso Unificado Racional (Rational Unified Process - RUP) [11], el cual emplea el Lenguaje Unificado de Modelado (Unified Modeling Language - UML), que es adecuada para el presente proyecto de tesis por las razones que se describirán seguidamente.

En la figura 2.1 se presenta un gráfico representativo de la metodología en el que se observa las etapas y fases para el desarrollo de un proyecto.

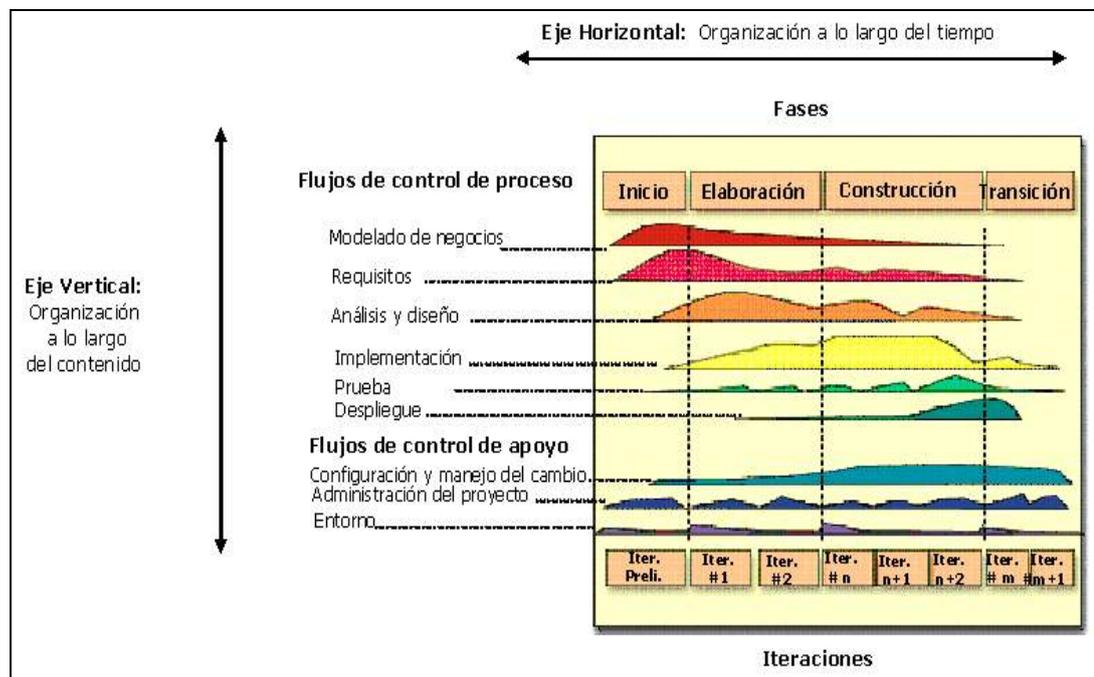


Figura 2.1 Metodología base para el proyecto: Estructura del RUP [11].

La selección de la metodología RUP es adecuada porque contempla características fundamentales para el desarrollo del presente proyecto; algunas de las características de RUP que permite esto se presentan a continuación [31]:

- Dirigido por los Casos de Uso, herramienta que especifica los requisitos del sistema y guía las etapas de diseño e implementación.
- Ciclo de vida iterativo-incremental.
- Gestión de requisitos.
- Modelamiento visual a través del Unified Modeling Language (UML).

A continuación se describe brevemente cada una de las fases de la metodología del proyecto, así como sus objetivos y resultados:

Concepción

Durante la fase de concepción o inicio se define la oportunidad y el alcance. En el sistema se identificarán todas las entidades con las que se trata y se definirá la interacción a un nivel de abstracción.

Los objetivos de esta fase son:

- Establecer el ámbito del sistema y sus límites.
- Encontrar los Casos de Uso del sistema, los escenarios básicos que definen la funcionalidad.
- Estimar el coste en recursos y tiempo de todo el proyecto.

Los resultados de la fase de concepción del sistema deben ser:

- Plan del proyecto, mostrando fases e iteraciones.
- Catálogo de Requisitos completo.
- Modelo inicial de Casos de Uso (10-20% completado).
- Un modelo de negocios si fuese necesario.

Al terminar la fase de inicio se deben comprobar los criterios de evaluación para continuar:

- Entendimiento de los requisitos, como evidencia de la fidelidad de los Casos de Uso principales.
- Credibilidad en las estimaciones, prioridades, riesgos y procesos de desarrollo.

Elaboración

El propósito de la fase de elaboración en el presente proyecto es analizar el dominio del sistema, establecer los cimientos de la arquitectura y diseño, y eliminar los mayores riesgos que puedan presentarse en el sistema.

Los objetivos de esta fase son:

- Definir, validar y cimentar la arquitectura del sistema.

Al terminar deben obtenerse los siguientes resultados para el sistema:

- Un modelo de Casos de Uso completo: todos los caso con sus especificaciones y actores identificados.
- Identificación de entidades (diagrama de clases y diseño).
- Descripción de la arquitectura software.
- Modelo inicial de la base de datos.

En la fase de elaboración se actualizan todos los productos de la fase de inicio.

Construcción

Durante esta fase todos los componentes, características y requisitos deben ser implementados, integrados y probados en su totalidad, obteniendo una versión aceptable del producto.

Los objetivos concretos son:

- Analizar el dominio de la construcción del algoritmo.
- Se realizará el producto de software integrado y corriendo en la plataforma adecuada.

Los resultados de la fase de construcción deben ser:

- Modelos Completos (Casos de Uso, Análisis, Diseño, Despliegue e Implementación).
- Descripción de tecnología y patrones de programación.
- Realización del diseño de pruebas.
- Arquitectura íntegra (mantenida y mínimamente actualizada).
- Prototipo Operacional.

Los criterios de evaluación de esta fase son los siguientes:

- El producto es estable y maduro como para ser entregado.

2.2. Identificación de requerimientos

Para llevar a cabo la identificación de requerimientos del sistema se contactó con una empresa de taxis del Perú, la elección de dicha empresa fue tomada principalmente por sus objetivos dentro de la industria del taxi y por tener la tecnología como uno de sus valores para el logro de sus metas.

En la tabla 2.1 se presenta la lista de exigencias para el módulo de seguridad, la cual permite la recopilación de los requerimientos funcionales de este módulo.

Módulo de Seguridad				
Nº	Descripción	Dif.	Prio.	Ex.
1	El sistema permitirá registrar, modificar, eliminar y consultar usuarios.	3	2	E
2	El sistema permitirá registrar, modificar, eliminar y consultar roles.	2	2	E
3	El sistema permitirá asignar un rol a un usuario.	3	2	E
4	El sistema permitirá brindar autenticación de usuarios mediante una contraseña.	3	2	E
5	El sistema permitirá modificar la contraseña del usuario.	3	2	E

Tabla 2.1 Módulo de Seguridad.

En la tabla 2.2 se presenta la lista de exigencias para el módulo de mantenimiento, la cual permite la recopilación de los requerimientos funcionales de este módulo.

Módulo de Mantenimiento				
Nº	Descripción	Dif.	Prio.	Ex.
1	El sistema permitirá el registro, modificación y eliminación de clientes.	3	2	E
2	El sistema permitirá el registro, modificación y eliminación de los conductores.	2	2	E

Módulo de Mantenimiento				
Nº	Descripción	Dif.	Prio.	Ex.
3	El sistema permitirá el registro, modificación y eliminación de los vehículos.	3	2	E
4	El sistema permitirá el registro, modificación y eliminación de las tarifas y cotizaciones.	3	2	E
5	El sistema permitirá el registro, modificación y eliminación de lugares de referencia.	3	2	E
6	El sistema permitirá el registro y modificación de parámetros de optimización.	3	2	E
7	El sistema permitirá el registro y modificación de parámetros del sistema.	3	2	E
8	El sistema permitirá el registro de las vías.	1	1	E
9	El sistema permitirá la actualización del costo de las vías.	2	1	E
10	El sistema permitirá la actualización de los horarios de operación para brindar servicio.	3	2	D

Tabla 2.2 Módulo de Mantenimiento.

En la tabla 2.3 se presenta la lista de exigencias para el módulo de Recepción y Asignación de servicios, la cual permite la recopilación de los requerimientos funcionales de este módulo.

Módulo de Recepción y Asignación de servicios				
Nº	Descripción	Dif.	Prio.	Ex.
1	El sistema permitirá el registro, modificación y anulación de una solicitud de servicio.	1	1	E
2	El sistema permitirá el registro, modificación y anulación de una reserva.	1	1	E
3	El sistema permitirá la asignación de la unidad más disponible para un servicio.	2	1	E

Módulo de Recepción y Asignación de servicios				
Nº	Descripción	Dif.	Prio.	Ex.
4	El sistema permitirá realizar un seguimiento del estado de las unidades.	2	3	E
5	El sistema permitirá realizar consultas de clientes.	2	2	E
6	El sistema permitirá realizar consultas de solicitudes.	2	2	E
7	El sistema permitirá realizar consultas de reservas.	2	2	E
8	El sistema permitirá obtener y registrar la ruta óptima para atender un determinado servicio.	1	1	E
9	El sistema permitirá mostrar una interpretación de la ruta óptima obtenida.	1	2	D

Tabla 2.3 Módulo de Recepción y Asignación de servicios.

En la tabla 2.4 se presenta la lista de exigencias para el módulo de Apoyo a la Gestión, la cual permite la recopilación de los requerimientos funcionales de este módulo.

Módulo de Apoyo a la Gestión de servicios				
Nº	Descripción	Dif.	Prio.	Ex.
1	El sistema permitirá generar reportes de solicitudes atendidas.	2	1	E
2	El sistema permitirá generar reportes de reservas atendidas.	2	1	E
3	El sistema generará reportes estadísticos mensuales y anuales de solicitudes tanto realizadas como anuladas.	1	1	E
4	El sistema generará reportes estadísticos mensuales y anuales de solicitudes y reservas por cada distrito.	1	1	E

Módulo de Apoyo a la Gestión de servicios				
Nº	Descripción	Dif.	Prio.	Ex.
5	El sistema generará reportes estadísticos de distritos con mayor frecuencia de solicitudes.	1	1	E
6	El sistema permitirá mostrar gráficos en los reportes estadísticos.	1	1	E

Tabla 2.4 Módulo de Apoyo a la Gestión de servicios.

En la tabla 2.5 se presenta la lista de exigencias para los requerimientos no funcionales con los que el sistema deberá cumplir.

Requerimientos No funcionales				
Nº	Descripción	Dif.	Prio.	Ex.
1	El sistema deberá permitir terminar la sesión, cerrando todas las ventanas devolviendo el control a la pantalla principal.	3	2	E
2	El sistema trabajará bajo el sistema operativo Linux.	2	2	D
3	El sistema trabajará bajo el sistema operativo Windows.	3	3	E
4	El sistema empleará el manejador de base de datos MySQL.	3	2	E
5	El sistema deberá mostrar al usuario el resultado de sus acciones (mensajes de confirmación, error, etc.).	3	1	E
6	El sistema deberá tener un entorno gráfico de ventanas.	3	1	E
7	La herramienta incluirá un instalador y manual de usuario.	3	3	D
8	La arquitectura del sistema seguirá el modelo cliente-servidor.	3	1	E

Tabla 2.5 Requerimientos No Funcionales.

Dif/Prio: Dificultad / Prioridad

Valores	Descripción
1	Alta
2	Media
3	Baja

Ex: Exigible o deseable

Valores	Descripción
E	Exigible
D	Deseable

2.2.1. Diagramas de caso de uso

En este punto se presentan los casos de uso que se han elaborado para el sistema a partir de la especificación de requisitos, los cuales permiten mostrar en alto nivel las funcionalidades que el sistema realizará. En el anexo A se visualizará el detalle de cada caso de uso descrito.

I. Catálogo de actores

En la figura 2.2 se muestra el Catálogo de Actores que interactúa con el sistema, los que se describen a continuación.

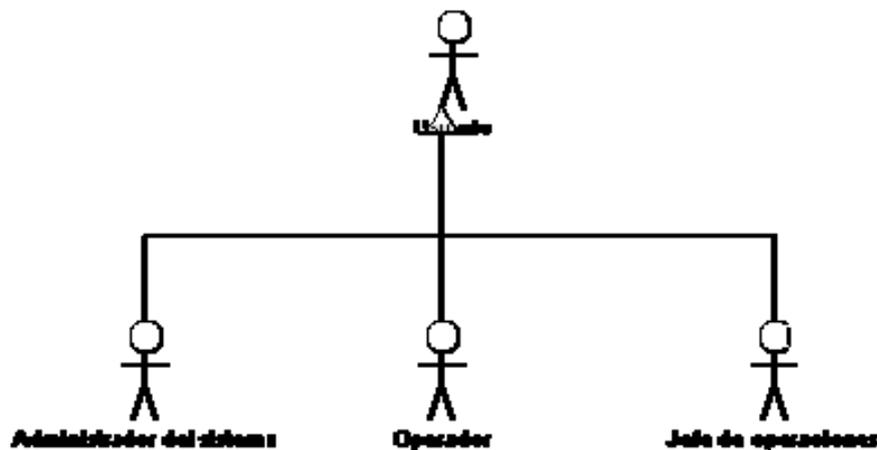


Figura 2.2: Catálogo de actores.

Administrador del Sistema:

Usuario que debe conocer por completo el sistema y que poseerá conocimientos suficientes sobre computación. Esta persona que registrará y brindará acceso a través de perfiles a los otros usuarios del sistema, siguiendo las políticas de seguridad y ética. Además se encargará de configurar y dar soporte al sistema para que cumpla con los objetivos que desea la empresa.

Operador:

Este usuario maneja las operaciones del negocio que corresponde a los procesos operativos de la empresa como atención de solicitudes de servicios, registro de servicios a atender y anulación de solicitudes. Además de asignar la unidad más disponible a una solicitud de servicio.

Jefe de Operaciones:

Es aquel usuario que realiza los mantenimientos pertinentes del sistema para las operaciones de la empresa, así como tiene la función de generar reportes en base a la información recibida. Además estará a cargo de la administración de los datos maestros necesarios para que el operador pueda interactuar con el sistema.

Se asume que este usuario posee educación superior y por lo tanto posee un manejo intermedio de computación como también debe tener mayor conocimiento del negocio de la empresa.

II. Casos de uso

El sistema en desarrollo propuesto consta de 4 módulos: Módulo de Seguridad, Módulo de Mantenimiento, Módulo de Recepción y Asignación de Servicios, y Módulo de apoyo a la gestión de servicios. A continuación se presenta los casos de uso de cada módulo.

A. Módulo de Seguridad

Este módulo contiene los casos de uso relacionados con la administración de usuarios y asignación de roles que determinará su comportamiento en el sistema.

- ✓ Que permita el registro de usuarios del sistema.
- ✓ Que permita el registro de roles para clasificar a los usuarios.
- ✓ Que permita la asignación de roles a los usuarios.
- ✓ Que permita la modificación de la contraseña del usuario.

Los casos de uso que se han elaborado para este módulo son los que se muestran gráficamente en la figura 2.3.

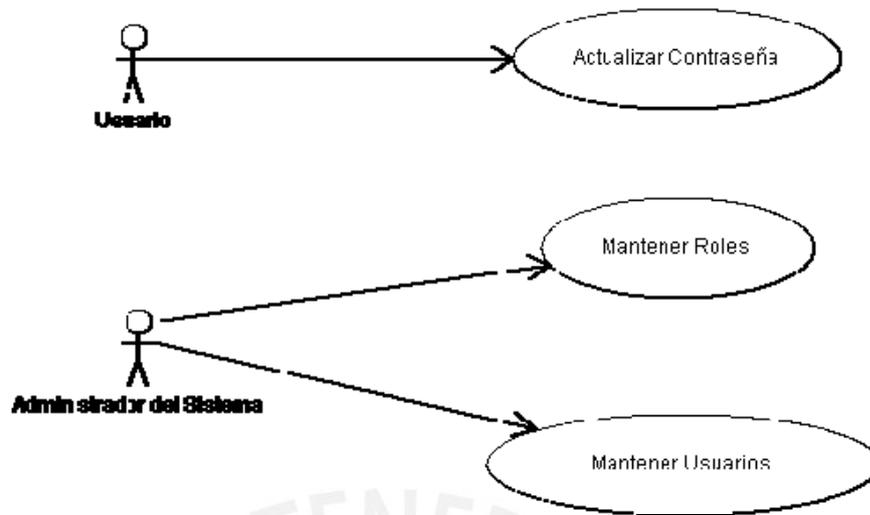


Figura 2.3 Diagrama de casos de uso del módulo de Seguridad.

Actualizar contraseña

El sistema permitirá al usuario cambiar la contraseña para acceder al sistema.

Mantener roles

El sistema permitirá administrar los roles que permitirán controlar la participación de cada usuario del sistema.

Mantener usuarios

El sistema permitirá administrar a los usuarios del sistema para poder definir posteriormente el rol que toma cada uno y su participación en el negocio.

B. Módulo de Mantenimiento

Este módulo contiene los casos de uso relacionados con la administración de datos maestros del sistema para funcionalidades que cumplan los objetivos que se detallan a continuación.

- ✓ Que permita el registro de las vías (avenidas, calles, pasajes, etc.), donde circulan las unidades, además este registro será actualizado de tal manera que tenga una información real para el sistema.
- ✓ Que permita el registro de elementos fundamentales para la operación como clientes, conductores y vehículos.

- ✓ Que contenga un catálogo con las unidades de la empresa de taxis.
- ✓ Que permita el registro de tarifas y cotizaciones del servicio.
- ✓ Que permita el registro de parámetros del sistema y de optimización.

Los casos de uso que se han elaborado para este módulo son los que se muestran gráficamente en las figuras 2.4, 2.5 y 2.6.



Figura 2.4 Diagrama de casos de uso del módulo de Mantenimiento.

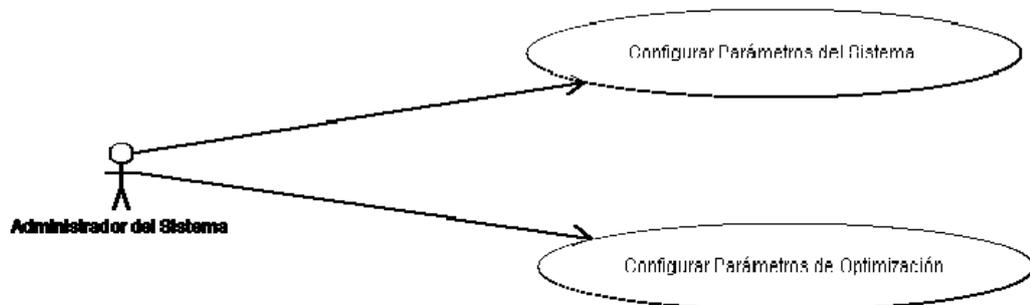


Figura 2.5 Diagrama de casos de uso del módulo de Mantenimiento.

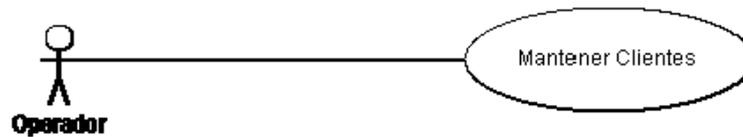


Figura 2.6 Diagrama de casos de uso del módulo de Mantenimiento.

Configurar parámetros del sistema

El sistema permitirá la configuración de parámetros generales del sistema relacionados a las actividades operativas.

Configurar parámetros de optimización

El sistema permitirá la configuración de parámetros del algoritmo a emplear para generar la ruta óptima.

Mantener clientes

El sistema permitirá poder administrar la información de los clientes organizacionales y particulares.

Mantener conductores

El sistema permitirá administrar la información de los conductores o taxistas, así mismo sirve para realizar un seguimiento sobre sus labores.

Mantener vehículos

El sistema permitirá poder administrar la información sobre las unidades vehiculares con las que cuenta la empresa.

Mantener tarifas y cotizaciones

El sistema permitirá administrar el costo de los servicios ofrecidos.

Mantener lugares de referencia

El sistema permitirá administrar lugares tomados como referencia para una mejor ubicación como centro de estudios, centros comerciales, etc.

Cargar Grafo

El sistema permitirá registrar una representación de las vías para obtener la trayectoria óptima.

Configurar vías

El sistema permitirá configurar las vías que formarán parte del recorrido de las unidades, limitadas por la cobertura operativa de la empresa.

Configurar horarios de operación

El sistema permitirá actualizar el intervalo de horas durante el día por el cual se pueda distinguir tiempos con mayor congestión, tráfico, etc.

C. Módulo de Recepción y Asignación de Servicios

Este módulo presenta los casos de uso que corresponden a las funciones principales para atender un servicio con funcionalidades que cumplan los siguientes objetivos.

- ✓ Que permita registrar solicitudes de servicio y reservas.
- ✓ Que permita consultar solicitudes de servicio y reservas.
- ✓ Dar seguimiento a las unidades disponibles.
- ✓ Asignar una unidad al servicio.
- ✓ Generar la ruta óptima para ofrecer el servicio.

Los casos de uso que se han elaborado para este módulo son los que se muestran gráficamente en la figuras 2.7, 2.8 y 2.9.

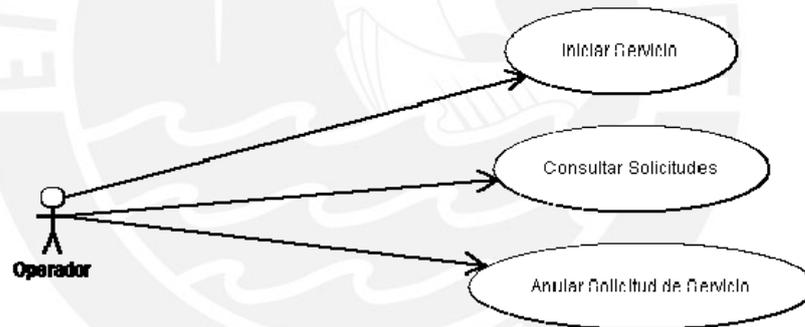


Figura 2.7 Diagrama de casos de uso del módulo de Recepción y Asignación de servicios.

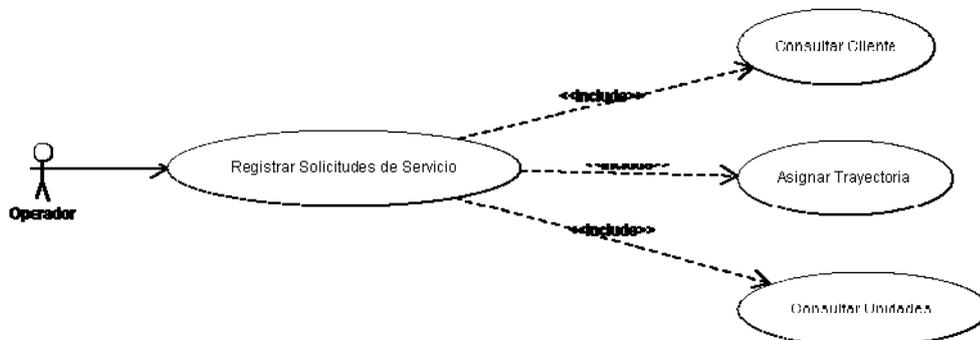


Figura 2.8 Diagrama de casos de uso del módulo de Recepción y Asignación de servicios.

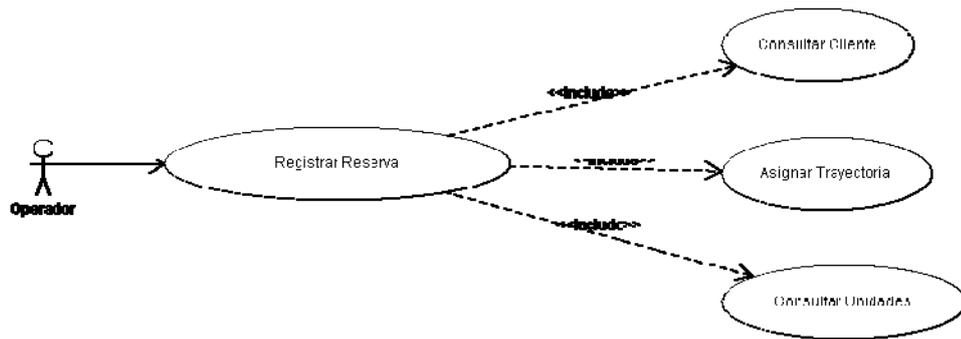


Figura 2.9 Diagrama de casos de uso del módulo de Recepción y Asignación de servicios.

Registrar Solicitudes de Servicio

El sistema permitirá el registro de una solicitud de servicio cuando el cliente llame a la central.

Registrar Reservas

El sistema permitirá el registro de una reserva cuando el cliente llame a la central.

Consultar clientes

El sistema permitirá realizar la consulta a determinado cliente para seguir con el registro del servicio.

Consultar solicitudes

El sistema permitirá realizar la consulta de solicitudes y reservas para conocer el estado en el que se encuentran.

Consultar unidades

El sistema permitirá realizar el seguimiento de las unidades actualmente disponibles al momento de generar una solicitud de servicio.

Iniciar Servicio

El sistema permitirá registrar aquellas solicitudes como servicios una vez la unidad se encuentre con el cliente y listo para ofrecer el servicio.

Obtener Trayectoria Óptima

El sistema permitirá asignar la ruta óptima para cada servicio por atender.

Anular Solicitudes

El sistema permitirá anular solicitudes de servicios y reservas ya sea por algún error operativo o el cliente no cuenta con las aprobaciones necesarias.

D. Módulo de apoyo a la gestión

Este paquete contiene los casos de uso para brindar información de apoyo a la gestión con funcionalidades que cumplan ciertos objetivos.

- ✓ Obtener reportes de solicitudes y reservas realizadas.
- ✓ Obtener reportes de solicitudes anuladas.
- ✓ Obtener reportes de servicios reservados.
- ✓ Proporcionar gráficos estadísticos.

Los casos de uso que se han elaborado para este módulo son los que se muestran gráficamente en la figura 2.10.

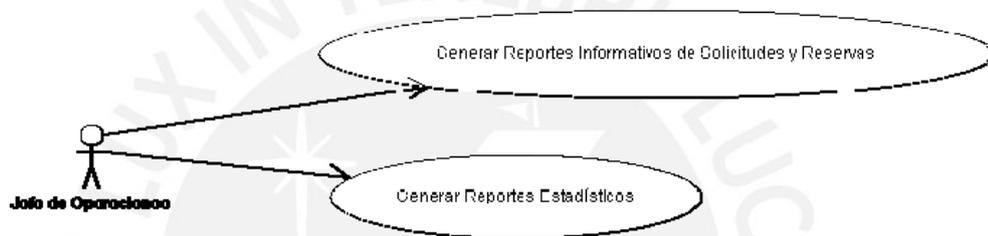


Figura 2.10 Diagrama de casos de uso del módulo de Apoyo a la Gestión.

Generar reportes informativos de solicitudes y servicios

El sistema permitirá generar reportes con la información de solicitudes y reservas atendidas, entre las cuales se encuentran: identificador de solicitud, fecha de registro, punto de recojo, punto de destino, costo, cliente y conductor.

Generar reportes estadísticos

El sistema permitirá generar reportes con información y gráficos estadísticos.

Para las solicitudes y reservas de servicio, el sistema generará reportes estadísticos mensuales y anuales por distrito.

Para las solicitudes y reservas de servicio tanto realizadas como anuladas, el sistema generará reportes estadísticos mensuales y anuales.

Para los distritos donde hay cobertura del servicio, el sistema generará reportes que permitirá identificar cuales tienen una mayor frecuencia de solicitudes.

2.3. Análisis de la solución

2.3.1. Viabilidad

El objetivo de esta sección del documento es realizar un estudio detallado del sistema que permita satisfacer las necesidades del usuario usando las tecnologías disponibles y teniendo además mecanismos de control para su buen funcionamiento. De esta manera se determina en forma objetiva si el proyecto es viable.

Esto se logra mediante la evaluación de una serie de factores, entre los que se encuentran la viabilidad económica, (es decir los costos de desarrollo del sistema sopesado con los beneficios obtenidos por su implementación), viabilidad técnica (función, rendimiento y restricciones que puede afectar la implementación de un sistema) y alternativas (una evaluación de las posibles soluciones alternativas al desarrollo del sistema.)

2.3.2. Análisis Técnico y Económico

A continuación se presenta una evaluación del sistema que comprende un análisis técnico y económico del mismo.

Análisis Técnico

Para desarrollar el sistema propuesto basado en su descripción se usará el lenguaje de programación Java JDK 6, para el almacenamiento de datos se usará el motor de base de datos MySQL Server 5.1, el sistema operativo será Windows y se trabajará con IDE de NetBeans. Es decir, técnicamente es viable porque el sistema ha implementar tiene los medios técnicos para su desarrollo y construcción.

Análisis Económico

El análisis económico involucra a todos los factores económicos que nos permitan llevar a cabo el desarrollo del sistema. Dando como resultado final un monto que nos indica cuánto costaría desarrollar el sistema y por tanto poder ver en el aspecto económico si el sistema es viable o no.

A continuación, en la tabla 2.6 se presenta un presupuesto que describe los costos involucrados en el desarrollo del sistema:

Presupuesto para el desarrollo				
Actividades	Esfuerzo (Días)	Horas Diarias	Esfuerzo Total (Horas) (*)	Costo (USD)
Laborales				
Análisis	60	3.5	210	840
Diseño	20	3.5	70	280
Implementación	110	3.5	385	1540
Pruebas	10	3.5	35	140
Total Laborales				2800
No Laborales				
Transporte				150
Materiales				70
Impresiones				120
Otros gastos				30
Total No Laborales				370
Total				3170

Tabla 2.6 Presupuesto para el desarrollo.

(*) Costo Horas – Hombre: \$ 4.00.

2.3.3. Asignación de Funciones

Hardware

El hardware cumplirá la función de servir de soporte para los aplicativos que se desarrollen. El hardware hace funcionar el sistema operativo sobre el cual funciona el sistema de información.

El hardware, las computadoras y el servidor, debe brindar una disponibilidad de 24 horas al día, que es el tiempo diario de trabajo de la mayoría de empresas que brinda servicios de taxis. Durante el tiempo de funcionamiento el hardware no debe tener fallas.

Se usará la arquitectura Cliente – Servidor por prestarse a las características de una central, además una organización de este tipo no requiere de una arquitectura complicada para desenvolver sus labores con eficiencia.

Software

El software empleado en la solución está compuesto por el sistema operativo Windows, el lenguaje de programación Java y el manejador de base de datos MySQL Server que alojará la información.

El sistema operativo funciona como plataforma para el funcionamiento del sistema de información, se ha seleccionado Windows por ser amigable y familiar para los usuarios; el lenguaje de programación Java que servirá para su construcción por ser totalmente libre y contar con entornos de desarrollo integrado (IDE's) también gratuitos; el tipo de empresas, para lo cual está destinada este proyecto, trata de invertir la menor cantidad de capital en tecnologías de información, es por ello que uno de los criterios de evaluación fue el costo.

Usuarios

Básicamente, el sistema será manejado por tres usuarios. Los operadores, quienes realizarán las labores anteriormente descritas como la recepción de solicitudes de servicios, asignación de unidades y generación de rutas óptimas. El jefe de operaciones, quien estará encargado del registro de conductores y unidades de trabajo, de las tarifas y cotizaciones, de datos para la información real del estado de las zonas y calles, además de generar los reportes pertinentes. Por último el usuario administrador, quien tendrá acceso a toda la funcionalidad del sistema.

Para este sistema se contará con un determinado rol y sus respectivos permisos.

2.3.4. Restricciones de Costo y Tiempo

El presente proyecto de fin de carrera no tiene restricciones económicas por utilizar herramientas libres, como es el caso del entorno de desarrollo integrado NetBeans para el lenguaje de programación Java, MySQL Workbench para modelar la base de datos, ArgoUML para el diseño de diagramas bajo el estándar UML e iReport para diseñar los reportes.

Por otra parte en lo que respecta a recursos humanos, se debe de tener en cuenta el costo del alumno involucrado por el desarrollo del software. (Ver Tabla 2.6)

En cuanto al tiempo, el presente proyecto de fin de carrera presenta restricciones ajustadas al plan de proyecto especificado en el Capítulo I y debe de estar disponible en la fecha definida para su entrega.

2.3.5. Diagrama de clases de análisis

Las clases de análisis que se han definido para la realización del sistema son las que se muestran en el anexo B.

2.3.6. Diagrama de paquetes

En la presente sección se describe los módulos del sistema y las dependencias que existen entre estos se muestran en la figura 2.11.

- Módulo Seguridad.
- Módulo Mantenimiento.
- Módulo Recepción y Asignación de Servicios.
- Módulo de Apoyo a la Gestión.

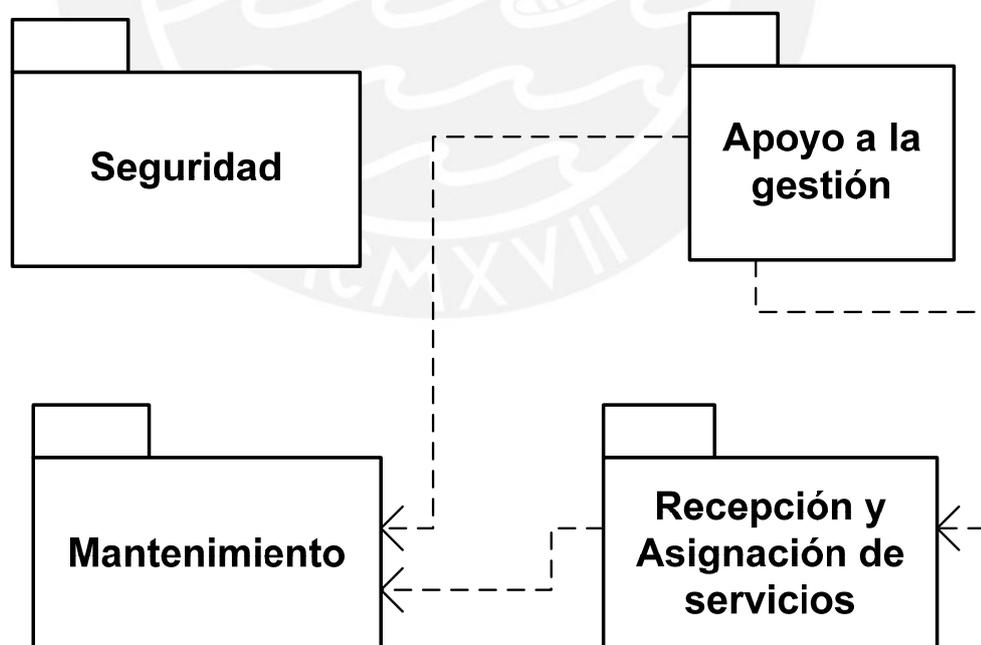


Figura 2.11 Diagrama de Paquetes.

Capítulo III: Diseño

El objetivo de este capítulo es presentar la arquitectura del sistema y el diseño de las interfaces gráficas de tal manera que satisfaga las necesidades del presente proyecto.

3.1. Arquitectura de la solución

La decisión de qué arquitectura se utilizará es fundamental, por tanto debe seleccionarse de acuerdo a las expectativas de crecimiento y los servicios que se deseen ofrecer. En este sentido, la definición de la arquitectura de la solución para este proyecto busca que la aplicación sea flexible, coherente, eficiente y sencilla de utilizarse, pues éste es el diseño de más alto nivel de la estructura de un sistema de información.

Algunas consideraciones que la arquitectura deberá de tomar respecto al sistema son:

- El sistema debe soportar las funciones de la empresa definidas anteriormente en la lista de requerimientos (Ver sección 2.2), plasmándolo en una base de datos relacional.
- El acceso al sistema se realizará por medio de las computadoras que se encuentren conectadas a través de una misma red y cuenten con el sistema instalado.
- El sistema proporcionará una interfaz gráfica para que el usuario pueda realizar las operaciones que le correspondan.

La arquitectura empleada para la programación del Sistema de Información es de 3 capas. La interacción entre éstas es mostrada en la figura 3.1 y se describe cada capa en la tabla 3.1.

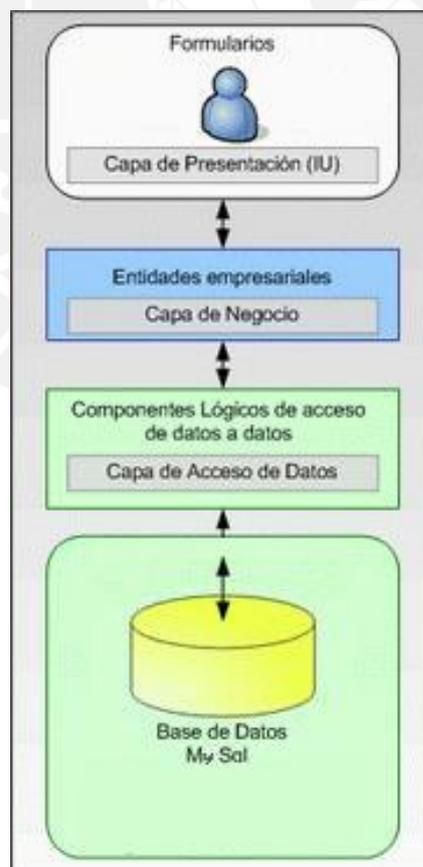


Figura 3.1 Arquitectura de la solución.

Arquitectura de la solución	
Capa	Descripción
Capa de presentación	Representa la capa de Interfaz de Usuario y contiene las clases que permiten al usuario interactuar con el sistema mostrando los elementos GUI (ventanas, botones y controles) con los cuales se accede a las funcionalidades del sistema.
Capa de negocio	Representa la capa lógica del negocio y contiene las clases que permitirán realizar las operaciones del negocio (consultas, procesos y mantenimientos). Estas clases se encuentran en los paquetes de mantenimiento, atención y servicios, y apoyo a la gestión.
Capa de datos	Representa la capa de datos y contiene todas las clases que proveen mecanismos de acceso a la base de datos del sistema. Se encarga de las Consultas y Procesos a la Base de Datos.

Tabla 3.1 Arquitectura de la solución.

3.1.1. Tecnología Requerida

Las estaciones de trabajo deben tener conexión a la red local (LAN) de la que es parte el servidor de base de datos; de ello depende el funcionamiento del sistema, sino no se podrá acceder a la información.

Existirán múltiples instancias del Cliente, las que podrán acceder de manera simultánea a la información de la base de datos.

El sistema debe ser capaz de manejar la demanda simultánea de los usuarios de los módulos de Mantenimiento, Atención de servicios y apoyo a la gestión.

3.1.2. Diagrama de Despliegue

En esta sección del documento se muestra cómo y dónde se desplegará el sistema, por ello la construcción interna estará conformada por 2 nodos: Terminal cliente y Servidor de Base de Datos.

La vista de despliegue que se muestra en la figura 3.2 permite conocer cada uno de los nodos físicos donde se encuentran cada una de las distintas capas de la aplicación.

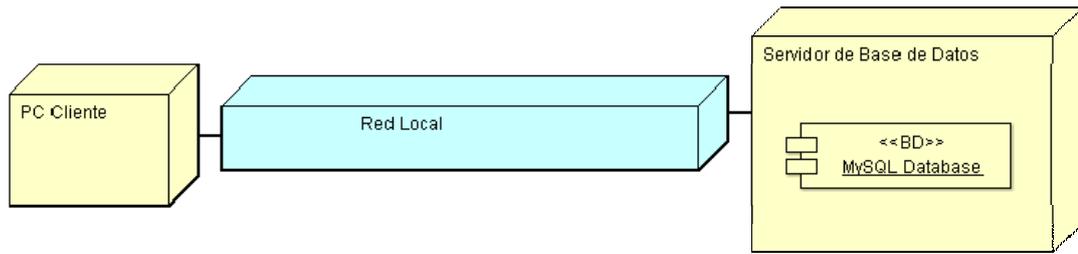


Figura 3.2 Diagrama de Despliegue.
[Fuente Propia]

En la tabla 3.2 se describen brevemente los nodos anteriormente mencionados.

Descripción de los nodos	
Nodo	Descripción
Terminal Cliente	Representa el equipo en el cual se desplegará la interfaz de usuario y los procesos de manejo de la lógica de negocio.
Servidor de Base de Datos	Representa el nodo en el cual se desplegará la administración de la base de datos.

Tabla 3.2 Descripción de los nodos.

3.1.3. Diagrama de Componentes

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas, pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, etc. Las relaciones de dependencia se utilizan en los diagramas de componentes para indicar que un componente utiliza los servicios ofrecidos por otro componente [32].

Un módulo de software se puede representar como componente. Algunos componentes existen en tiempo de compilación, algunos en tiempo de enlace y algunos en tiempo de ejecución.

En la figura 3.3 se visualiza el diagrama de componentes, el cual contiene los componentes que forman parte de la solución del presente proyecto.

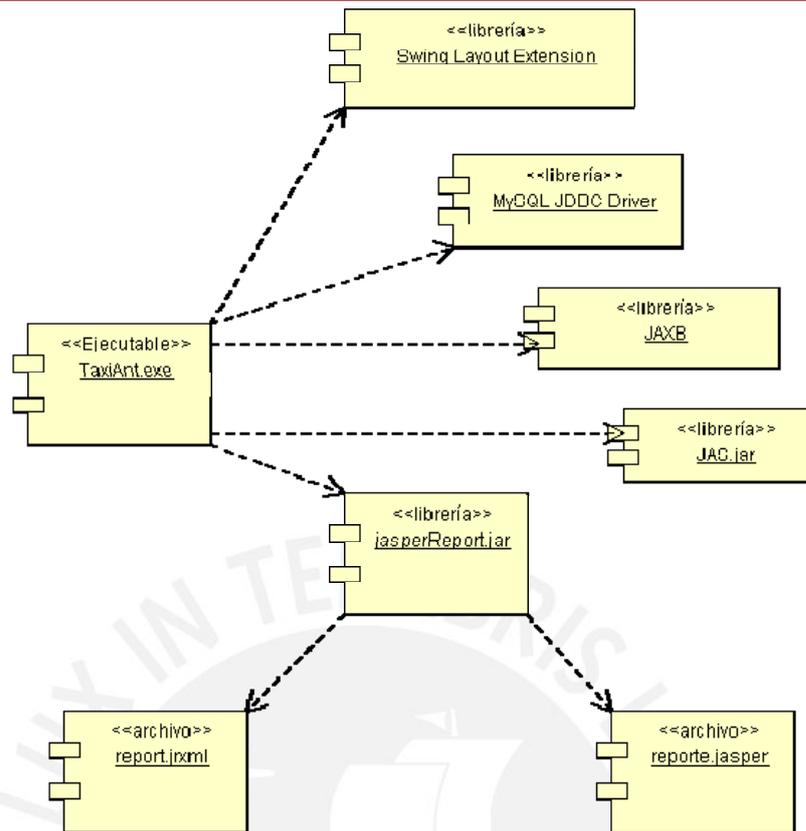


Figura 3.3 Diagrama de Componentes. [Fuente Propia]

3.1.4. Modelo de base de datos

El propósito de este artefacto es mostrar el modelo de base datos que se va a utilizar para apoyar la persistencia de datos del proyecto en los diferentes módulos del proyecto. En este diagrama se definen las tablas que usará el sistema, así como los atributos de cada una y las relaciones con las demás tablas.

Diagrama Entidad Relación

La base de datos presentará dos esquemas distintos debido a la funcionalidad para la que es indispensable cada uno.

El primer esquema es del negocio y contiene las tablas que permitirán el correcto funcionamiento de la empresa.

El segundo esquema es el modelo de rutas y contiene las tablas que permitirán representar tanto el grafo como la zona geográfica presente.

A continuación ambos esquemas se visualizan en las figuras 3.4 y 3.5 respectivamente.

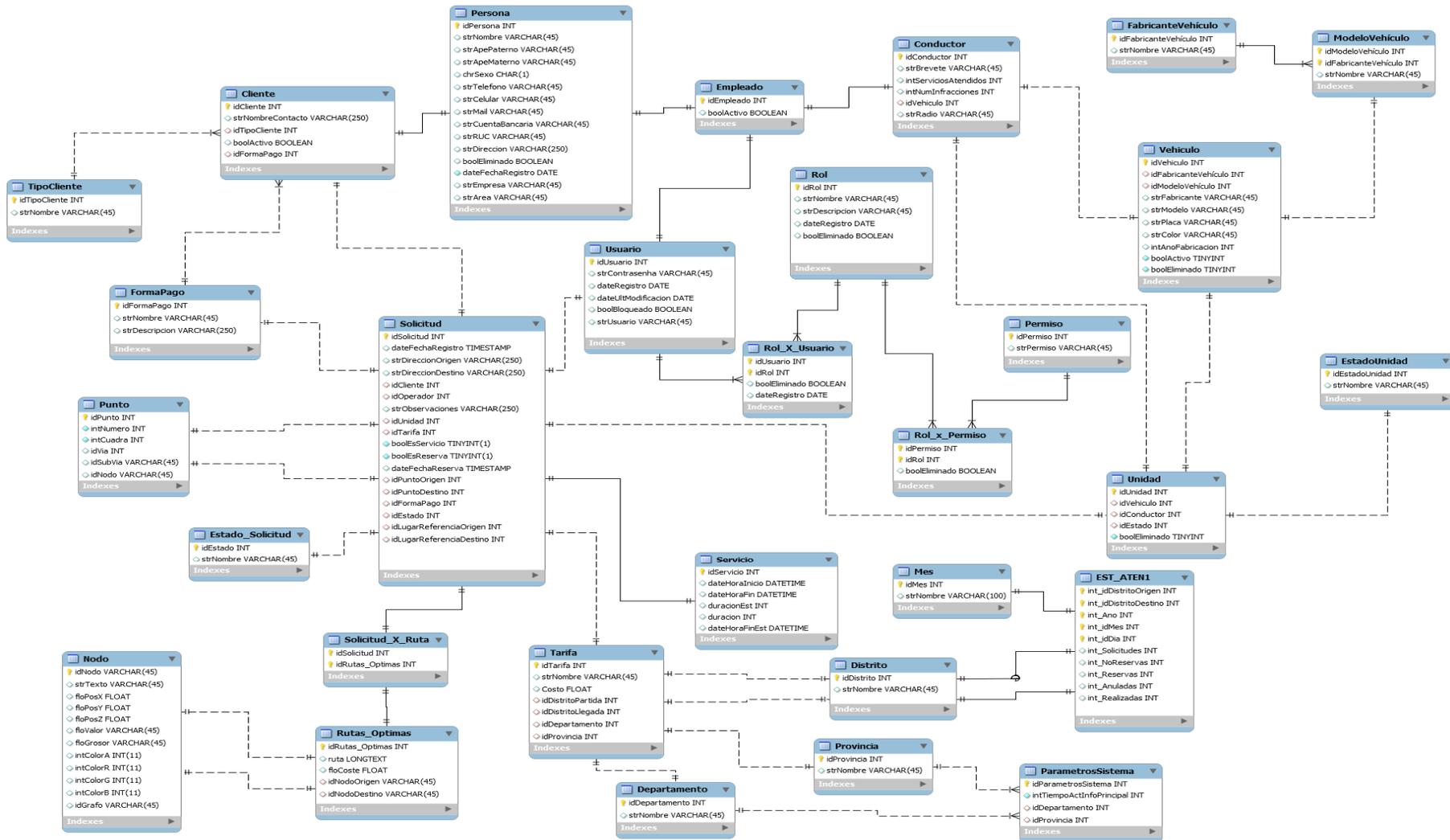


Figura 3.4 Diagrama Entidad Relación: Esquema Negocio.

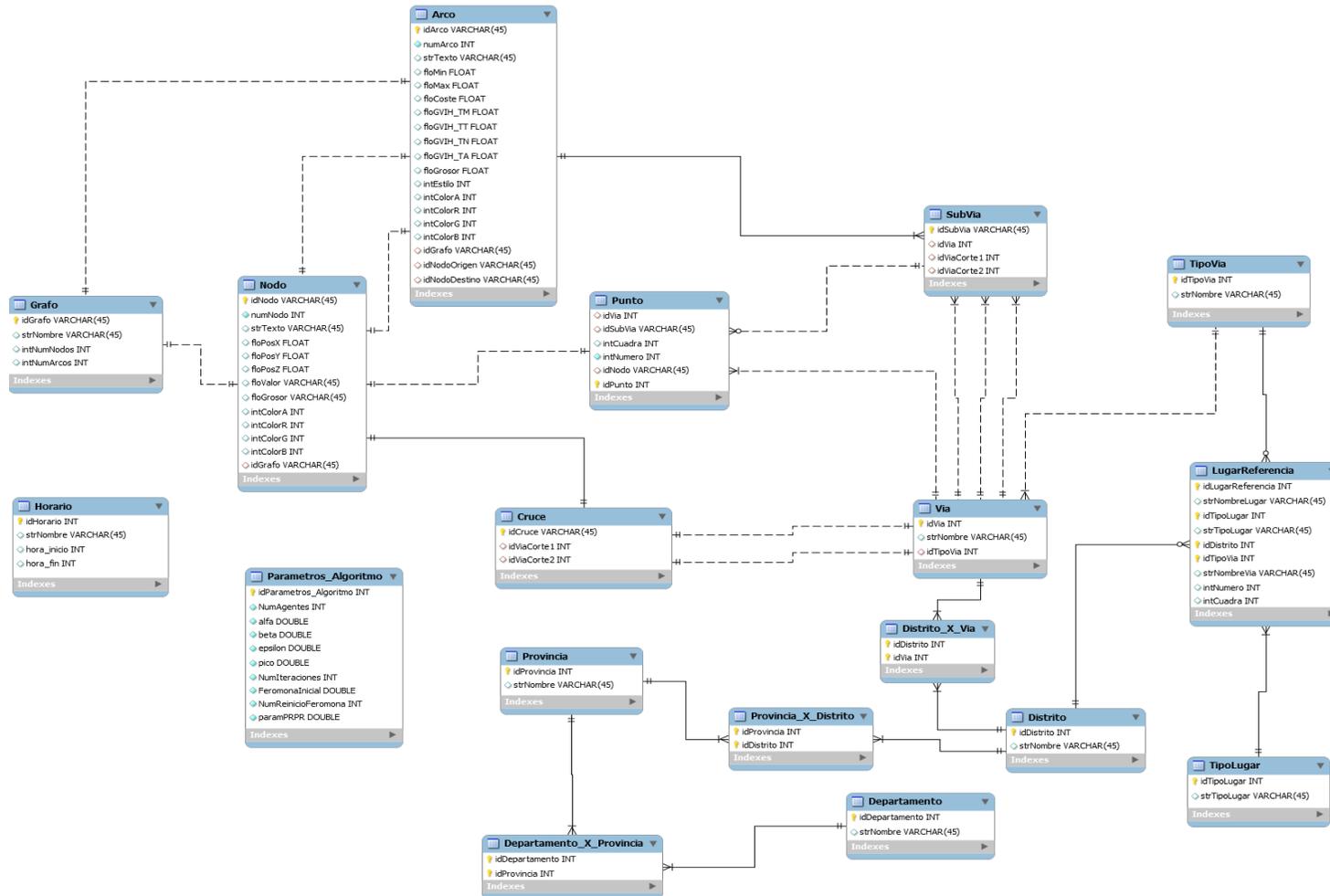


Figura 3.5 Diagrama Entidad Relación: Esquema Grafo.

3.2. Diseño de Interfaz Gráfica

Se van a detallar los criterios para el diseño de la interfaz gráfica de usuario (GUI) tales como: componentes, características y propiedades que tendrán las pantallas para facilitar la interacción con el usuario.

3.2.1. Principios Básicos

La Interfaz Gráfica debe cumplir lo siguiente:

- El sistema es dirigido por el usuario.
- Los usuarios tienen el control del diálogo.
- El sistema se encarga de la asociación de datos-programas.
- Debe ser consistente con los datos.

3.2.2. Sobre el diseño en general

Usabilidad

El uso de la interfaz se realizará a través del teclado y el *mouse*. El *mouse* se utilizará para seleccionar componentes y comandos independientes dentro de la pantalla. El teclado se usará para el ingreso de datos y para seleccionar componentes y comandos de manera ordenada a través de la tecla "Tab", de izquierda a derecha y de arriba abajo. Si se encuentra en el último componente, al presionar la tecla "Tab", el foco regresa al primero.

Presentación

Se refiere al aspecto visual de la interfaz. Los sistemas tienen dos tipos de elementos a presentar:

- Objetos: Cualquier objeto manipulable, son el centro de atención del usuario.
- Acciones: Permiten al usuario crear o manipular objetos.

Entre las consideraciones de presentación estarán:

- Se empleará diferentes colores en la interfaz gráfica para diferenciar los componentes no editables de los editables.

- A cada usuario que acceda al sistema, se le mostrará inicialmente en el cuerpo de la pantalla una imagen distintiva (logo de la organización).
- Todas las ventanas del sistema están compuestas de la siguiente forma: barra de herramientas, cabecera y cuerpo.

Acciones

Se refiere a las opciones que los componentes del sistema ofrecen al usuario. Estas opciones se describirán según los componentes

Navegación

Para la navegación se tiene en cuenta la forma en que se presentan las pantallas del sistema de acuerdo al flujo desde que el usuario ingresa al sistema.

Primero se muestra la ventana de acceso al sistema, una vez que el usuario acceda al sistema, se cerrará esta ventana y se abrirá la ventana principal del sistema donde el usuario podrá ver las opciones que le correspondan según el perfil de usuario al cual pertenece.

La ventana principal contará con un área de trabajo donde se abrirán ventanas internas para cada opción del menú seleccionada.

El usuario realizará mantenimientos, registros, actualización de información, consultas y reportes a través de las ventanas internas.

3.2.3. Diseño de pantallas

A continuación se describen el diseño de algunas pantallas del Sistema con las características antes mencionadas.

Ventana de Autenticación de usuario

La funcionalidad de esta ventana es permitir autenticar al usuario que desee ingresar al sistema, de esta manera se tendrá un control de acceso para la seguridad de las operaciones. En la figura 3.6 se puede visualizar el diseño de esta ventana.



Figura 3.6 Ventana Autenticación de Usuario.

Ventana Principal del sistema

La funcionalidad de esta ventana es permitir al operador tener acceso a las operaciones correspondientes a la atención de algún servicio y mostrar el estado de las solicitudes, así como de las unidades disponibles. En la figura 3.7 se puede visualizar el diseño de esta ventana.

Información Operativa

- * Hay 0 reserva(s) por asignar unidad actualmente
- * Número de unidades disponibles: 2
- * Realizando 3 servicio(s) actualmente

Unidades con Pasajeros

N°	Tipo	Hora Recojo	Hora Fin Est.	Conductor	Dirección Destino	Dirección Origen	Cliente
1	Servicio	14:31 PM	14:51 PM	Jose Carlos Fernandez	Jiron Paracas 310, Cuadra 3, Pueblo Libre	Avenida Universitaria 1810, Cuadra 18, S...	Jose Mori
2	Servicio	14:31 PM	14:46 PM	Mario Lopez	Jiron Paracas 210, Cuadra 2, Pueblo Libre	Avenida Universitaria 1810, Cuadra 18, S...	Luis Lock
3	Servicio	14:31 PM	15:06 PM	Diego Zelada	Jiron Paracas 210, Cuadra 2, Pueblo Libre	Avenida Universitaria 1810, Cuadra 18, S...	Luis Lock
4	Solicitud			Juan Carlos Borja	Avenida Uni 415, Cuadra 4, San Miguel	Avenida Uni 300, San Miguel	Nora La Serna
5	Solicitud			Alberto Aguirre	Avenida UNIVERSITARIA 1910, San Miguel	Avenida Universitaria 1810, San Miguel	Jose Mori

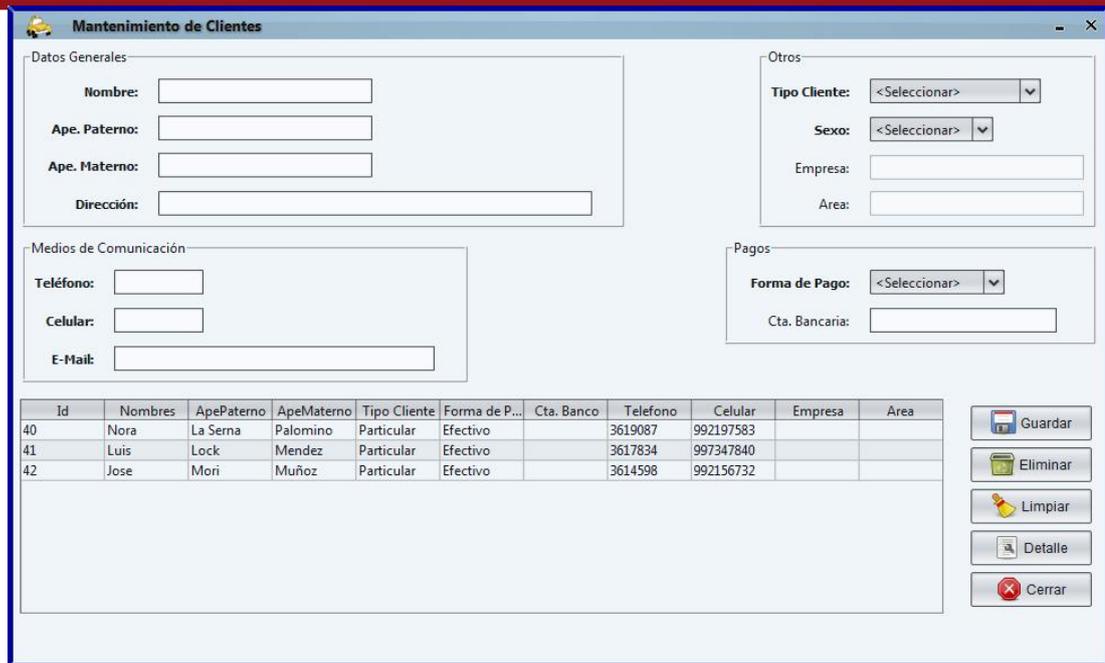
Unidades Disponibles

Disponible	Hora Dis. Est.	Dirección Destino	Vehiculo	Radio	Conductor	Solicitud	Estado
Si			AOH-466	2222*007	Dennis Huaman		
Si			TAK-414	2222*006	Bruno Peña		
No		Avenida UNIVERSITARIA 1910, San Miguel	LOP-891	2222*005	Alberto Aguirre	5	Pendiente
No	15:06 PM	Jiron Paracas 210, Cuadra 2, Pueblo Libre	BAD-114	2222*004	Juan Carlos Borja	4	Pendiente
No	14:46 PM	Jiron Paracas 210, Cuadra 2, Pueblo Libre	GDH-213	2222*003	Diego Zelada	3	En transcurso
No	14:46 PM	Jiron Paracas 210, Cuadra 2, Pueblo Libre	RIP-782	2222*002	Mario Lopez	2	En transcurso
No	14:51 PM	Jiron Paracas 310, Cuadra 3, Pueblo Libre	VIR-001	2222*001	Jose Carlos Fernandez	1	En transcurso

Figura 3.7 Ventana Principal del sistema.

Ventana de Mantenimiento Estándar

La funcionalidad de esta ventana es permitir el registro, actualización y eliminación de empleados, vehículos, clientes, etc. Correspondientes al módulo de Mantenimiento, de esta manera el uso del sistema será simple y rápido para los usuarios. En la figura 3.8 se puede visualizar el diseño de esta ventana.



Id	Nombres	ApePaterno	ApeMaterno	Tipo Cliente	Forma de P...	Cta. Banco	Telefono	Celular	Empresa	Area
40	Nora	La Serna	Palomino	Particular	Efectivo		3619087	992197583		
41	Luis	Lock	Mendez	Particular	Efectivo		3617834	997347840		
42	Jose	Mori	Muñoz	Particular	Efectivo		3614598	992156732		

Figura 3.8 Ventana de Mantenimiento Estándar.

Ventana de Registro de Solicitud

La funcionalidad de esta ventana es permitir el registro de solicitudes de servicio de taxi, con todos los requisitos para poder ingresar una solicitud. Esta ventana corresponde al módulo de Recepción y Asignación de servicios, en la figura 3.9 se puede visualizar su diseño.

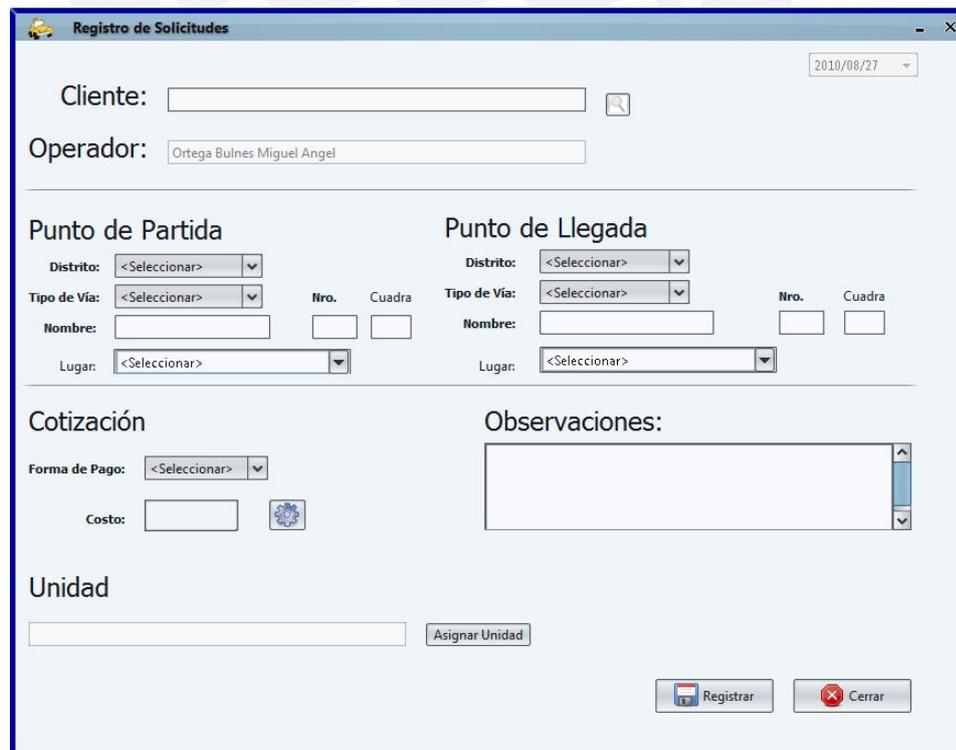


Figura 3.9 Ventana de Registro Estándar.

Ventana de Reporte Informativo de Solicitudes

La funcionalidad de esta ventana es brindar información de los servicios realizados como número de solicitud, fecha de registro, punto de recojo, punto de destino, costo, cliente y conductor, durante un periodo dado. Esta ventana corresponde al módulo de Apoyo a la Gestión y se puede visualizar su diseño en la figura 3.10.

Taxi Ant							Reporte de Solicitudes
							domingo 07 noviembre
SOLICITUD	FECHA DE REGISTRO	PUNTO DE RECOJO	PUNTO DE DESTINO	COSTO (S/.)	CLIENTE	CONDUCTOR	
10	07/11/10 06:36 PM	Avenida Javier Prado 2567, Cuadra 23, San Isidro	Jiron Las Cascadas 656, Cuadra 6, Barranco	24.0	Armando Vera Sarmiento	Jose Carlos Fernandez Rivera	
9	07/11/10 06:35 PM	Avenida La Marina 2367, Cuadra 23, Pueblo Libre	Avenida Lima 1341, Cuadra 13, San Miguel	17.0	Nora La Serna Palomino	Eduardo Gutierrez Villarán	
8	07/11/10 06:31 PM	Avenida Los Militares 467, Cuadra 4, Barranco	Avenida Navarrete 123, Cuadra 1, San Isidro	24.0	Armando Vera Sarmiento	Mario Lopez Albuja	
7	07/11/10 06:29 PM	Avenida Jacarandaes 2341, Cuadra 23, Breña	Jiron Riva Agüero 456, Cuadra 4, San Miguel	14.0	Luis Lock Mendez	Pedro Perez Portugal	
6	07/11/10 06:28 PM	Avenida La Marina 2667, Cuadra 26, San Miguel	Jiron Libertad 245, Cuadra 2, Magdalena	16.0	Nora La Serna Palomino	Alberto Aguirre Carrasco	
5	07/11/10 06:27 PM	Calle Miramar 132, Cuadra 1, San Miguel	Calle Las Magnolias 233, Cuadra 2, Breña	14.0	Salvador Mayolo Bulnes	Bruno Peña Cataño	
4	07/11/10 06:25 PM	Calle Las Begonias 655, Cuadra 6, San Isidro	Jiron Salaverry 366, Cuadra 3, Barranco	24.0	Jose Mori Muñoz	Dennis Huaman Ricasca	
3	03/11/10 07:29 PM	Avenida Universitaria 2000, Cuadra 20, San Miguel	Avenida Ayacucho 1210, Cuadra 12, Pueblo Libre	16.0	Jose Mori Muñoz	Dennis Huaman Ricasca	
2	03/11/10 07:25 PM	Avenida Universitaria 2000, Cuadra 20, San Miguel	Avenida Ayacucho 1210, Cuadra 12, Pueblo Libre	16.0	Luis Lock Mendez	Eduardo Gutierrez Villarán	
1	03/11/10 07:19 PM	Avenida Universitaria 2000, Cuadra 20, San Miguel	Avenida Ayacucho 1210, Cuadra 12, Pueblo Libre	16.0	Nora La Serna Palomino	Pedro Perez Portugal	

Figura 3.10 Ventana de Reporte Informativo de Solicitudes.

Ventana de Reporte Estadístico de Solicitudes

La funcionalidad de esta ventana es generar información específica al momento de tomar decisiones para la correcta gestión operativa. Esta ventana corresponde al módulo de Apoyo a la Gestión y se puede visualizar un ejemplo en las figuras 3.11.

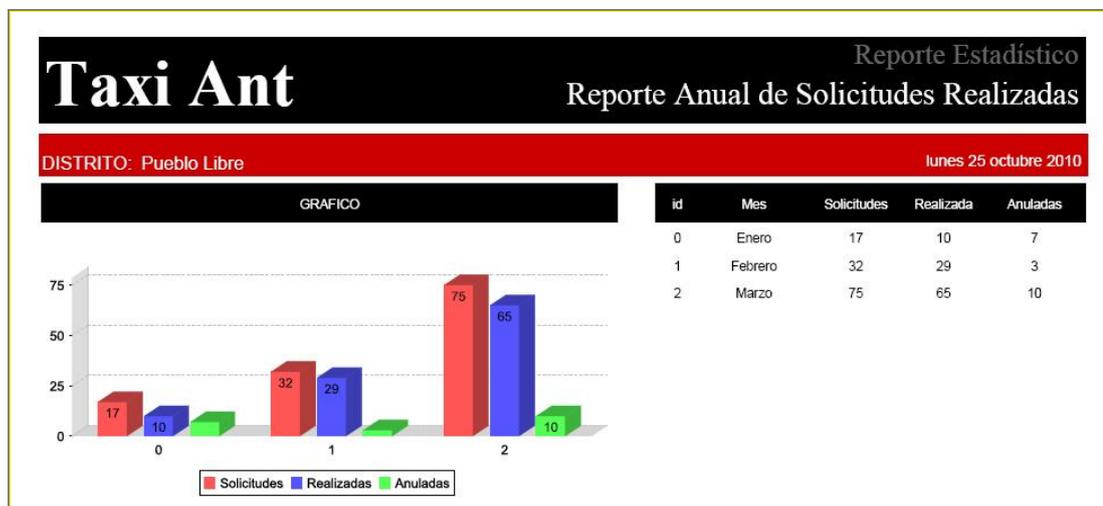


Figura 3.11 Ventana de Reporte Estadístico de Solicitudes.

Capítulo IV: Construcción

El objetivo de este capítulo es describir las decisiones relacionadas a la construcción de la solución, así como la estrategia de pruebas utilizada y el resultado de la ejecución de estas pruebas.

4.1. Construcción

4.1.1. Tecnología usada

Sistema Operativo Windows

Es un sistema operativo desarrollado por Microsoft, disponible para entornos informáticos, incluyendo computadoras domésticas o de negocios.

Para el desarrollo de la presente solución se recurre a este sistema operativo tanto por poseer una interfaz de uso más fácil para el usuario como por integrarse con el entorno de desarrollo NetBeans y el sistema de administración de base de datos MySQL.

Lenguaje Java

La selección de java es debido al alto rendimiento que se consigue con su uso en aplicaciones robustas sobre múltiples plataformas en redes heterogéneas y distribuidas [33]. Es un lenguaje completamente orientado a objetos lo cual facilita el desarrollo del sistema propuesto.

Se recurre al uso de esta tecnología por su facilidad de uso y portabilidad debido a ser de arquitectura neutral, lo que permite que la aplicación resultante sea adaptable tanto a Linux como a Windows. También es usada por la experiencia que se cuenta en su uso, lo cual provee mayor eficiencia en el desarrollo.

Sistema de gestión de base de datos MySQL

El software MySQL® proporciona un servidor de base de datos SQL (Structured Query Language) muy rápido, multiusuario y robusto. El servidor MySQL está diseñado para entornos de producción críticos, con alta carga de trabajo así como para integrarse en software para ser distribuido [34].

Esta tecnología forma parte de la solución por soportar todos los tipos de dato y características necesarios para el desarrollo del presente proyecto. Además, es bastante estable, disminuye los costos de mantenimiento y es de uso gratuito.

Entorno de Desarrollo NetBeans

El entorno de desarrollo integrado (IDE) NetBeans supera los puntos débiles asociados con otras plataformas de desarrollo, y abre nuevas posibilidades para crear en forma rápida y eficientemente aplicaciones multiplataforma [35]. En su núcleo, el IDE NetBeans es una herramienta de desarrollo para Java escrita empleando tecnología Java pura, por lo que se ejecuta en cualquier parte donde se ejecute Java, lo cual es, por supuesto, en casi todas partes.

Para el desarrollo de la presente solución se utiliza NetBeans 6.0 debido a que es un producto de código abierto, hace el desarrollo más eficiente y permite ahorrar cientos de horas de tiempo al realizar la implementación. Además de poder integrarse con elementos críticos requeridos para el sistema como es el caso del componente IReports para la generación de reportes.

A continuación, se describen las tecnologías más representativas utilizadas en el sistema propuesto, para más detalle ver el anexo E.

IReports

Es un software para el diseño de informes visuales, poderoso, intuitivo y fácil de usar que puede ser mostrado en formatos como PDF, OpenOffice, DOCX y muchos otros [36]. Para ello hace uso de JasperReports, la librería de reportes para Java más popular y de código abierto. De esta manera si IReports permite diseñar reportes, JasperReports permite ejecutarlos y generarlos en una aplicación hecha en Java.

Se requiere el uso de esta tecnología para la generación de reportes en formato PDF con información gráfica, que se utilizará en el módulo de apoyo a la gestión para brindar información estadística de las operaciones realizadas.

Software Grafos

Es un software para la construcción, edición y análisis de grafos [37], por tanto pretende ser de utilidad para la docencia y el aprendizaje de la teoría de grafos, y otras disciplinas relacionadas como la ingeniería de organización industrial, la logística y el transporte, investigación operativa, diseño de redes, etc. De esta manera Grafos se puede usar perfectamente para el modelado y resolución de problemas reales.

Esta herramienta es útil por permitir diseñar y construir un grafo, requerido por el algoritmo, a partir de la imagen de un mapa distrital representativo; además, cuenta con la posibilidad de exportar e importar los datos del grafo al formato de fichero estándar GraphML.

GraphML File Format

GraphML es un formato comprensible y fácil de usar para especificar grafos. Consta de un núcleo de lenguaje capaz de describir las propiedades estructurales de un grafo, y un mecanismo flexible de extensiones para agregar información específica para cada aplicación [38]. Es decir, no incluye obligatoriamente las propiedades de dibujo, a diferencia de otros formatos como el SVG (Scalable Vector Graphics), más bien se centra en especificar un grafo únicamente con propiedades útiles para los algoritmos.

Para la carga del grafo con el registro de zonas y calles se hace uso de este formato por ser un fichero de estructura XML y especialmente hecho para el lenguaje Java, por tanto será de utilidad para el desarrollo de la presente solución.

4.1.2. Algoritmo de optimización propuesto

Las razones por las cuales se selecciona la meta heurística OCH como algoritmo de optimización son las siguientes:

- Forma parte de un interés académico del presente, para desarrollar un sistema de información que haga uso de la inteligencia bio-inspirada para resolver el problema de la ruta óptima.
- Forma parte de la familia de algoritmos meta heurísticos reconocidos como promesas para resolver problemas difíciles de optimización combinatoria.
- Es actualmente un campo de investigación maduro.

Se pretende aplicar de la mejor manera el algoritmo de optimización propuesto en este proyecto de fin de carrera, por tal motivo se seguirá 5 pasos para resolver el problema planteado por la meta heurística OCH (Ver sección 1.2.7). Los cuales detallamos a continuación:

Primero: Representar el problema en un Grafo $G_C (V, E)$.

Se tiene un grafo G que consiste en un conjunto de vértices V y arcos E .

Se tiene también un conjunto finito de componentes de la solución.

$$C = \{c_{ij}\}, i=1, \dots, n; j=1, \dots, |D_i|$$

Donde el conjunto de componentes C es asociado a cada conjunto de los vértices V del grafo G_C o al conjunto de arcos E y n es el número de hormigas artificiales.

Lo descrito es válido para el problema planteado puesto que C representará al conjunto de nodos de nuestro grafo, quienes a su vez interpretarán la intersección entre dos vías, incluyendo lugares de referencia.

Por otro lado, se plantea en este trabajo de tesis dos métodos para mejorar la aplicación de la meta heurística OCH debido a su naturaleza de operación.

- A. Acotar el grafo o el espacio de búsqueda de las hormigas del algoritmo OCH, puesto que se ha determinado que la solución óptima siempre es encontrada dentro del dominio ubicado entre el nodo origen y el nodo de destino. La acotación es definida como la siguiente:

*Dado S la superficie rectangular ubicada en el plano XY que contiene al grafo $G (V, E)$, y cuya área A es definida por la ecuación:

$$A = (x_{i+1} - x_i) * (y_{i+1} - y_i)$$

Siendo $N_i (x_i, y_i)$ y $N_{i+1}(x_{i+1}, y_{i+1})$ dos nodos que pertenecen al grafo G de tal manera que $x_i \leq x_{i+1}$, $y_i \leq y_{i+1}$, $x_i \geq 0$, $y_i \geq 0$.

*Dado S' la superficie rectangular ubicada en el plano XY que contiene a otro grafo $G'(V', E')$ cuyos vértices V' pertenecen al conjunto de vértices V y los arcos E' pertenecen al conjunto de arcos E del grafo G ; y cuya área A' es definida por la ecuación:

$$A' = (x_{j+1} - x_j) * (y_{j+1} - y_j)$$

Siendo $N_j (x_j, y_j)$ y $N_{j+1}(x_{j+1}, y_{j+1})$ dos nodos que pertenecen al grafo G' de tal manera que: $x_j \leq x_{j+1}$, $y_j \leq y_{j+1}$, $x_j \geq x_i$, $y_j \geq y_i$, $x_{i+1} \geq x_{j+1}$, $y_{i+1} \geq y_{j+1}$.

De esta manera G es el grafo original que contiene a todos los nodos del problema y G' es el grafo, subconjunto de G , que contiene a los nodos ubicados dentro la región S' , en cuyo límite se encuentran el nodo origen (N_j) y nodo destino (N_{j+1}) de la solución óptima.

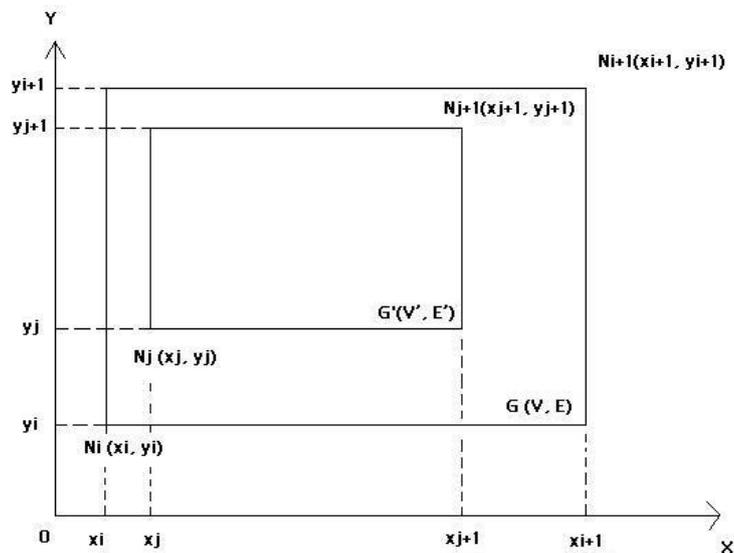


Figura 4.1 Acotación del espacio de búsqueda [Fuente Propia].

La figura 4.1 muestra de manera gráfica la acotación del espacio de búsqueda descrito anteriormente.

Ahora bien, es menester tener en cuenta que en la realidad no siempre todos los nodos de la solución óptima forman parte de la superficie S' que contiene al grafo G' . Es por ello que, para efectos de aplicar de la mejor forma la meta heurística OCH al contexto del problema planteado, se utiliza un grafo $G_r (V_r, E_r)$, cuya área A_r es definida por la ecuación:

$$A_r = (x_{k+1} - x_k) * (y_{k+1} - y_k)$$

Siendo $N_k (x_k, y_k)$ y $N_{k+1}(x_{k+1}, y_{k+1})$ dos nodos que pertenecen al grafo G_r de tal manera que $x_k \leq x_{k+1}$, $y_k \leq y_{k+1}$, $x_k \geq 0$, $y_k \geq 0$.

Además de:

$$x_k = x_j - \Delta;$$

$$x_{k+1} = x_{j+1} + \Delta;$$

$$y_k = y_j - \Delta;$$

$$y_{k+1} = y_{j+1} + \Delta;$$

Siendo $\Delta > 0$ una constante que permitirá incluir nodos aledaños con probabilidad de pertenecer a la solución óptima.

Como resultado se tiene un menor tiempo computacional al hallar la ruta óptima entre dos puntos utilizando la meta heurística OCH.

- B. Mejorar la búsqueda de otras soluciones factibles permitiendo tener diferentes soluciones hasta llegar a encontrar la óptima. Hasta este punto sabemos que una de las principales características de la meta heurística OCH es elegir, en base a una decisión probabilística, el mejor nodo a partir de otro; pero existe una situación que minimiza las probabilidades de obtener la solución óptima, la cual es presentada a continuación:

Dado los nodos n_1 , n_2 y n_3 ; y los arcos $a_1 (n_1, n_2)$ y $a_2 (n_1, n_3)$ con longitudes D_1 y D_2 respectivamente.

Siendo $D_2 = D_1 * q$, donde $D_2 > D_1$, $q > 1$ y q es un número entero.

Se tiene un cuarto nodo n_4 al cual se le ubica en el arco a_2 . De esta manera los nodos correspondientes a los arcos a_1 , a_2 y un nuevo arco a_3 son: $a_1 (n_1, n_2)$, $a_2 (n_1, n_4)$ y $a_3 (n_4, n_3)$ con distancias d_1 , d_2 , y d_3 respectivamente.

Siendo $d_2 + d_3 = D_2$, $d_1 = D_1$ y $d_2 \approx d_1$.

Es recomendable hacer uso de este paso al momento de formar el grafo $G (V, E)$ por la necesidad de encontrar mejores soluciones al momento de aplicar el algoritmo de optimización.

Segundo: *Definir apropiadamente el nivel del rastro de feromona.*

Este paso corresponde a tomar en cuenta el valor de la feromona depositada tanto al inicio como al momento de construir las soluciones del algoritmo. No es una tarea trivial por ende los valores deben ser evaluados al ejecutar varias experimentaciones del algoritmo aplicado al caso.

Tercero: *Definir apropiadamente la preferencia heurística.*

Esta tarea corresponde a definir una correcta y acertada información heurística η_{rs} , que estará asociada a cada componente o transición de un grafo G . En la literatura este valor es determinado por la inversa de la distancia entre dos nodos; la fórmula es la siguiente:

$$\eta_{rs} = \frac{1}{d_{rs}}$$

Siendo d_{rs} la distancia entre los nodos r y s .

El caso ideal es tomar en cuenta esta única variable d_{rs} ; pero al querer resolver un problema del mundo real aplicando la meta heurística OCH, como es el caso de optimizar rutas de transporte para una empresa que brinde servicios de taxi, observamos que esto no sucede así, puesto que en un circuito vial se encuentran diversos factores que guardan relación entre sí e intervienen al momento de encontrar el mejor camino, entre ellos tenemos:

- Nivel de tráfico.
- Horario de viaje.
- Disponibilidad y/o estado de las vías.
- Etc.

Ante esta situación, se plantea en este trabajo de tesis un método empírico, que permitirá contar con una apropiada información heurística en base a muestras reales, de tal manera que el sistema ha desarrollar haga uso de esta información para brindar una solución óptima aplicada al caso.

El método consiste en hallar un valor teórico del tiempo que demora en recorrer un vehículo por una vía correspondiente a un arco $A(r, s)$, para ello hacemos uso de la ecuación:

$$\text{Distancia} = \text{Velocidad} * \text{Tiempo}$$

Desde luego, para poder hallar el tiempo teórico es necesario conocer las dos demás variables (distancia y velocidad). Siendo evidente indicar que son variables ya conocidas, por las siguientes razones:

- La distancia D entre los nodos del arco $A(r, s)$ proviene de un levantamiento de información durante la fase de construcción del grafo G que permita interpretar un plano o mapa con las vías y lugares de referencia.
- La velocidad V es una información obtenida en base a una política de la empresa de servicio de taxis, quien establece una velocidad promedio para que el conductor lleve al cliente a su lugar de destino. Esta velocidad no debe ser ni muy rápida, para evitar accidentes, ni muy lenta, para evitar retrasos.

Como resultado se tiene un tiempo $T_{\text{teórico}}$ que es el tiempo en que se demora un vehículo en recorrer el camino entre los nodos del arco A .

Hasta este punto el tiempo $T_{\text{teórico}}$ no considera los factores que hemos mencionado anteriormente, es por esta razón que se debe de tomar muestras para poder hallar el tiempo real que se demora un vehículo en recorrer la vía correspondiente.

Es decir, se construye una población con N elementos (X_1, X_2, \dots, X_N) , que son los tiempos medidos al recorrer la vía correspondiente al arco A . Siendo μ la media de la población definida por la siguiente fórmula:

$$\mu = \frac{\sum_{i=1}^N X_i}{N}$$

Considerando μ como el tiempo promedio real medido y que está afectado por los niveles de tráfico, etc. Se tiene entonces un tiempo T_{real} , mayor igual a $T_{\text{teórico}}$, equivalente a μ que nos permitirá obtener el grado de variación respecto al tiempo teórico $T_{\text{teórico}}$. Grado que será calculado mediante la siguiente fórmula:

$$\lambda = \frac{(T_{\text{real}} - T_{\text{teórico}})}{T_{\text{teórico}}} * 100 \%$$

Siendo λ el grado de variación de la información heurística, mayor igual a 0, reflejado finalmente en la siguiente ecuación:

$$\eta_{rs} = \frac{1}{d_{rs} * (1 + \lambda)}$$

Donde η_{rs} es la información heurística y d_{rs} la distancia entre los nodos r y s .

Esta última función será considerada por el algoritmo del sistema teniendo en cuenta que se aplicará a aquellas vías cuyo tiempo de recorrido varía significativamente en los turnos mañana, tarde, noche y madrugada.

Cuarto: *Escoger un algoritmo específico de OCH.*

Para propósitos de optimización de las rutas de transporte que forma parte importante para la realización de este sistema se implementará una variante de la meta heurística OCH, llamada Sistema de Colonia de Hormigas (Ant Colony System – ACS).

El proceso de construcción en esta variante se denota de la siguiente manera:

1 **Procedimiento Construcción_Solución_Hormiga ()**

2 Repetir para $k=1$ a m (número de hormigas - agentes)

3 Hormiga_Nueva(k);

4 Fin Repetir

5 Para cada arco $a_{rs} \in S_{\text{global-best}}$ hacer

6 $\tau_{rs} \leftarrow (1 - \rho) \cdot \tau_{rs} + \rho \cdot f(C(S_{\text{global-best}}))$

7 Fin Para

9 **Fin Procedimiento**

1 **Procedimiento Hormiga_Nueva (Id_Hormiga)**

2 $k = \text{Id_Hormiga}$; $r = \text{estado_inicial}$; $S_k = r$;

3 $L_k = r$;

4 Mientras ($\text{estado_actual} \neq \text{estado_objetivo}$)

5 Para cada $s \in \mathcal{N}_k(r)$ hacer $b_{rs} = \tau_{rs} \cdot \eta_{rs}^\beta$

6 $q = \text{Generar_Valor_Aleatorio}(0, 1)$

7 Si ($q \leq q_0$) entonces

8 $\text{estado_Proximo} = \text{Max}(b_{rs}, \mathcal{N}_k(r))$

9 Caso contrario

10 Para cada $s \in \mathcal{N}_k(r)$ hacer $p_{rs}^k = \frac{[\tau_{rs}]^\alpha \cdot [\eta_{rs}]^\beta}{\sum_{u \in \mathcal{N}_k(r)} [\tau_{ru}]^\alpha \cdot [\eta_{ru}]^\beta}$

11 $\text{estado_Proximo} = \text{Aplicar_Decision}(P, \mathcal{N}_k(r))$

12 Fin Si

13 $r = \text{estado_Proximo}$; $S_k = \langle S_k, r \rangle$; $L_k = L_k \cup r$

14 $\tau_{rs} \leftarrow (1 - \varphi) \cdot \tau_{rs} + \varphi \cdot \tau_0$

15 Fin Mientras

16 $S_{\text{global-best}} = \text{Mejor_Solucion}(S_{\text{global-best}}, S_k)$

17 **Fin Procedimiento**

Siendo b_{rs} la variable que mide el efecto de la variable heurística y probabilística en el arco a_{rs} , S_k la solución local y L_k la solución guardada en memoria de la hormiga k . Además $S_{\text{global-best}}$ la mejor solución conseguida por los agentes.

Quinto: *Ajustar los parámetros del algoritmo OCH hasta obtener buenas soluciones.*

Por último, considerado también un paso importante, obtener los valores adecuados de los parámetros del algoritmo, de tal manera que se pueda tener como resultado buenas soluciones.

Los parámetros a evaluar son los siguientes:

- Primer coeficiente de evaporación (ρ).
- Segundo coeficiente de evaporación (φ).
- Valor inicial de la feromona (τ_0).
- Grado de variación de la información heurística (λ) (Nuevo).
- Número de iteraciones.
- Número de agentes (m).
- Importancia relativa de la información heurística o visibilidad (β).
- Importancia relativa de la traza de feromona (α).
- Número de iteraciones para reiniciar el rastro de feromona.
- Valor aleatorio para la regla proporcional (q_0).

Para realizar el procedimiento de ajustar los parámetros nos basaremos en un enfoque empírico, el cual se basa en principios tomados de la literatura para los parámetros del algoritmo Sistema Colonia de Hormigas, propuesto en este trabajo de tesis.

El enfoque empírico será elaborado mediante una experimentación numérica que comprenderá 15 casos o instancias para resolver, es decir, obtener la trayectoria óptima, ver anexo D.

El proceso de afinar los parámetros del algoritmo fue realizado en trabajos para problemas específicos y sus resultados serán la base para empezar la experimentación; a continuación se presenta los trabajos desarrollados:

- **“Optimización de Colonia de Hormigas”. [27]**

En este trabajo se da solución al problema del agente viajero utilizando el algoritmo Sistema Colonia de Hormigas.

Resultados:

- Se determina α igual a 1 y β igual a 2.
- Se determina Q_0 igual a 0.9.
- Se determina ρ y φ igual a 0.1

Conclusiones:

- Predomina la información heurística (β) sobre la feromona (α).
- Predomina la explotación del conocimiento disponible sobre la exploración controlada.

- **“Un algoritmo Sistema Colonia de Hormigas mejorado para el problema de ruteo de vehículos”. [45]**

En este trabajo se da solución al problema del ruteo de vehículos.

Resultados:

- Se determina α igual a 1 y β igual a 4.
- Se determina Q_0 igual a 0.5.
- Se determina ρ igual a 0.5 y φ igual a 0.1.

Conclusiones:

- Predomina la información heurística (β) sobre la feromona (α).
- Predomina un balance entre la explotación del conocimiento disponible y la exploración controlada.

- **“Resolviendo un único problema del camino más corto usando optimización colonia de hormigas”. [44]**

En este trabajo se da solución al problema del camino más corto utilizando un algoritmo de optimización colonia de hormigas.

Resultados:

- Se determina α , β igual a $\{(1, 1); (1, 2); (1, 3); (1, 5); (2, 4)\}$.

Conclusiones:

- Predomina la visibilidad (β) sobre la feromona (α).
- Se asocia una unidad de feromona a cada arco del grafo.

4.2. Pruebas

Para el sistema de gestión de servicios de taxis las pruebas llevadas a cabo representarán un conjunto de condiciones que evalúen las características, la capacidad y coherencia de los resultados propios del sistema, con la finalidad de determinar si el requisito de la aplicación es parcial o completamente satisfecho y de esta manera descubrir defectos, errores y evaluar los cambios que se le puede aplicar al sistema hasta que funcione correctamente.

4.2.1. Pruebas de aceptación

Durante el desarrollo del proyecto se llevará a cabo la partición equivalente como un método de prueba de caja negra, derivándose en casos de pruebas, quienes comprobarán que cada parte del sistema cumple con lo que se propone en el documento de Especificación de Requisitos del Software [12].

Para realizar el diseño de casos de prueba para partición equivalente se procede en dos pasos [13]:

1. Se identifican las clases de equivalencia, tomando cada condición de entrada. Se aprecia dos tipos: clases de equivalencia válidas, quienes representan entradas válidas al programa, y las clases de equivalencia inválidas, quienes representan estados erróneos.
2. Se definen los casos de prueba, en donde se asigna un número único a cada clase de equivalencia. Hasta que los casos de pruebas hayan cubierto las clases de equivalencia válidas e inválidas.

Por otro lado, para llevar a cabo las pruebas del algoritmo de optimización se realizará un método de prueba de caja blanca, de esta manera se encontrará los mejores parámetros para las soluciones.

4.2.2. Casos de pruebas

El contenido de las clases de equivalencia identificadas y los casos de pruebas se visualizan en el anexo C y será clasificada de acuerdo a los módulos del sistema.

A continuación, en la tabla 4.1, se muestra un ejemplo para la funcionalidad Actualizar Contraseña del módulo de Seguridad.

Campos\Clases	Clases Válidas	Clases No Válidas
Contraseña Actual	1<=Cant. caracteres<=6 A	Cant. caracteres <1 o Cant. caracteres >6 B
		Vacío C
Contraseña Nueva	1<=Cant. caracteres<=6 D	Cant. caracteres <1 o Cant. caracteres >6 E
		Vacío F
Repita Contraseña	1<=Cant. caracteres<=6 G	Cant. caracteres <1 o Cant. caracteres >6 H
		Vacío I

Tabla 4.1 Clases equivalentes – Actualizar Contraseña.

Por otro lado, las pruebas del algoritmo de optimización realizadas durante una experimentación serán realizadas en base a las siguientes variables y serán mejor mostradas en el anexo D:

- Número de corrida, número de Iteraciones, número de Agentes.
- Número de iteraciones para reiniciar el rastro de feromona.
- Valor probabilístico, valor Heurístico.
- Primer y segundo coeficiente de evaporación.
- Valor aleatorio para la regla proporcional.
- Feromona Inicial.
- Distancia resultado de la corrida, distancia mínima a optimizar.
- Variación positiva.
- Nodo de Origen, nodo de Destino.
- Tamaño de la muestra.

4.2.3. Catálogo de pruebas

Los resultados de las pruebas realizadas en base a los casos de pruebas anteriormente identificados se visualizan detalladamente en el anexo C.

A continuación, en la tabla 4.2, se muestra el resultado de los casos de pruebas para la funcionalidad Actualizar Contraseña.

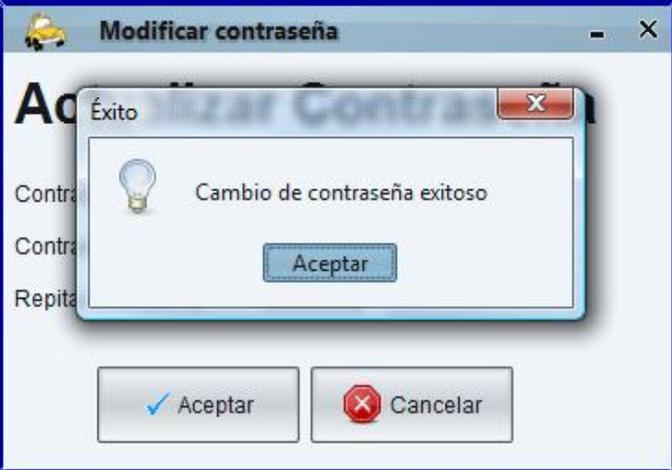
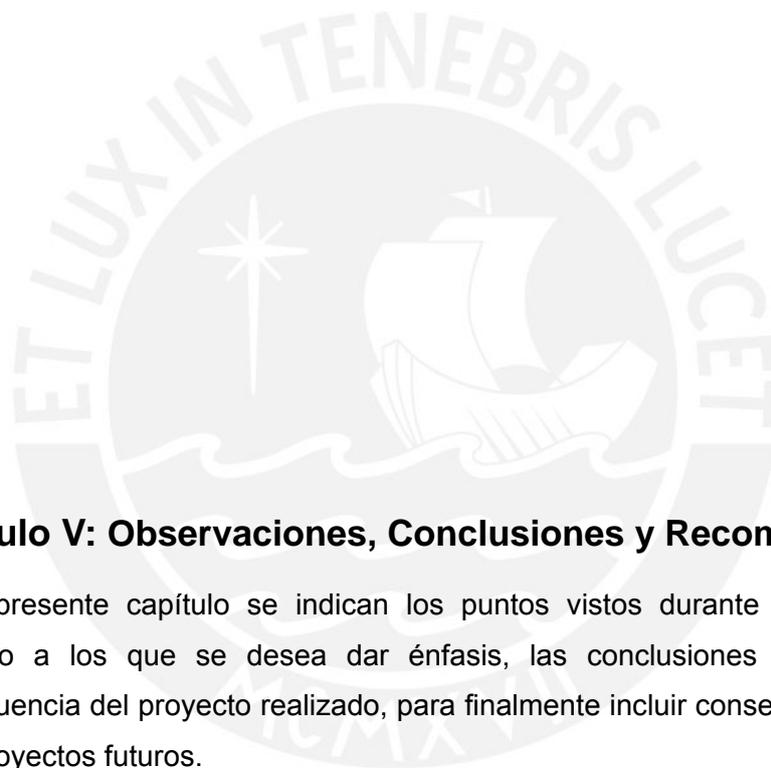
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ADG	123456, 654321, 654321	Mensaje: "Cambio de contraseña exitoso"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Contraseña" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Aceptar. 	Se muestra el mensaje de éxito esperado.
BDG	1234567, 654321,654321	Mensaje: "Contraseña Actual Incorrecta"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Contraseña" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Aceptar. 	Se muestra el mensaje de error esperado.

Tabla 4.2 Casos de Pruebas – Actualizar Contraseña.



Capítulo V: Observaciones, Conclusiones y Recomendaciones

En el presente capítulo se indican los puntos vistos durante el desarrollo del proyecto a los que se desea dar énfasis, las conclusiones deducidas como consecuencia del proyecto realizado, para finalmente incluir consejos y sugerencias para proyectos futuros.

5.1. Observaciones

- El presente proyecto de fin de carrera ha abordado el desarrollo de un sistema de información que dé servicio a los principales procesos para brindar un servicio de taxi; siendo una necesidad reducir el tiempo que se demora en ofrecer el servicio y el costo de gasolina para las unidades de taxi.
- En el presente trabajo, además, se ha considerado el plano de una parte de una zona importante de Lima Metropolitana, como los distritos de San Miguel y Pueblo Libre, de tal manera que se pueda llevar a cabo la solución.

- Para el desarrollo de la solución se ha utilizado software libre como principio de investigación y se ha tomado como base la metodología RUP aplicada al proyecto.
- Dentro de la investigación el proceso de levantamiento de información se realizó gracias al apoyo de una empresa que brinda servicio de taxis en Lima, para poseer credibilidad en el proyecto.
- Ha sido necesario un estudio y selección de una variante de la meta heurística Optimización Colonia de Hormigas que sea apropiada para el problema planteado.

5.2. Conclusiones

- Se ha realizado satisfactoriamente el análisis, diseño e implementación del sistema propuesto en el proyecto de tesis.
- Se ha elaborado apropiadamente un modelo de representación de zonas y calles para llevar a cabo la optimización de las rutas de transporte.
- Se ha implementado eficazmente el algoritmo Sistema de Colonia de Hormigas para mejorar la trayectoria del servicio de taxi en cuanto a distancia y tiempo.
- Se ha demostrado que es factible utilizar las técnicas de la Inteligencia Artificial en la aplicación del problema planteado, el cual es un problema real.
- Se ha utilizado en gran parte los conocimientos y experiencia adquiridos a lo largo de la carrera de ingeniería informática.

5.3. Recomendaciones

- Se recomienda que en futuras investigaciones se analice la posibilidad de incorporar otros módulos de gestión como, por ejemplo, contabilidad.
- También se recomienda una investigación sobre el empleo de otros algoritmos de optimización para observar si se mejora o reduce el tiempo de la trayectoria obtenido con el algoritmo Sistema Colonia de Hormigas.
- Además se aconseja realizar una interfaz gráfica con los resultados de las rutas óptimas para su mejor visualización.
- Por último, se recomienda analizar la viabilidad de tener un sistema de posicionamiento global (GPS), para ello se debe analizar los costos en los que se incurriría, debido a que este tipo de empresas cuenta con recursos muy limitados para ser invertidos en tecnologías de información.

Bibliografía

Libros

- [1] I. Sommerville, "Ingeniería de software", 7ma edición. Pearson Educación. Madrid. 2005. Pags.250-260.
- [2] Dorigo M., Krzysztof Socha. "An Introduction to Ant Colony Optimization". IRIDIA, Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle. Université Libre de Bruxelles. 2006.
- [3] M. Dorigo, V. Maniezzo, and A.Coloni. "The Ant System: Optimization by a colony of cooperating agents". IEEE Transactions on Systems, Man, and Cybernetics – Part B. 1996. Pags.29-41.
- [4] Oscar Córdón, Francisco Herrera, Thomas Stutzle. "A Review on the Ant Colony Optimization Metaheuristic: Basis, Models and New Trends". Mathware & Soft Computing 9. 2002.
- [5] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. "Introduction to Algorithms: Dijkstra's algorithm", Second Edition. MIT Press and McGraw-Hill. Section 24.3. 2001. Pags.595–601.
- [6] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. "Introduction to Algorithms: The Bellman-Ford algorithm", Second Edition. MIT Press and McGraw-Hill. Section 24.1. 2001. Pags.588–592.
- [7] Thomas H. Cormen, Charles E. Leiserson; Ronald L. Rivest, and Clifford Stein. "Introduction to Algorithms: The Floyd–Warshall algorithm", First Edition, MIT Press y McGraw-Hill. Section 26.2. 2001. Pags.558–565.
- [8] M. G. C. Resende and C. C. Ribeiro. "Greedy randomized adaptive search procedures, in Handbook of Metaheuristics". F. Glover and G. Kochenberger, eds., Kluwer Academic Publishers. 2002. Pags.219-249.
- [9] Laporte, G., "The vehicle routing problem: an overview of exact and approximate algorithms". European Journal of Operational Research. 1992. Pags.345-358.
- [10] *PMBOK®* de PMI® 4ta. Edición. 2008.
- [11] JACOBSON, Ivar; BOOCH, Grady; RUMBAUGH, James. "El Proceso Unificado de Desarrollo de Software". Pearson Addison-Wesley. 2000.
- [12] MacGrawHill. "Ingeniería del software, un enfoque práctico", 5ta Edición. 2002. Pags.281-322.
- [13] Wiley. "The art of software Testing". 2da Edición, 2004. Pag.234.
- [14] Gómez Vieites, Álvaro y Suarez Rey, Carlos. "Sistemas de Información: Herramientas prácticas para la gestión empresarial". Ed. Alfaomega. 2da Edición. 2006.
- [15] M.L. Fisher. "Lagrangian Optimization Algorithms for Vehicle Routing Problems," Operational Research. G.K. Rand, ed. 1988. Pags. 635-649.
- [16] Papadimitriou, Christos H."Combinatorial Optimization". Prentice Hall. 1982.
- [17] Hillier F, Lieberman. "Introducción a la investigación de operaciones". McGraw-Hill. México. 1997.
- [18] Ibrahim H. Osman, James P. Kelly. "Meta-Heuristics: Theory and Applications". Kluwer Academic Publishers. 1996.
- [19] F. Glover and G. Kochenberger. "Handbook of Metaheuristics". Kluwer Academic Publishers. 2002.

- [20] Fred Glover, Manuel Laguna. "Tabu Search". Kluwer Academic Publisher. 1997.
- [21] C.A. Floudas and P.M. Pardalos. "Encyclopedia of Optimization". Kluwer Academic Publishers. 2001.
- [22] Marco Dorigo and Thomas Stutzle. "Ant Colony Optimization: Overview and Recent Advances". IRIDIA – Technical Report Series. No. TR/IRIDIA/2009-013. 2009. Pag.14.
- [23] Juan R. Rabunal and Julian Dorado. "Artificial neural networks in real-life applications". Idea Group Publishing. 2006.
- [24] Frank Neumann and Carsten Witt. "Bioinspired Computation in Combinatorial Optimization: Algorithms and their Computational Complexity". Natural Computing Series. Springer. 1998.
- [25] Sriram Pemmaraju, Steven Skiena. "Computational Discrete Mathematics: Combinatorics and Graph Theory with Mathematica". Cambridge University Press. 2003. Pag. 323.
- [26] Gallart Suárez, Joseph. "Análisis, diseño e implementación de un algoritmo meta heurístico GRASP que permita resolver el problema de rutas de vehículos con capacidad". PUCP. Hemeroteca FACI TIN 2 0194. 2009.
- [27] Dorigo M., and Stutzle T. "Ant Colony Optimization". The MIT Press, Massachusetts USA. 1994.

Referencias de fuentes electrónicas

- [28] Registro Mercantil de Santander, Tomo 723, Folio 165, Sección 8ª, Hoja S11607, Inscripción 1ª Editorial Cantabria Interactiva S.L. CIF: B39495460. Disponible en: http://www.eldiariomontanes.es/prensa/20061222/santander/taxistas-informatizan-para-servicio_20061222.html
Consultado en Octubre del 2008.
- [29] Editorial Leoncio Rodríguez, S.A. (Avda. Buenos Aires, 71 S/C de Tenerife. 38005. España.) CIF: A38017844. Disponible en: <http://www.eldia.es/2008-09-10/palma/palma5.htm>
Consultado en Octubre del 2008.
- [30] IAPUCP, "Desarrollo de un sistema Inteligente de Búsqueda de Rutas para el Recojo de Basura en Municipios". Disponible en: http://iapucp.com/index.php?option=com_docman&task=cat_view&gid=24&Itemid=48
Consultado en Octubre del 2008.
- [31] Universidad Politécnica de Valencia. Rational Unified Process (RUP). Disponible en: <https://pid.dsic.upv.es/C1/Material/Documentos%20Disponibles/Introducción%20a%20RUP.doc>
Consultado en Octubre del 2008.
- [32] Practical UML: A Hands-On Introduction for Developers Code Gear from Borland. Disponible en: <http://dn.codegear.com/article/31863#component-and-deployment-diagrams>
Consultado en Octubre del 2008.
- [33] The Java Language Environment White Paper Sun Microsystems. Disponible en: <http://java.sun.com/docs/white/langenv/Intro.doc2.html#334>
Consultado en Octubre del 2008.
- [34] MySQL 5.1 Reference Manual. 1997-2008 MySQL AB. Disponible en: <http://dev.mysql.com/doc/refman/5.1/en/>
Consultado en Setiembre del 2009.

- [35] Conozca el nuevo NetBeans. Desarrollo de aplicaciones multiplataforma con NetBeans IDE. Disponible en:
http://www.sun.com/emrkt/innercircle/newsletter/spain/0207spain_feature.html
Consultado en Setiembre del 2009.
- [36] iReport. Getting Started. What is iReport? Disponible en:
http://jasperforge.org//website/ireportwebsite/IR%20Website/ir_getting_started.html?header=project&target=ireport
Consultado en Octubre del 2010.
- [37] Grafos. ¿Qué es Grafos? Disponible en:
<http://personales.upv.es/arodrigu/grafos/index.htm>
Consultado en Octubre del 2010.
- [38] The GraphML File Format. What is GraphML? Disponible en:
<http://graphml.graphdrawing.org/>
Consultado en Octubre del 2010.
- [39] Algoritmos genéticos y computación evolutiva. Disponible en:
<http://www.the-geek.org/docs/algen/algen.html#key-46>
Consultado en Noviembre del 2010.
- [40] Sachit Abeysundara, Baladasan Giritharan, Saluka Kodithuwakku. "A Genetic Algorithm Approach to Solve the Shortest Path Problem for Road Maps". International Conference on Information and Automation. 2005.
<http://www.ent.mrt.ac.lk/iml/ICIA2005/Papers/SL008CRC.pdf>
Consultado en Noviembre del 2010.
- [41] Mohemmed, A.W. Sahoo, N.C. Tam Kim Geok. Paper "Particle Swarm optimization with noising metaheuristics for solving network shortest path problem". IEEE International Conference. 2007.
http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4448659
Consultado en Noviembre del 2010.
- [42] Sudha Rani Kolavali and Shalabh Bhatnagar. "Ant Colony Optimization Algorithms for Shortest Path Problem". LNCS. Volume 5425. 2009.
<http://www.springerlink.com/content/u150213j16757681/>
Consultado en Noviembre del 2010.
- [43] Yong Jiang, Wan-Liang Wang, Yan-Wei Zhao. "Solving the shortest path problem in vehicle navigation system by ant colony algorithm". WSEAS International Conference. 2007.
<http://portal.acm.org/citation.cfm?id=1364624>
Consultado en Noviembre del 2010.
- [44] Daniel Angus. "Solving a unique Shortest Path problem using Ant Colony Optimization".
<http://elec.uq.edu.au/~uqdangus/Ant%20Colony%20Optimisation%20applied%20to%200a%203D%20shortest%20path%20problem.pdf>
Consultado en Noviembre del 2010.
- [45] Chia-Ho Chen. "An improved ant colony system algorithm for the vehicle routing problem". Journal of the Chinese Institute of Industrial Engineers, Vol. 23, No. 2. 2006. pp.115-126
http://front.cc.nctu.edu.tw/Richfiles/16456-23_2_3.pdf
Consultado en Noviembre del 2010.

Anexos

Anexo A: Especificación de requisitos del software.

Anexo B: Diagrama de clases.

Anexo C: Catálogo de pruebas.

Anexo D: Resultados de la experimentación del algoritmo.

Anexo E: Tecnologías utilizadas.



ANEXO A: Especificación de requisitos

1. Requisitos específicos

Esta sección presenta detalladamente las funcionalidades y restricciones del sistema. El detalle de la funcionalidad se determina con la especificación de los casos de uso.

1.1. Especificación de casos de uso

1.1.1. Módulo de Seguridad

<i>Actualizar Contraseña</i>	
Descripción	El propósito de este caso de uso es modificar la contraseña del usuario del sistema.
Actor	Usuario
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Actualizar Contraseña	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Contraseña. 2. El sistema muestra tres cajas de texto con la contraseña actual, la nueva y otra más para confirmar la nueva contraseña. 3. El usuario ingresa los valores correspondientes en las cajas de texto. 4. El usuario selecciona la opción Aceptar. 5. El sistema verifica que todo haya sido correctamente ingresado y actualiza la contraseña del usuario. 6. Los pasos 2-5 se repiten cada vez que se desee actualizar una contraseña. 7. El caso de uso finaliza. 	
Postcondición	La contraseña del usuario ha sido modificada.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.1 Actualizar Contraseña

<i>Mantener Roles</i>	
Descripción	El propósito de este caso de uso es administrar los roles que determinarán los permisos al sistema.
Actor	Administrador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Registrar Rol	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Roles. 2. El sistema muestra las siguientes cajas de texto: Rol y Descripción. 3. El sistema muestra una lista de permisos existentes y otra lista de permisos asignados. 4. El usuario ingresa los datos del nuevo rol y selecciona los permisos que serán asignados. 5. El usuario selecciona la opción Guardar. 6. El sistema registra el nuevo rol junto con los permisos asignados y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee ingresar un nuevo rol. 8. El caso de uso finaliza.
Postcondición	El nuevo rol queda registrado en el sistema.
Flujo alternativo: Modificar Rol	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Roles. 2. El sistema muestra una lista con todos los roles registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario selecciona uno de los roles de la lista. 4. El sistema muestra en pantalla los datos del rol y sus permisos asignados. 5. El usuario modifica los campos y permisos convenientes y selecciona Guardar. 6. El sistema modifica los datos del rol y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee modificar un rol. 8. El caso de uso finaliza.
Flujo alternativo: Eliminar Rol	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Roles. 2. El sistema muestra una lista con todos los roles registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario escoge uno de los roles de la lista y selecciona Eliminar. 4. El sistema muestra un mensaje de verificación. 5. El usuario acepta la eliminación. 6. El sistema elimina al rol y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee eliminar un rol. 8. El caso de uso finaliza.
Puntos de Excepción	
	<ul style="list-style-type: none"> • No existe punto de excepción.

Tabla A.2 Mantener Roles

<i>Mantener Usuarios</i>	
Descripción	El propósito de este caso de uso es administrar los usuarios del sistema permitiendo asignarles roles del sistema.
Actor	Administrador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Registrar Usuario	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Usuarios. 2. El sistema muestra las siguientes cajas de texto: Usuario, Nombres, Apellido Paterno y Apellido Materno. 3. El usuario ingresa los datos del nuevo usuario y selecciona los roles que serán asignados. 4. El usuario selecciona la opción Guardar. 5. El sistema registra al nuevo usuario junto con los roles asignados y muestra un mensaje de conformidad. 6. Los pasos 2-5 se repiten cada vez que se desee ingresar un nuevo usuario. 7. El caso de uso finaliza.
Postcondición	El nuevo usuario queda registrado en el sistema.
Flujo alternativo: Modificar Usuario	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Usuarios. 2. El sistema muestra una lista con todos los usuarios registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario selecciona uno de los usuarios de la lista. 4. El sistema muestra en pantalla los datos del usuario y sus roles asignados. 5. El usuario modifica los campos y roles convenientes y selecciona Guardar. 6. El sistema modifica los datos del usuario y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee modificar un usuario. 8. El caso de uso finaliza.
Flujo alternativo: Eliminar Usuario	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Usuario. 2. El sistema muestra una lista con todos los usuarios registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario escoge uno de los usuarios de la lista y selecciona Eliminar. 4. El sistema muestra un mensaje de verificación. 5. El usuario acepta la eliminación. 6. El sistema elimina al usuario y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee eliminar un usuario. 8. El caso de uso finaliza.
Puntos de Excepción	
	<ul style="list-style-type: none"> • No existe punto de excepción.

Tabla A.3 Mantener Usuarios

1.1.2. Módulo de Mantenimiento

<i>Configurar Parámetros del Sistema</i>	
Descripción	El propósito de este caso de uso es la configuración de parámetros generales del sistema relacionados a las actividades operativas.
Actor	Administrador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Configurar parámetros del sistema	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Parámetros -> Sistema. 2. El sistema muestra una caja de texto con el tiempo para refrescar la información en la ventana principal y una lista para escoger el departamento y provincia para la cobertura de operaciones. 3. El usuario ingresa el campo mencionado, elige la cobertura y selecciona la opción Guardar. 4. El sistema verifica que todos los campos ingresados estén correctos. 5. El sistema registra los valores y muestra un mensaje de confirmación. 6. Los pasos 2-5 se repiten cada vez que se desee cambiar los parámetros del sistema. 7. El caso de uso finaliza. 	
Postcondición	Los parámetros del sistema son configurados en el sistema.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.4 Configurar Parámetros del Sistema

<i>Configurar Parámetros de Optimización</i>	
Descripción	El propósito de este caso de uso es la configuración de los parámetros para el algoritmo de optimización.
Actor	Administrador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Configurar parámetros de optimización	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Parámetros -> Algoritmo. 2. El sistema muestra un formulario con los siguientes campos: un intervalo que limita la cantidad de feromona depositada, la cantidad de agentes, los pesos de las variables tanto probabilística como heurística, el porcentaje que se disminuirá la feromona en el grafo debido a la actualización y el número de pasos a iterar del algoritmo ACO. Los campos estarán vacíos si es la primera vez que se efectúa este caso de uso. 3. El usuario ingresa todos los campos del formulario y selecciona la opción Guardar. 4. El sistema verifica que todos los campos ingresados estén correctos. 5. El sistema registra los valores y muestra un mensaje de confirmación. 6. Los pasos 2-5 se repiten cada vez que se desee cambiar los parámetros de optimización. 7. El caso de uso finaliza cuando el usuario selecciona cancelar o cerrar. 	
Postcondición	Los parámetros de optimización son actualizados en el sistema.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.5 Configurar Parámetros de Optimización

<i>Mantener Clientes</i>	
Descripción	El propósito de este caso de uso es de administrar los clientes a quienes se les brinda el servicio
Actor	Operador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Registrar Cliente	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Clientes. 2. El sistema muestra las siguientes cajas de texto: Nombre, Apellido Paterno, Apellido Materno, Dirección, Teléfono, Celular, E-Mail y Cuenta Bancaria, además de un listado de Tipo de cliente, Sexo y Forma de pago. 3. El usuario ingresa todos los datos del nuevo cliente. 4. El usuario selecciona la opción Guardar. 5. El sistema registra el nuevo cliente y muestra un mensaje de conformidad. 6. Los pasos 2-5 se repiten cada vez que se desee ingresar un nuevo cliente. 7. El caso de uso finaliza. 	
Postcondición	El nuevo cliente queda registrado en el sistema.
Flujo alternativo: Modificar Cliente	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Clientes. 2. El sistema muestra una lista con todos los clientes registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario selecciona uno de los clientes de la lista. 4. El sistema muestra en pantalla los datos del cliente. 5. El usuario modifica los campos convenientes y selecciona Guardar 6. El sistema modifica los datos del cliente y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee modificar un cliente. 8. El caso de uso finaliza. 	
Flujo alternativo: Eliminar Cliente	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Clientes. 2. El sistema muestra una lista con todos los clientes registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario selecciona uno de los clientes de la lista y selecciona Eliminar. 4. El sistema muestra un mensaje de verificación. 5. El usuario acepta la eliminación. 6. El sistema elimina al cliente y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee eliminar un cliente. 8. El caso de uso finaliza. 	
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.6 Mantener Clientes

<i>Mantener Conductores</i>	
Descripción	El propósito de este caso de uso es de administrar los conductores quienes prestan el servicio de taxi.
Actor	Jefe de Operaciones
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Registrar Conductor	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Conductores. 2. El sistema muestra las siguientes cajas de texto: Nombre, Apellido Paterno, Apellido Materno, Dirección, Fecha de Asignación, Teléfono, Celular, E-Mail y Nro. de Brevete, una opción para seleccionar si se encuentra activo o no, además de un listado de Sexo y Vehículo por asignar. 3. El usuario ingresa los datos del nuevo conductor. 4. El usuario selecciona la opción Guardar. 5. El sistema registra el nuevo conductor y muestra un mensaje de conformidad. 6. Los pasos 2-5 se repiten cada vez que se desee ingresar un nuevo conductor. 7. El caso de uso finaliza. 	
Postcondición	El nuevo conductor queda registrado en el sistema.
Flujo alternativo: Modificar Conductor	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Conductores. 2. El sistema muestra una lista con todos los conductores registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario selecciona uno de los conductores de la lista. 4. El sistema muestra en pantalla los datos del conductor. 5. El usuario modifica los campos convenientes y selecciona Guardar. 6. El sistema modifica los datos del conductor y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee modificar un conductor. 8. El caso de uso finaliza. 	
Flujo alternativo: Eliminar Conductor	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Conductores. 2. El sistema muestra una lista con todos los conductores registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario escoge uno de los conductores de la lista y selecciona Eliminar. 4. El sistema muestra un mensaje de verificación. 5. El usuario acepta la eliminación. 6. El sistema elimina al conductor y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee eliminar un conductor. 8. El caso de uso finaliza. 	
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.7 Mantener Conductores

<i>Mantener Vehículos</i>	
Descripción	El propósito de este caso de uso es de administrar la flota de taxis de la empresa.
Actor	Jefe de Operaciones
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Registrar Vehículo	
<ol style="list-style-type: none"> 1. El usuario es coge la opción Vehículos. 2. El sistema muestra las siguientes cajas de texto: Placa, Año, una opción para seleccionar si se encuentra activo o no, además de un listado para saber la Marca, Modelo y Color. 3. El usuario ingresa todos los datos del nuevo vehículo. 4. El usuario selecciona la opción Guardar. 5. El sistema registra el nuevo vehículo y muestra un mensaje de conformidad. 6. Los pasos 2-5 se repiten cada vez que se desee ingresar un nuevo vehículo. 7. El caso de uso finaliza cuando. 	
Postcondición	El nuevo vehículo queda registrado en el sistema.
Flujo alternativo: Modificar Vehículo	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Vehículos. 2. El sistema muestra una lista con todos los vehículos registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario selecciona uno de los vehículos de la lista. 4. El sistema muestra en pantalla los datos del vehículo. 5. El usuario modifica los campos convenientes y selecciona Guardar. 6. El sistema modifica los datos del vehículo y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee modificar un vehículo. 8. El caso de uso finaliza. 	
Flujo alternativo: Eliminar Vehículo	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Vehículos. 2. El sistema muestra una lista con todos los vehículos registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario selecciona uno de los vehículos de la lista y selecciona Eliminar. 4. El sistema muestra un mensaje de verificación. 5. El usuario acepta la eliminación. 6. El sistema elimina el vehículo y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee eliminar un vehículo. 8. El caso de uso finaliza. 	
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.8 Mantener Vehículos

<i>Mantener Tarifas y Cotizaciones</i>	
Descripción	El propósito de este caso de uso es de administrar las tarifas para manejar el costo de los servicios.
Actor	Jefe de Operaciones
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Registrar Tarifa	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Tarifas. 2. El sistema muestra un listado de Departamentos, Provincias y Distritos tanto para el punto inicial y final. 3. El usuario ingresa los datos de la nueva tarifa. 4. El usuario selecciona la opción Guardar. 5. El sistema registra la nueva tarifa y muestra un mensaje de conformidad. 6. Los pasos 2-5 se repiten cada vez que se desee ingresar una nueva tarifa. 7. El caso de uso finaliza.
Postcondición	La nueva tarifa queda registrada en el sistema.
Flujo alternativo: Modificar Tarifa	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Tarifas. 2. El sistema muestra una lista con todas las tarifas registradas, con la posibilidad de agregar, modificar o eliminar una ya existente. 3. El usuario selecciona una de las tarifas de la lista. 4. El sistema muestra en pantalla los datos de la tarifa. 5. El usuario modifica los campos convenientes y selecciona Guardar 6. El sistema modifica los datos de la tarifa y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee modificar una tarifa. 8. El caso de uso finaliza.
Flujo alternativo: Eliminar Tarifa	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Tarifas. 2. El sistema muestra una lista con todas las tarifas registradas, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario selecciona una de las tarifas de la lista y selecciona Eliminar. 4. El sistema muestra un mensaje de verificación. 5. El usuario acepta la eliminación. 6. El sistema elimina la tarifa y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee eliminar una tarifa. 8. El caso de uso finaliza.
Puntos de Excepción	
	<ul style="list-style-type: none"> • No existe punto de excepción.

Tabla A.9 Mantener Tarifas y Cotizaciones

<i>Mantener Lugares de Referencia</i>	
Descripción	El propósito de este caso de uso es de administrar lugares de referencia los cuales servirán para facilitar el registro de solicitudes.
Actor	Jefe de Operaciones
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Registrar Lugar de Referencia	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Lugares. 2. El sistema muestra los campos generales para su ingreso por el usuario como el tipo de lugar y el nombre del lugar. 3. El sistema muestra los campos correspondientes a la dirección del lugar como Distrito, Tipo de Vía, Nombre de la vía, Número y Cuadra. 4. El usuario ingresa los datos generales del lugar y su dirección. 5. El usuario selecciona la opción Guardar. 6. El sistema registra el nuevo lugar y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee ingresar un nuevo lugar de referencia. 8. El caso de uso finaliza. 	
Postcondición	El nuevo lugar de referencia queda registrado en el sistema.
Flujo alternativo: Modificar Lugar de Referencia	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Lugares. 2. El sistema muestra una lista con todos los lugares de referencia registrados, con la posibilidad de agregar, modificar o eliminar una ya existente. 3. El usuario selecciona uno de los lugares de referencia de la lista. 4. El sistema muestra en pantalla los datos del lugar de referencia. 5. El usuario modifica los campos convenientes y selecciona Guardar 6. El sistema modifica los datos del lugar de referencia y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee modificar un lugar de referencia. 8. El caso de uso finaliza. 	
Flujo alternativo: Eliminar Lugar de Referencia	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Lugares. 2. El sistema muestra una lista con todos los lugares de referencia registrados, con la posibilidad de agregar, modificar o eliminar uno ya existente. 3. El usuario selecciona uno de los lugares de referencia de la lista y selecciona Eliminar. 4. El sistema muestra un mensaje de verificación. 5. El usuario acepta la eliminación. 6. El sistema elimina el lugar de referencia y muestra un mensaje de conformidad. 7. Los pasos 2-6 se repiten cada vez que se desee eliminar un lugar de referencia. 8. El caso de uso finaliza. 	
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.10 Mantener Lugares de Referencia

Cargar Grafo	
Descripción	El propósito de este caso de uso es de registrar un grafo representativo de las vías para obtener la trayectoria óptima.
Actor	Jefe de Operaciones
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Cargar Grafo	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Vías -> Cargar Grafo. 2. El sistema muestra la opción Importar. 3. El sistema muestra la opción Guardar. 4. El usuario selecciona la opción Importar. 5. El usuario elige el archivo con la información del grafo. 6. El sistema muestra una tabla con la información de los nodos importados: id, texto, posición y valor si fuese necesario. 7. El sistema muestra una tabla con la información de los arcos importados: id, origen, destino, texto, costo, costo mínimo, costo máximo y grado de variación de la información heurística. 8. El usuario selecciona la opción Guardar. 9. El sistema verifica que todo haya sido correctamente importado y registra el grafo. 10. Los pasos 2-9 se repiten cada vez que se desee cargar un grafo. 11. El caso de uso finaliza. 	
Postcondición	El grafo ha sido cargado.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.11 Cargar Grafo

<i>Configurar Vías</i>	
Descripción	El propósito de este caso de uso es configurar el costo de las vías por donde circulan las unidades o taxis.
Actor	Jefe de Operaciones
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Configurar Vías	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Vías -> Configurar Vías. 2. El sistema muestra una gráfica del grafo. 3. El sistema muestra dos opciones a escoger por el usuario, correspondientes al punto de Origen y el punto Destino, así como el coste y los grados de variación heurística relacionada al arco a configurar. 4. El usuario elige tanto origen como destino y selecciona la opción Buscar. 5. El sistema muestra el coste y los grados de variación heurística relacionados al arco encontrado. 6. El usuario modifica los valores mencionados y selecciona Guardar. 7. El sistema actualiza la información del arco. 8. Los pasos 2-7 se repiten cada vez que se desee configurar una vía. 9. El caso de uso finaliza. 	
Postcondición	La vía ha sido configurada.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.12 Configurar Vías

<i>Configurar Horarios</i>	
Descripción	El propósito de este caso de uso es actualizar el intervalo de horas durante el día por el cual se pueda distinguir tiempos con mayor congestión, tráfico, etc.
Actor	Jefe de Operaciones
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Configurar Horarios	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Vías -> Configurar Horarios. 2. El sistema muestra los turnos operativos del día junto con el intervalo de horas. 3. El usuario selecciona diferentes horas finales para cada turno de acuerdo a sus expectativas. 4. El usuario selecciona la opción Guardar. 5. El sistema actualiza la información de los horarios. 6. Los pasos 2-5 se repiten cada vez que se desee configurar un horario. 7. El caso de uso finaliza.
Postcondición	Los horarios han sido configurados.
Puntos de Excepción	
	<ul style="list-style-type: none"> • No existe punto de excepción.

Tabla A.13 Configurar Horarios

1.1.3. Módulo de Recepción y Asignación de Servicios

<i>Registrar Solicitudes de Servicio</i>	
Descripción	El propósito de este caso de uso es permitir que el usuario pueda iniciar la atención de una solicitud de servicio de taxi.
Actor	Operador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Registrar Solicitud	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Registrar Solicitud. 2. El sistema muestra una pantalla que permita registrar al cliente, la dirección de origen y la dirección de destino (Distrito, Tipo de vía, Nombre de vía, Número y Cuadra; así como opciones de lugares de referencia). Además de registrar la forma de pago, costo y observaciones a la solicitud. Así mismo las opciones Consultar Cliente y Asignar Unidad. 3. Se realiza el caso de uso Consultar Cliente. 4. Se realiza el caso de uso Consultar Unidades. 5. El usuario termina por ingresar los datos para el registro de la solicitud. 6. El usuario selecciona la opción Guardar. 7. Se realiza el caso de uso Obtener Trayectoria Óptima. 8. El sistema registra la solicitud y muestra un mensaje de conformidad. 9. Los pasos 2-8 se repiten cada vez que se desee registrar una solicitud. 10. El caso de uso finaliza cuando el usuario selecciona guardar, cancelar o cerrar. 	
Postcondición	Se registró una solicitud. El servicio está por registrar.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.14 Registrar solicitudes de servicio

<i>Registrar Reservas</i>	
Descripción	El propósito de este caso de uso es permitir que el usuario pueda iniciar la atención de una reserva de taxi.
Actor	Operador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Registrar Reserva	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Registrar Reserva. 2. El sistema muestra una pantalla que permita registrar al cliente, la fecha y hora de recojo, la dirección de origen y la dirección de destino (Distrito, Tipo de vía, Nombre de vía, Número y Cuadra; así como opciones de lugares de referencia). Además de registrar la forma de pago, costo y observaciones a la solicitud. Así mismo las opciones Consultar Cliente y Asignar Unidad. 3. Se realiza el caso de uso Consultar Cliente. 4. Se realiza el caso de uso Consultar Unidades. 5. El usuario termina por ingresar los datos para el registro de la reserva. 6. El usuario selecciona la opción Guardar. 7. Se realiza el caso de uso Obtener Trayectoria Óptima. 8. El sistema registra la reserva y muestra un mensaje de conformidad. 9. Los pasos 2-8 se repiten cada vez que se desee registrar una reserva. 10. El caso de uso finaliza cuando el usuario selecciona guardar, cancelar o cerrar. 	
Postcondición	Se registró una reserva. El servicio está por registrar.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.15 Registrar Reservas

<i>Consultar Clientes</i>	
Descripción	El propósito de este caso de uso es permitir que el usuario pueda acceder a los datos del cliente por atender.
Actor	Operador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Consultar Cliente	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Consultar Cliente. 2. El sistema muestra una lista con los principales clientes registrados. 3. El sistema muestra filtros para realizar la búsqueda: Tipo de cliente, Nombre, Apellido Paterno, Apellido Materno y si es el caso por Empresa o Área. 4. El usuario ingresa los filtros para su búsqueda y selecciona la opción Buscar. 5. El sistema muestra los siguientes datos del cliente: id, nombres, tipo de cliente, apellido paterno, apellido materno, teléfono, celular, forma de pago, empresa y área. 6. Los pasos 2-5 se repiten cada vez que se desee consultar nuevo cliente. 7. El caso de uso finaliza. 	
Postcondición	El cliente ingresado es consultado.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.16 Consultar Clientes

<i>Consultar Solicitudes</i>	
Descripción	El propósito de este caso de uso es permitir que se realice un seguimiento a las solicitudes y reservas ingresadas.
Actor	Operador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Consultar Solicitudes	
<ol style="list-style-type: none"> 1. El usuario escoge la opción Consultar Solicitudes. 2. El sistema muestra una lista con las principales solicitudes registradas. 3. El sistema muestra filtros para realizar la búsqueda: Identificador, Identificador de Unidad, Estado y Cliente. 4. El usuario ingresa los filtros para su búsqueda y selecciona la opción Buscar. 5. El sistema muestra los siguientes datos de la solicitud: Id, Fecha Registro, Dir. Origen, Dir. Destino, Costo, Cliente, Conductor, Estado y para el caso de reservas Fecha de Reservación. 6. Los pasos 2-5 se repiten cada vez que se desee consultar una solicitud. 7. El caso de uso finaliza. 	
Postcondición	La solicitud es consultada.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.17 Consultar Solicitudes

<i>Consultar Unidades</i>	
Descripción	El propósito de este caso de uso es permitir que se de seguimiento a las unidades actualmente disponibles al momento de generar una solicitud de servicio.
Actor	Operador
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Consultar Unidades	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Consultar Unidades. 2. El sistema muestra una lista con las unidades más disponibles, mostrando para cada una de ellas: Identificador, disponibilidad, hora de disponibilidad estimada, dirección de destino, vehículo, radio, conductor, vehículo, solicitud y estado. 3. Los pasos 1-2 se repiten cada vez que se desee consultar unidades. 4. El caso de uso finaliza.
Postcondición	Las unidades son consultadas.
Puntos de Excepción	
	<ul style="list-style-type: none"> • No existe punto de excepción.

Tabla A.18 Consultar Unidades

<i>Iniciar Servicio</i>	
Descripción	El propósito de este caso de uso es permitir que el usuario pueda iniciar un servicio una vez la unidad haya acudido a recoger al cliente.
Actor	Operador
Precondición	El usuario tiene que haberse autenticado. Debe de haber una solicitud de por medio, por ende un taxista asignado.
Flujo básico: Iniciar Servicio	
<ol style="list-style-type: none"> 1. Se ejecuta el caso de uso Consultar Solicitudes. 2. El usuario selecciona la solicitud. 3. El sistema muestra la información de la solicitud registrada. 4. El usuario selecciona la opción Iniciar. 5. El sistema inicializa el servicio. 6. Los pasos 2-5 se repiten cada vez que se desee iniciar un servicio. 7. El caso de uso finaliza. 	
Postcondición	Se registró un servicio.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.19 Iniciar Servicio

<i>Obtener Trayectoria Óptima</i>	
Descripción	El propósito de este caso de uso es obtener la ruta óptima para determinado servicio.
Actor	Operador
Precondición	Tiene que haberse registrado la configuración de parámetros de optimización. Se registra una solicitud de servicio. A los puntos de Origen y de Destino les corresponde nodos del grafo ingresado.
Flujo básico: Registrar Servicio	
	<ol style="list-style-type: none"> 1. El sistema ejecuta el algoritmo para hallar la ruta óptima. 2. El caso de uso finaliza cuando el sistema halla la ruta óptima.
Postcondición	Se halló la ruta óptima.
Puntos de Excepción	
	<ul style="list-style-type: none"> • No existe punto de excepción.

Tabla A.20 Obtener Trayectoria Óptima

<i>Anular solicitudes</i>	
Descripción	El propósito de este caso de uso es permitir anular una solicitud de servicio o reserva por causas externas.
Actor	Operador
Precondición	El usuario tiene que haberse autenticado. Debe haberse registrado la solicitud.
Flujo básico: Anular solicitudes	
<ol style="list-style-type: none"> 1. Se ejecuta el caso de uso Consultar Solicitudes 2. El usuario selecciona la solicitud. 3. El sistema muestra la información de la solicitud registrada. 4. El usuario selecciona la opción Anular. 5. El sistema anula la solicitud. 6. Los pasos 2-5 se repiten cada vez que se desee anular una solicitud. 7. El caso de uso finaliza. 	
Postcondición	Se anuló una solicitud.
Puntos de Excepción	
<ul style="list-style-type: none"> • No existe punto de excepción. 	

Tabla A.21 Anular solicitudes

1.1.4. Módulo de apoyo a la gestión

<i>Generar Reporte Informativo de Solicitudes</i>	
Descripción	El propósito de este caso de uso es permitir generar reportes con información de los servicios realizados.
Actor	Jefe de operaciones
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Generar reporte informativo de solicitudes	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Reportes. 2. El sistema mostrará en pantalla que tipo de servicios desea generar: solicitudes o reservas. 3. El usuario elige el tipo de servicio. 4. El sistema muestra los filtros para generar el reporte: Todos o Identificador de servicio, rango de fechas. 5. El usuario completa los filtros y selecciona Generar. 6. El sistema muestra una tabla preliminar con los resultados: identificador, fecha registro, dirección de origen, dirección de destino, costo, cliente, conductor. 7. El usuario selecciona Ver PDF. 8. El sistema muestra en pantalla el reporte. 9. Los pasos 2-8 se repiten cada vez que se desee generar un reporte. 10. El caso de uso finaliza.
Postcondición	Se genera el reporte informativo
Puntos de Excepción	
	<ul style="list-style-type: none"> • No existe punto de excepción.

Tabla A.22 Generar Reporte Informativo de Solicitudes

<i>Generar Reporte Estadístico</i>	
Descripción	El propósito de este caso de uso es permitir generar reportes con información estadística de solicitudes anuladas y distritos con mayor frecuencia de atenciones.
Actor	Jefe de operaciones
Precondición	El usuario tiene que haberse autenticado.
Flujo básico: Generar reporte Estadístico	
	<ol style="list-style-type: none"> 1. El usuario escoge la opción Reportes. 2. El sistema mostrará en pantalla que tipo de servicios desea generar: solicitudes o reservas. 3. El usuario elige el tipo de servicio. 4. El sistema muestra los filtros para generar el reporte estadístico: Periodo (Anual o Mensual), Ubicación (Por Distrito o todos). 5. El sistema muestra en pantalla el reporte estadístico. 6. Los pasos 2-5 se repiten cada vez que se desee generar un reporte. 7. El caso de uso finaliza.
Postcondición	Se genera el reporte estadístico.
Puntos de Excepción	
	<ul style="list-style-type: none"> • No existe punto de excepción.

Tabla A.23 Generar Reporte Estadístico

ANEXO B: Diagrama de clases

1. Diagrama de Clases de Análisis

A continuación en la figura B.1 se presenta las entidades para el sistema a desarrollar.

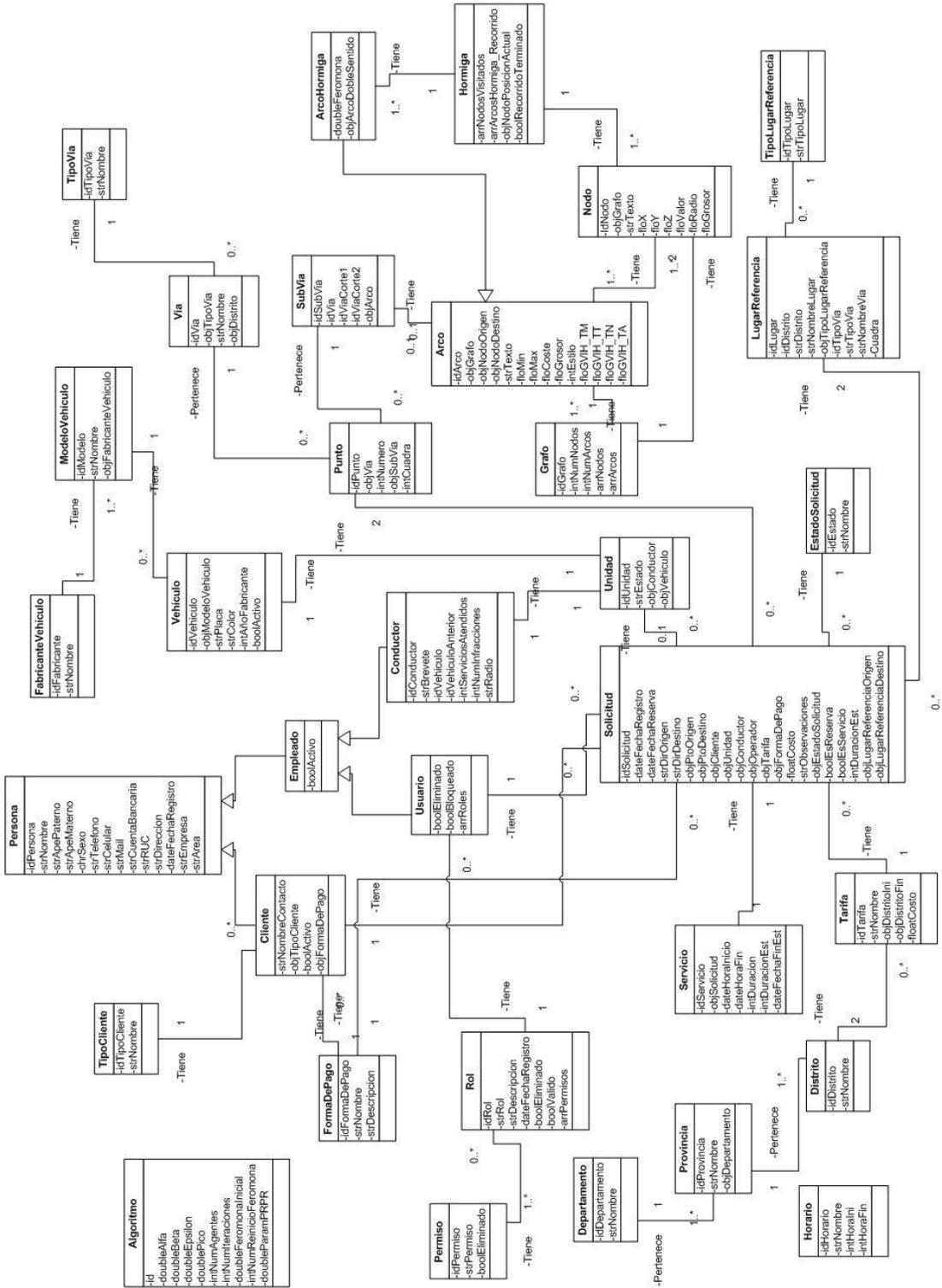


Figura B.1 Diagrama de clases de Análisis

2. Diccionario de datos

A continuación en la tabla B.1 se describe las entidades descritas.

NOMBRE CLASE	DESCRIPCIÓN
Algoritmo	Clase que contiene los parámetros de optimización del algoritmo.
Arco	Clase que representa las conexiones de los nodos de un grafo.
ArcoHormiga	Clase de tipo Arco que representará el recorrido de una hormiga artificial.
Cliente	Clase que contiene a todos los clientes asiduos a ofrecer el servicio.
Conductor	Clase que contiene a todos los conductores o taxistas.
Departamento	Esta clase representa el departamento donde habita una persona.
Distrito	Esta clase representa el distrito donde vive una persona.
Empleado	Clase que contiene los datos de cualquier empleado.
EstadoSolicitud	Clase que representará el estado de una solicitud.
FabricanteVehiculo	Clase que representará el fabricante o marca del vehículo utilizado para brindar servicio.
FormaDePago	Clase que representará la forma de pago del cliente.
Grafo	Clase que contiene todos los nodos y arcos utilizados para obtener la ruta óptima.
Horario	Clase que representará los horarios o turnos de operación.
Hormiga	Clase que representará al agente que realice los recorridos para hallar la ruta óptima.
LugarReferencia	Clase que representará los lugares de referencia para registrar solicitudes de manera rápida.
ModeloVehiculo	Clase que representará el modelo del vehículo utilizado para brindar servicio.
Nodo	Clase que representará los nodos de un grafo.
Permiso	Clase que representará los permisos para brindar acceso al sistema.
Persona	Clase que contiene los datos generales de cualquier persona.
Provincia	Esta clase representa la provincia donde habita una persona.

NOMBRE CLASE	DESCRIPCIÓN
Punto	Clase que representa la ubicación de inicio y destino al momento de brindar servicio.
Rol	Clase que contiene a todos los roles que permite el sistema.
Servicio	Clase que contiene a todos los servicios realizados por los taxistas.
Solicitud	Clase que contiene a todas las solicitudes de servicio por ofrecer.
SubVía	Clase que representa a los componentes de una vía.
Tarifa	Clase que contiene las cotizaciones o tarifario de la empresa.
TipoVía	Clase que representará el tipo de una vía.
Unidad	Clase que representará las unidades tanto conductor como vehículo asignado para ofrecer servicios de taxi.
Usuario	Clase que representará a los usuarios del sistema.
Vehículo	Clase que contiene a todos los taxis de la empresa.
Vía	Clase que representa a todas las vías de tránsito que se encuentren dentro de la cobertura de la empresa de taxi.

Tabla B.1 Diccionario de Datos

ANEXO C: Catálogo de Pruebas

A continuación se presenta las clases equivalentes y los casos de pruebas más relevantes para el sistema.

1. Clases de equivalencia

1.1. Módulo de Seguridad

1.1.1. Actualizar Contraseña

Campos\Clases	Clases Válidas	Clases No Válidas
Contraseña Actual	1<=Cant. caracteres<=6 A	Cant. caracteres <1 o Cant. caracteres >6 B
		Vacío C
Contraseña Nueva	1<=Cant. caracteres<=6 D	Cant. caracteres <1 o Cant. caracteres >6 E
		Vacío F
Repita Contraseña	1<=Cant. caracteres<=6 G	Cant. caracteres <1 o Cant. caracteres >6 H
		Vacío I

Tabla C.1 Actualizar Contraseña.

1.1.2. Mantener Roles

Campos\Clases	Clases Válidas	Clases No Válidas
Rol	1<=Cant. caracteres<=40 A	Cant. caracteres <1 o Cant. caracteres >40 B
		Vacío C
Descripción	1<=Cant. caracteres<=200 D	Cant. caracteres <1 o Cant. caracteres >200 E
		Vacío F
Permisos Existentes	Cant. Permisos Existentes>0 G	Cant. Permisos Existentes=0 H
Permisos Asignados	Cant. Permisos Asignados>0 I	Cant. Permisos Asignados=0 J

Tabla C.2 Mantener Roles.

1.1.3. Mantener Usuarios

Campos\Clases	Clases Válidas	Clases No Válidas
Usuario	1<=Cant. caracteres<=10 A	Cant. caracteres <1 o Cant. caracteres >10 B
		Vacío C
Nombres	1<=Cant. caracteres<=40 D	Cant. caracteres <1 o Cant. caracteres >40 E
		Vacío F
Ap. Paterno	1<=Cant. caracteres<=40 G	Cant. caracteres <1 o Cant. caracteres >40 H
Ap. Materno	1<=Cant. caracteres<=40 I	Cant. caracteres <1 o Cant. caracteres >40 J
Roles Disponibles	Cant. Roles Disponibles>0 K	Cant. Roles Disponibles=0 L
Roles Asignados	Cant. Roles Asignados>0 M	Cant. Roles Asignados =0 N

Tabla C.3 Mantener Usuarios.

1.2. Módulo de Mantenimiento

1.2.1. Configurar Parámetros del Sistema

Campos\Clases	Clases Válidas	Clases No Válidas
Tiempo Actualización de Ventana Principal	1<=Número<=60 A	Número<1 o Número>60 B
		No numérico C
		Vacío D
Departamento	Departamento <> "Seleccionar" E	Departamento = "Seleccionar" F
Provincia	Provincia <> "Seleccionar" G	Provincia = "Seleccionar" H

Tabla C.4 Configurar Parámetros del Sistema.

1.2.2. Configurar Parámetros de optimización

Campos\Clases	Clases Válidas	Clases No Válidas
Número de Agentes	$1 \leq \text{Número} \leq 10000$ A	Número < 1 o Número > 10000 B
		No numérico C
		Vacío D
Número de Iteraciones	$1 \leq \text{Número} \leq 10000$ E	Número < 1 o Número > 10000 F
		No numérico G
		Vacío H
Número de Reinicio de Feromona	$1 \leq \text{Número} \leq (\text{Núm. de Agentes} * \text{Núm. de Iteraciones})$ I	Número < 1 o Número > (Núm. de Agentes * Núm. de Iteraciones) J
		No numérico K
		Vacío L
Valor Probabilístico	$0 \leq \text{Número} \leq 1$ M	Número < 0 o Número > 1 N
		No numérico O
		Vacío P
Valor Heurístico	$0 \leq \text{Número} \leq 1$ Q	Número < 0 o Número > 1 R
		No numérico S
		Vacío T
Valor Aleatorio Proporcional	$0 \leq \text{Número} \leq 1$ U	Número < 0 o Número > 1 V
		No numérico W
		Vacío X
Primer coeficiente de evaporación (Varphi)	$0 \leq \text{Número} \leq 1$ Y	Número < 0 o Número > 1 Z
		No numérico A1
		Vacío B1
Segundo coeficiente de evaporación (Rho)	$0 \leq \text{Número} \leq 1$ C1	Número < 0 o Número > 1 D1
		No numérico E1
		Vacío F1
Feromona Inicial	$0 \leq \text{Número} \leq 40$ G1	Número < 0 o Número > 40 H1
		No numérico I1
		Vacío J1

Tabla C.5 Configurar Parámetros de Optimización.

1.2.3. Mantener Clientes

Campos\Clases	Clases Válidas	Clases No Válidas
Nombre	1<=Cant. Caracteres<=50 A	Cant. Caracteres <1 o Cant. Caracteres >50 B
		Numérico C
		Vacío D
Ape. Paterno	1<=Cant. Caracteres<=50 E	Cant. Caracteres <1 o Cant. Caracteres >50 F
		Numérico G
		Vacío H
Ape. Materno	1<=Cant. Caracteres<=50 I	Cant. Caracteres <1 o Cant. Caracteres >50 J
		Numérico K
		Vacío L
Dirección	1<=Cant. Caracteres<=100 M	Cant. Caracteres <1 o Cant. Caracteres >100 N
		Vacío O
Teléfono	1<=número de dígitos<=10 P	número de dígitos <1 o número de dígitos >10 Q
		Vacío R
Celular	1<= número de dígitos <=12 S	número de dígitos <1 o número de dígitos >12 T
E-Mail	1<=Cant. Caracteres<=50 U	Cant. Caracteres <1 o Cant. Caracteres >50 V
		Vacío W
Tipo Cliente	Tipo Cliente <> "Seleccionar" X	Tipo Cliente = "Seleccionar" Y
Sexo	Sexo <> "Seleccionar" Z	Sexo = "Seleccionar" A1
Empresa	1<=Cant. Caracteres<=50 B1	Cant. Caracteres <1 o Cant. Caracteres >50 C1
Área	1<=Cant. Caracteres<=50 D1	Cant. Caracteres <1 o Cant. Caracteres >50 E1
Forma de Pago	Forma de Pago <> "Seleccionar" F1	Forma de Pago = "Seleccionar" G1
Cta. Bancaria	1<=número de dígitos<=12 H1	número de dígitos <1 o número de dígitos >12 I1

Tabla C.6 Mantener Clientes.

1.2.4. Mantener Conductores

Campos\Clases	Clases Válidas	Clases No Válidas
Nombre	1<=Cant. Caracteres<=50 A	Cant. Caracteres <1 o Cant. Caracteres >50 B
		Numérico C
		Vacío D
Ape. Paterno	1<=Cant. Caracteres<=50 E	Cant. Caracteres <1 o Cant. Caracteres >50 F
		Numérico G
		Vacío H
Ape. Materno	1<=Cant. Caracteres<=50 I	Cant. Caracteres <1 o Cant. Caracteres >50 J
		Numérico K
		Vacío L
Dirección	1<=Cant. Caracteres<=100 M	Cant. Caracteres <1 o Cant. Caracteres >100 N
		Vacío O
Teléfono	1<=número de dígitos <=10 P	número de dígitos <1 o número de dígitos >10 Q
		Vacío R
Celular	1<= número de dígitos <=12 S	número de dígitos <1 o número de dígitos >12 T
E-Mail	1<=Cant. Caracteres<=50 U	Cant. Caracteres <1 o Cant. Caracteres >50 V
		Vacío W
Sexo	Sexo <> "Seleccionar" X	Sexo = "Seleccionar" Y
Nro. Brevete	1<=Cant. Caracteres<=15 Z	Cant. Caracteres <1 o Cant. Caracteres >15 A1
		Vacío B1
Unidad	Unidad <> "Seleccionar" C1	Unidad = "Seleccionar" D1

Tabla C.7 Mantener Conductores.

1.2.5. Mantener Vehículos

Campos\Clases	Clases Válidas	Clases No Válidas
Placa	1<=Cant. Caracteres<=7 A	Cant. Caracteres <1 o Cant. Caracteres >7 B
		Vacío C
Año	2000<=Número<=Año Actual D	Número <2000 o Número > Año Actual E
		Vacío F
Marca	Marca <> "Seleccionar" G	Marca = "Seleccionar" H
Modelo	Modelo <> "Seleccionar" I	Modelo = "Seleccionar" J
Color	Color <> "Seleccionar" K	Color = "Seleccionar" L

Tabla C.8 Mantener Vehículos.

1.2.6. Mantener Tarifas y Cotizaciones

Campos\Clases	Clases Válidas	Clases No Válidas
Distrito Inicial	Distrito Inicial <> "Seleccionar" A	Distrito Inicial = "Seleccionar" B
Distrito Final	Distrito Final <> "Seleccionar" C	Distrito Final = "Seleccionar" D
Costo	0<=Número<=100 E	Número <100 o Número > 0 F
		Vacío G

Tabla C.9 Mantener Tarifas y Cotizaciones.

1.2.7. Mantener Lugares de Referencia

Campos\Clases	Clases Válidas	Clases No Válidas
Tipo Lugar	Tipo Lugar <> "Seleccionar" A	Tipo Lugar = "Seleccionar" B
Nombre Lugar	1<=Cant. Caracteres<=50 C	Cant. Caracteres <1 o Cant. Caracteres >50 D
		Vacío E
Distrito	Distrito <> "Seleccionar" F	Distrito = "Seleccionar" G
Tipo Vía	Tipo Vía <> "Seleccionar" H	Tipo Vía = "Seleccionar" I
Nombre Vía	1<=Cant. Caracteres<=50 J	Cant. Caracteres <1 o Cant. Caracteres >50 K
		Vacío L
Número	0<=Número<=9000 M	Número <9000 o Número > 0 N
		Vacío O
Cuadra	0<=Número<=1000 P	Número <1000 o Número > 0 Q
		Vacío R

Tabla C.10 Mantener Lugares de Referencia.

1.2.8. Configurar Vías

Campos\Clases	Clases Válidas	Clases No Válidas
Nodo Origen	Nodo Origen <> "Seleccionar" A	Nodo Origen = "Seleccionar" B
Nodo Destino	Nodo Destino <> "Seleccionar" C	Nodo Destino = "Seleccionar" D
Costo	0<=Número<=1000 E	Número <1000 o Número > 0 F
		Vacío G
Costo Mín.	0<=Número<=1000 H	Número <1000 o Número > 0 I
		Vacío J
Costo Máx.	0<=Número<=1000 K	Número <1000 o Número > 0 L
		Vacío M
GVIH TA	0<=Número<=100 N	Número <100 o Número > 0 O
		Vacío P
GVIH TM	0<=Número<=100 Q	Número <100 o Número > 0 R
		Vacío S
GVIH TT	0<=Número<=100 T	Número <100 o Número > 0 U
		Vacío V
GVIH TN	0<=Número<=100 W	Número <100 o Número > 0 X

Tabla C.11 Configurar Vías.

1.2.9. Configurar horarios de operación

Campos\Clases	Clases Válidas	Clases No Válidas
Hora Fin Turno Madrugada	Hora Fin Turno Madrugada <> "Seleccionar" A	Hora Fin Turno Madrugada = "Seleccionar" B
Hora Fin Turno Mañana	Hora Fin Turno Mañana <> "Seleccionar" C	Hora Fin Turno Mañana = "Seleccionar" D
Hora Fin Turno Tarde	Hora Fin Turno Tarde <> "Seleccionar" E	Hora Fin Turno Tarde = "Seleccionar" F
Hora Fin Turno Noche	Hora Fin Turno Noche <> "Seleccionar" G	Hora Fin Turno Noche = "Seleccionar" H

Tabla C.12 Configurar Horarios de operación.

1.3. Módulo de Recepción y Asignación de Servicios

1.3.1. Registrar solicitudes de servicio

Campos\Clases	Clases Válidas	Clases No Válidas
Cliente	1<=Cant. Caracteres<=50 A	Cant. Caracteres <1 o Cant. Caracteres >50 B
		Vacío C
Operador	1<=Cant. Caracteres<=50 D	Cant. Caracteres <1 o Cant. Caracteres >50 E
		Vacío F
Distrito Pto. Partida	Distrito <> "Seleccionar" G	Distrito = "Seleccionar" H
Tipo Vía Pto. Partida	Tipo Vía <> "Seleccionar" I	Tipo Vía = "Seleccionar" J
Nombre Vía Pto. Partida	1<=Cant. Caracteres<=50 L	Cant. Caracteres <1 o Cant. Caracteres >50 M
		Vacío N
Número Pto. Partida	0<=Número<=9000 O	Número <9000 o Número > 0 P
		Vacío Q
Cuadra Pto. Partida	0<=Número<=1000 R	Número <1000 o Número > 0 S
		Vacío T
Distrito Pto. Llegada	Distrito <> "Seleccionar" U	Distrito = "Seleccionar" V
Tipo Vía Pto. Llegada	Tipo Vía <> "Seleccionar" W	Tipo Vía = "Seleccionar" X
Nombre Vía Pto. Llegada	1<=Cant. Caracteres<=50 Y	Cant. Caracteres <1 o Cant. Caracteres >50 Z
		Vacío A1
Número Pto. Llegada	0<=Número<=9000 B1	Número <9000 o Número > 0 C1
		Vacío D1
Cuadra Pto. Llegada	0<=Número<=1000 E1	Número <1000 o Número > 0 F1
		Vacío G1
Forma de Pago	Forma de Pago <> "Seleccionar" H1	Forma de Pago = "Seleccionar" I1
Costo	0<=Número<=100 J1	Número <100 o Número > 0 K1
		Vacío L1
Unidad	1<=Cant. Caracteres<=50 M1	Cant. Caracteres <1 o Cant. Caracteres >50 N1
		Vacío O1
Observaciones	1<=Cant. Caracteres<=1000 P1	Cant. Caracteres <1 o Cant. Caracteres >1000 Q1

Tabla C.13 Registrar solicitudes de servicio.

1.4. Módulo de Apoyo a la gestión

1.4.1. Generar Reporte Informativo

Campos\Clases	Clases Válidas	Clases No Válidas
Solicitud	Todas las Solicitudes A	Identificador Solicitud <1 o Identificador Solicitud >10000 C
	1<Identificador Solicitud<10000 B	Vacío D
Fecha Inicio	Fecha Inicio < Fecha Fin E	Fecha Inicio >= Fecha Fin F
Fecha Fin	Fecha Inicio < Fecha Fin G	Fecha Inicio >= Fecha Fin H

Tabla C.14 Generar Reporte Informativo.



2. Casos de pruebas

2.1. Módulo de Seguridad

2.1.1. Actualizar Contraseña

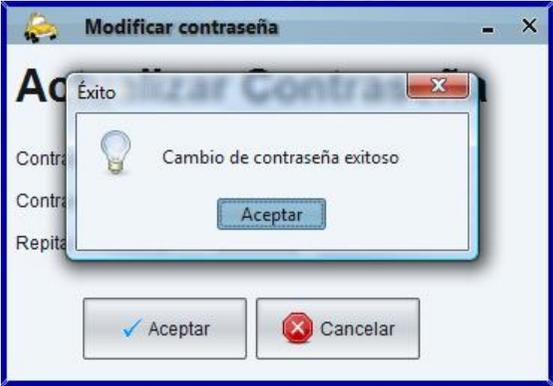
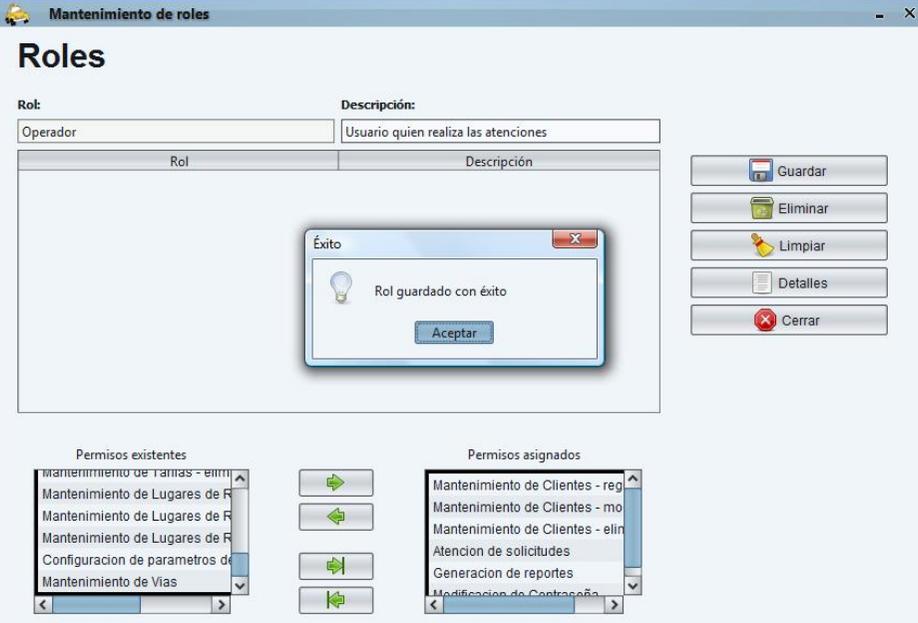
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ADG	123456, 654321, 654321	Mensaje: "Cambio de contraseña exitoso"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Contraseña" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Aceptar. 	Se muestra el mensaje de éxito esperado.
BDG	1234567, 654321,654321	Mensaje: "Contraseña Actual Incorrecta"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Contraseña" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Aceptar. 	Se muestra el mensaje de error esperado.

Tabla C.15 Actualizar Contraseña.

2.1.2. Mantener Roles

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ADGI	Operador, Usuario quien realiza las atenciones, <i>Todos los permisos, Permisos Asignados</i>	Mensaje: "Rol guardado con éxito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Roles" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

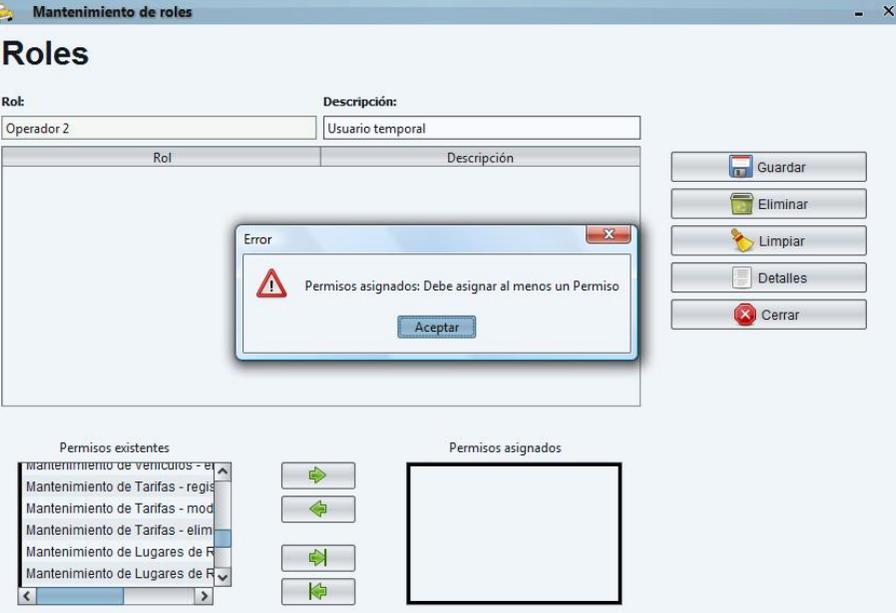
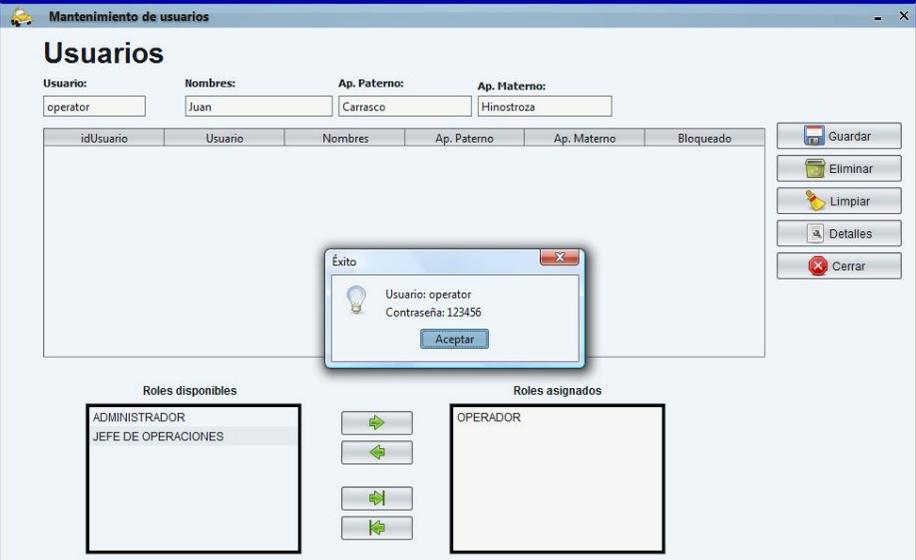
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ADGJ	Operador, Usuario quien realiza las atenciones, <i>Todos los permisos, No se asigna ningún permiso</i>	Mensaje: "Permisos Asignados: Debe asignar al menos un permiso"		<ol style="list-style-type: none"> Se ingresa a la opción "Roles" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Aceptar. 	Se muestra el mensaje de error esperado.

Tabla C.16 Mantener Roles.

2.1.3. Mantener Usuarios

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ADGIKM	Operador, Juan, Carrasco, Hinostroza, <i>Todos los roles disponibles, Un rol por asignar.</i>	Mensaje: "Usuario: Operador Contraseña: 13456"		<ol style="list-style-type: none"> Se ingresa a la opción "Usuarios" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

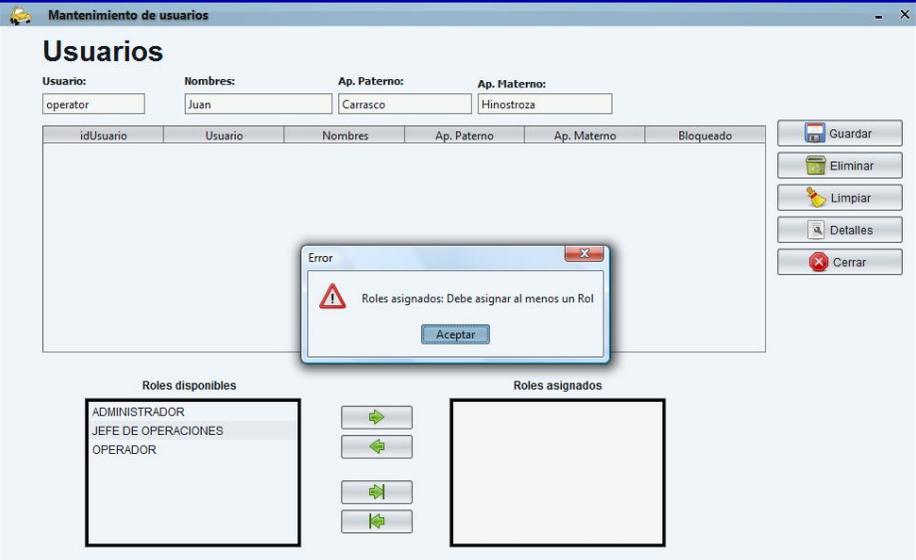
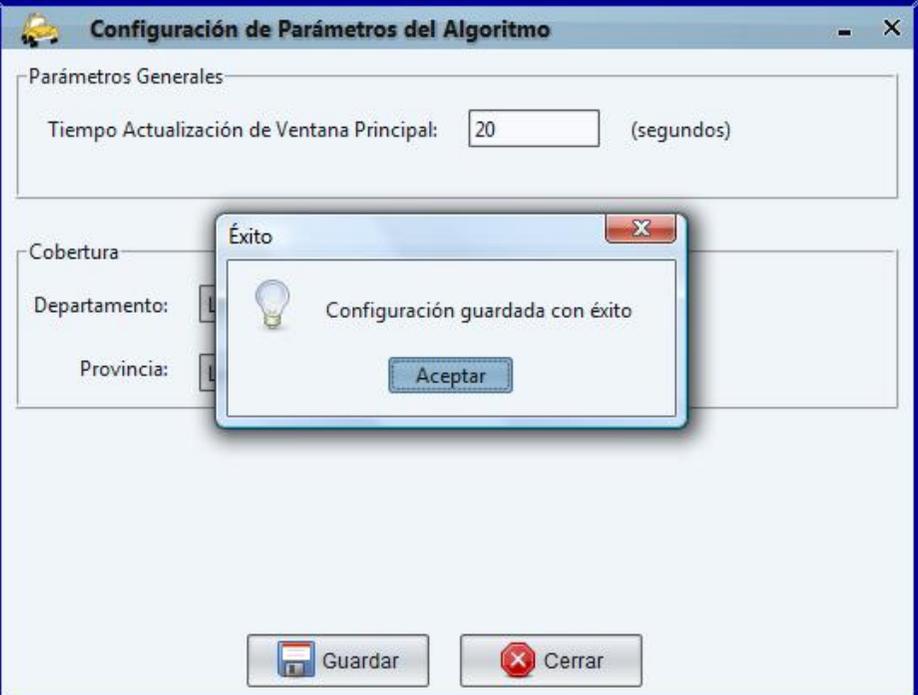
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ADGIKN	Operador, Juan, Carrasco, Hinostroza, <i>Todos los roles disponibles, Ningún rol asignado.</i>	Mensaje: "Roles Asignados: Debe asignar al menos un rol."		<ol style="list-style-type: none"> Se ingresa a la opción "Usuarios" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.17 Mantener Usuarios.

2.2. Módulo de Mantenimiento

2.2.1. Configurar Parámetros del Sistema

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
AEG	20, Lima, Lima	Mensaje: "Configuración guardada con éxito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Parámetros->Sistema" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

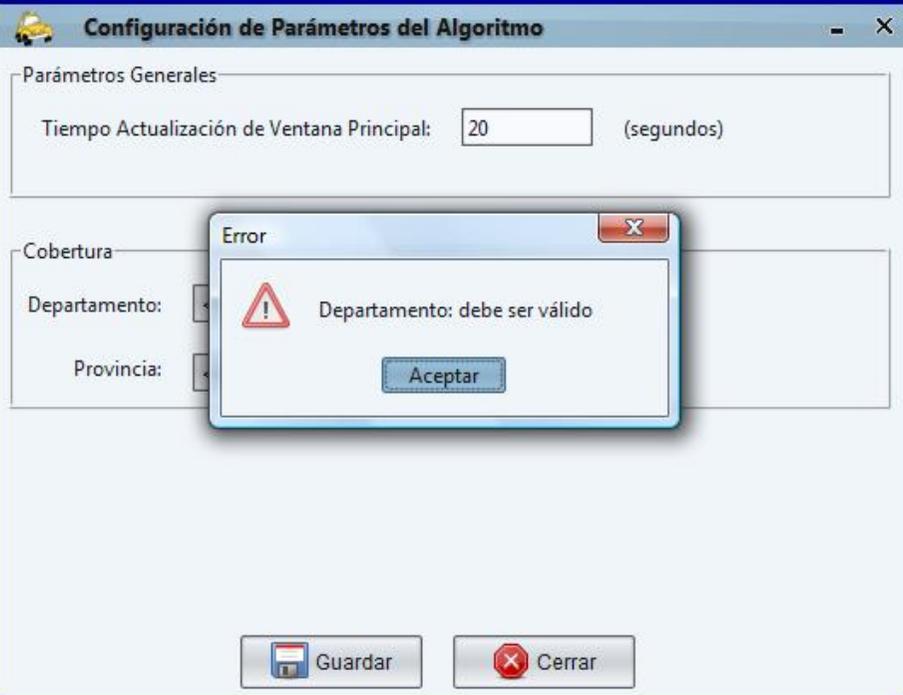
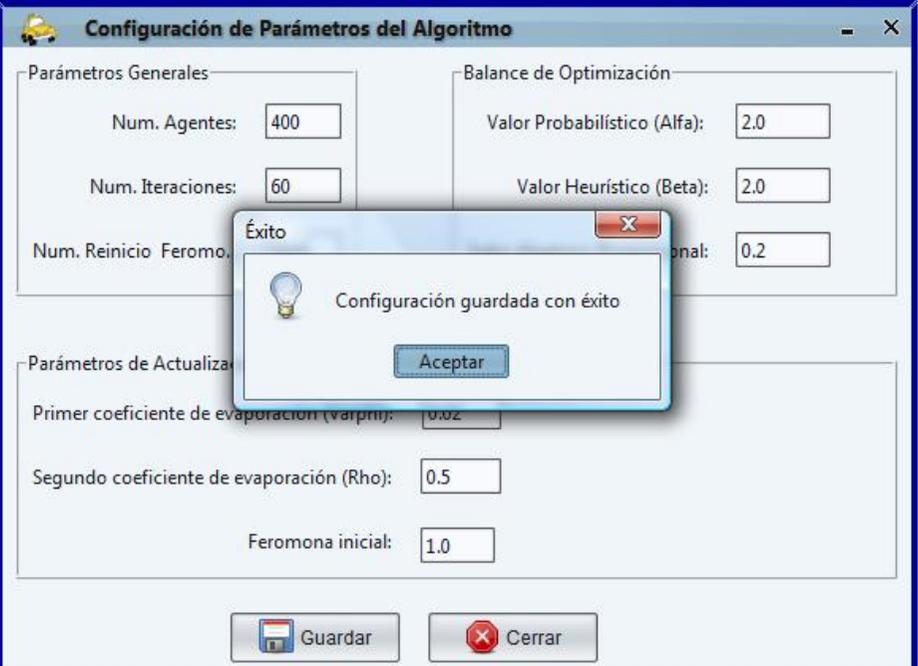
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
AFH	Lima, Seleccionar, Seleccionar	Mensaje: "Departamento: debe ser válido"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Parámetros->Sistema" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.18 Configurar Parámetros del Sistema.

2.2.2. Configurar Parámetros de optimización

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
AEIMQUY C1G1	400, 60, 2000, 2.0, 2.0, 0.2, 0.02, 0.5, 1.0	Mensaje: "Configuración guardada con éxito"		<ol style="list-style-type: none"> Se ingresa a la opción "Parámetros->Algoritmo" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

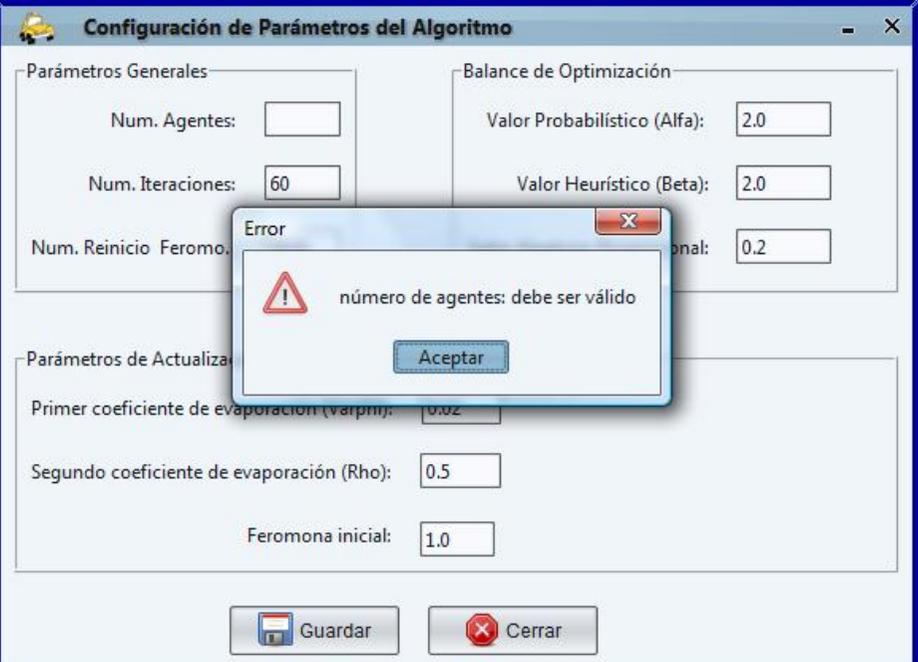
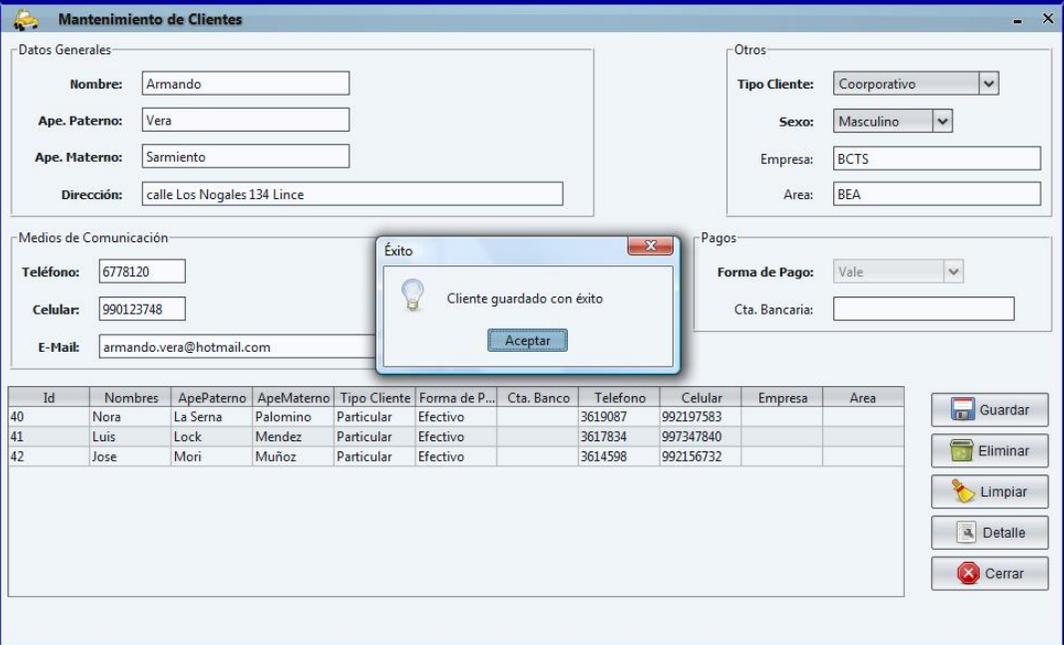
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
BEIMQUY C1G1	Cadena Vacía, 60, 2000, 2.0, 2.0, 0.2, 0.02, 0.5, 1.0	Mensaje: "Número de agentes: debe ser válido."		<ol style="list-style-type: none"> Se ingresa a la opción "Parámetros->Algoritmo" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.19 Configurar Parámetros de Optimización.

2.2.3. Mantener Clientes

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
AEIMPS UXZB1 D1F1H1	Armando, Vera, Sarmiento, calles Los Nogales 134 Lince, 6778120, 990123748, Corporativo, Masculino, BCTS, BEA, Vale	Mensaje: "Cliente guardada con éxito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Clientes" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

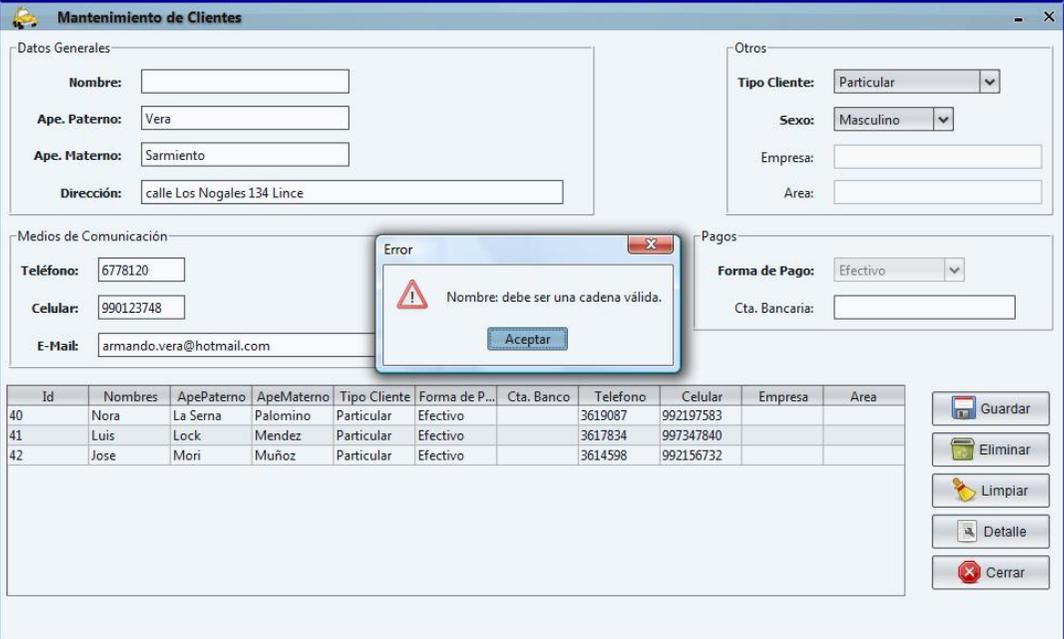
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
DEIMPS UXZB1 D1F1H1	Cadena Vacía, Vera, Sarmiento, calles Los Nogales 134 Lince, 6778120, 990123748, Corporativo, Masculino, BCTS, BEA, Vale.	Mensaje: "Nombre: debe ser válido."		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Clientes" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.20 Mantener Clientes.

2.2.4. Mantener Conductores

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
AEIMPS UXZC1	Eduardo, Gutiérrez, Villarán, Calles Los Jarabes 633 Lince, 5771284, 981736618, eduardo.gutierrez@hotmail.com , Masculino, 00112313412, AOH-466	Mensaje: "Conductor guardada con éxito"		<ol style="list-style-type: none"> Se ingresa a la opción "Conductores" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

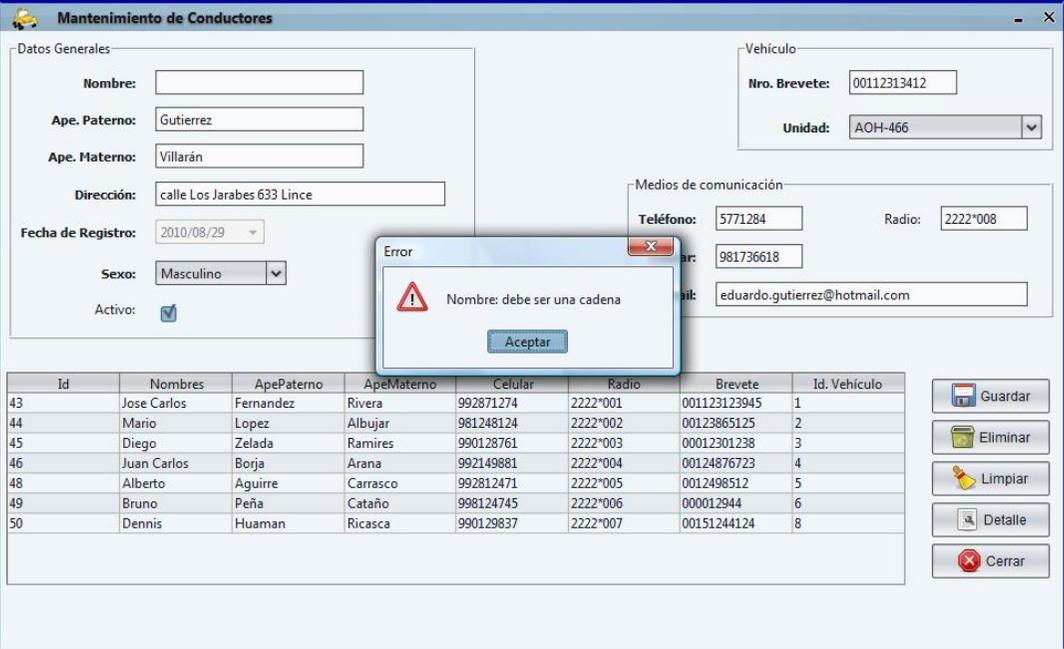
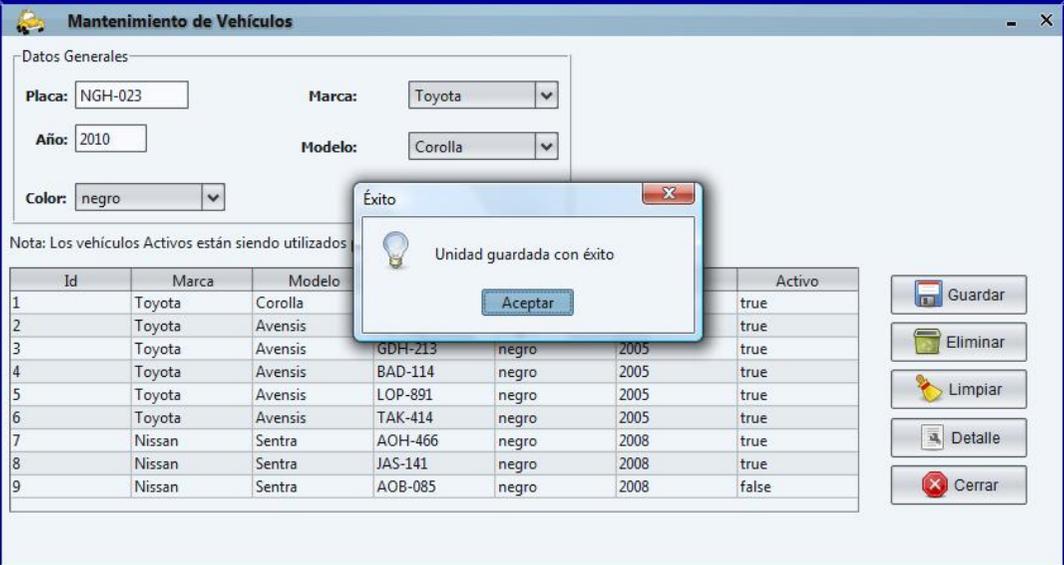
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
DEIMPS UXZC1	Cadena Vacía, Gutiérrez, Villarán, Calles Los Jarabes 633 Lince, 5771284, 981736618, eduardo.gutierrez@hotmail.com , Masculino, 00112313412, AOH-466	Mensaje: "Nombre: debe ser válido."	 <p>The screenshot shows a web application window titled "Mantenimiento de Conductores". It contains several form fields for "Datos Generales" (Nombre, Ape. Paterno, Ape. Materno, Dirección, Fecha de Registro, Sexo, Activo) and "Vehículo" (Nro. Brevete, Unidad). There is also a "Medios de comunicación" section with fields for Teléfono, Radio, Celular, and Email. An error dialog box is overlaid on the form, displaying a warning icon and the text "Error: Nombre: debe ser una cadena" with an "Aceptar" button. Below the form is a table listing drivers with columns for Id, Nombres, ApePaterno, ApeMaterno, Celular, Radio, Brevete, and Id. Vehiculo. On the right side of the application, there are buttons for "Guardar", "Eliminar", "Limpiar", "Detalle", and "Cerrar".</p>	<ol style="list-style-type: none"> 1. Se ingresa a la opción "Conductores" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.21 Mantener Conductores.

2.2.5. Mantener Vehículos

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ADGIK	NGH-023, 2010, Toyota, Corola, negro	Mensaje: "Unidad guardada con éxito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Vehículos" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

MCMXVII

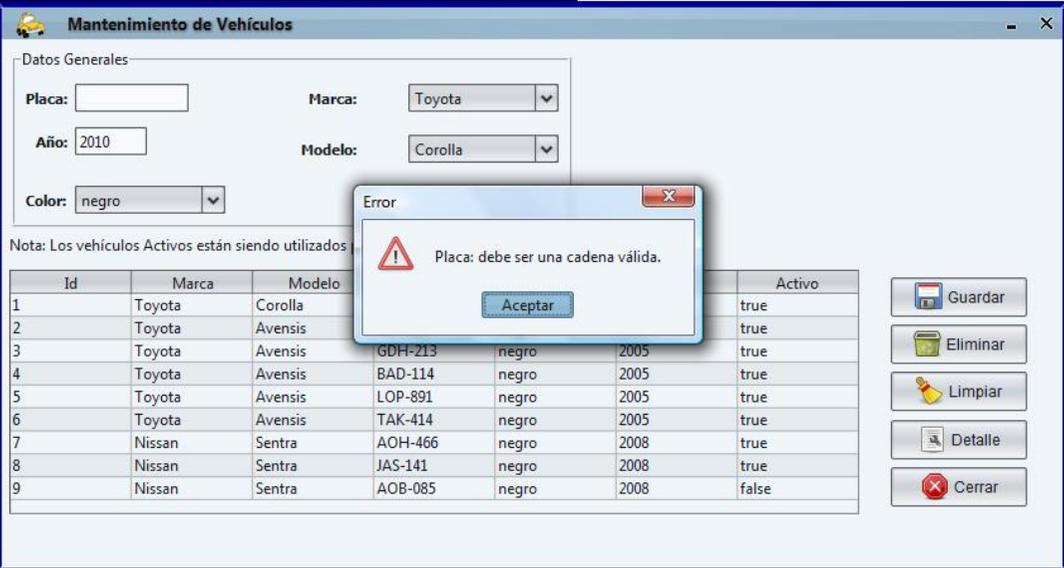
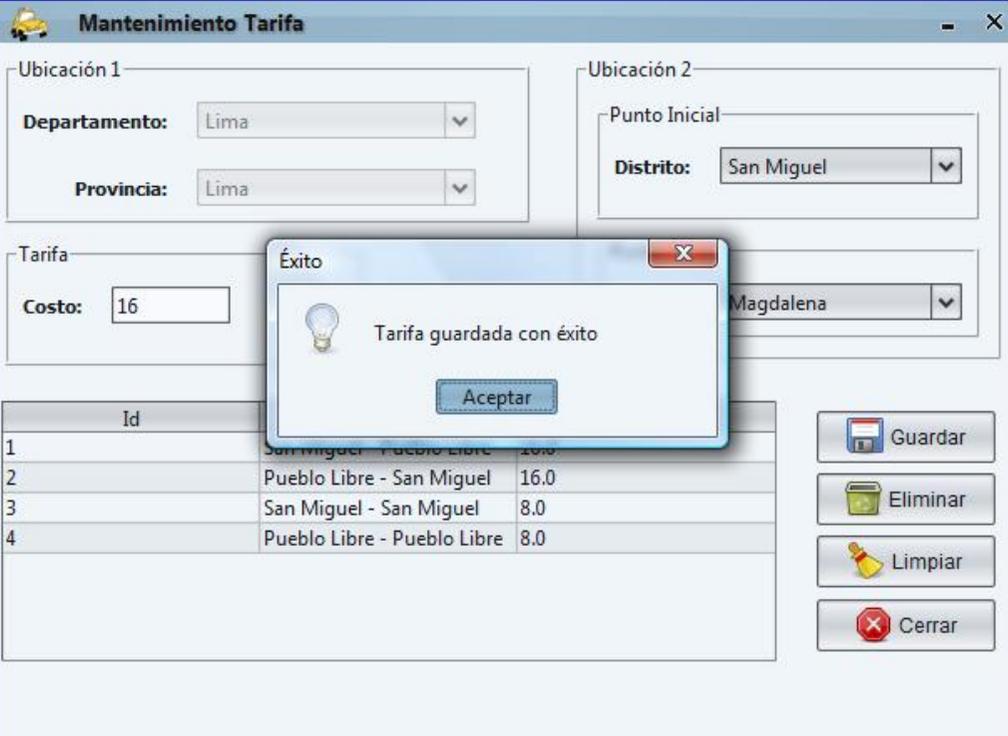
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
CDGIK	Cadena Vacía, 2010, Toyota, Corola, negro	Mensaje: "Placa: debe ser válido."		<ol style="list-style-type: none"> Se ingresa a la opción "Vehículos" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.22 Mantener Vehículos.

2.2.6. Mantener Tarifas y Cotizaciones

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ACE	San Miguel, Magdalena, 16.0	Mensaje: "Tarifa guardada con éxito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Tarifas" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

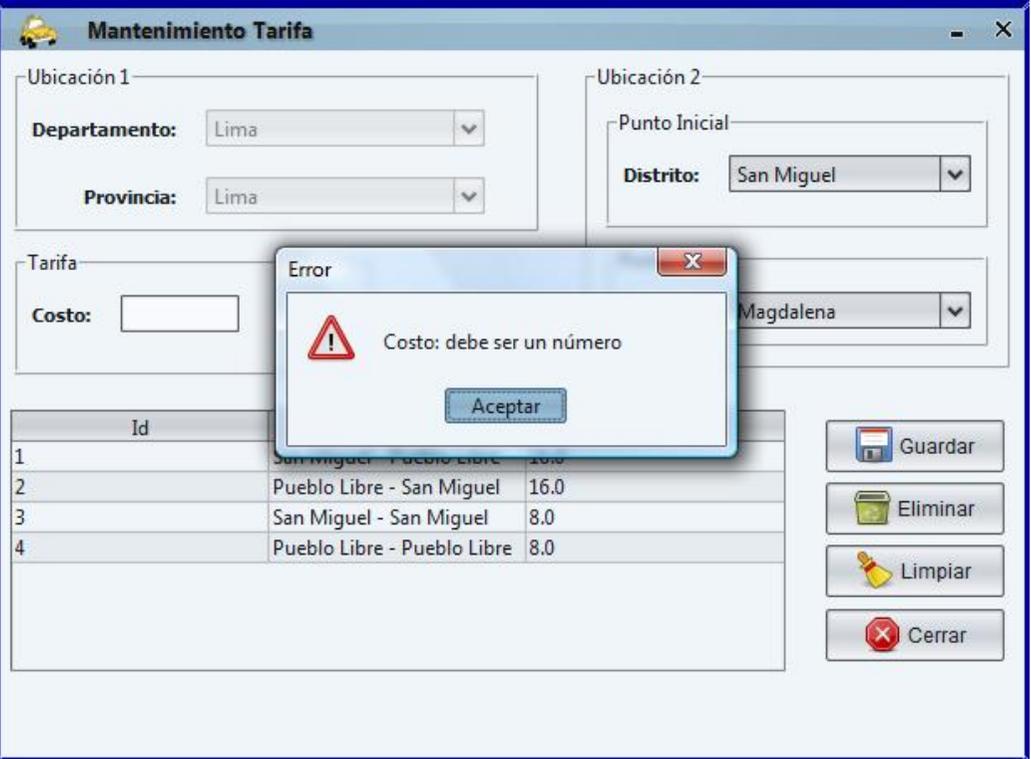
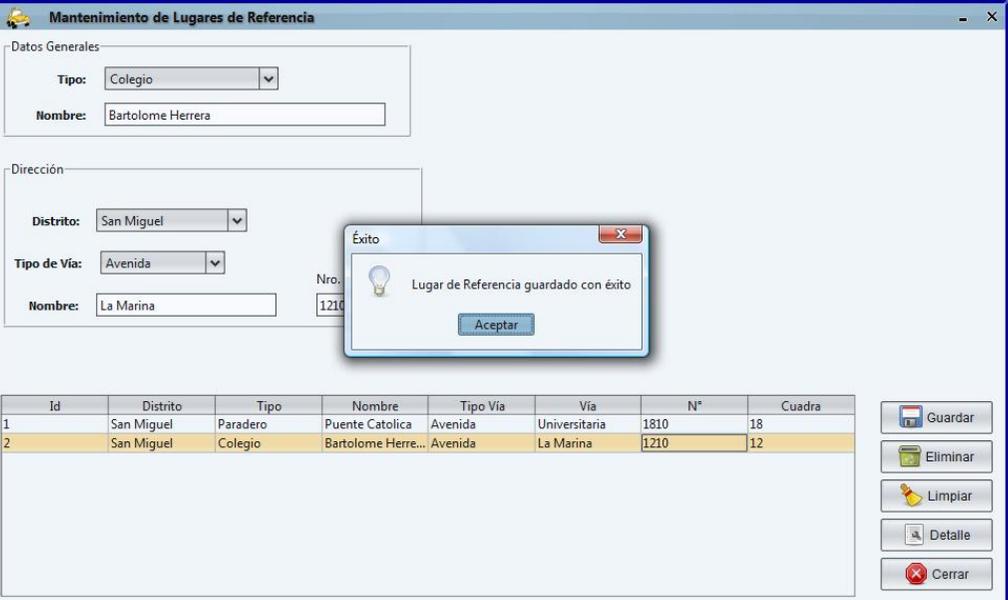
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ACG	San Miguel, Magdalena, Cadena Vacía,	Mensaje: "Costo: debe ser un número válido."		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Tarifas" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.23 Mantener Tarifas y Cotizaciones.

2.2.7. Mantener Lugares de Referencia

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ACFHJM P	Colegio, Bartolomé Herrera San Miguel, Avenida, La Marina 1210, 12	Mensaje: "Lugar de Referencia guardado con éxito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Lugar de Referencia" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

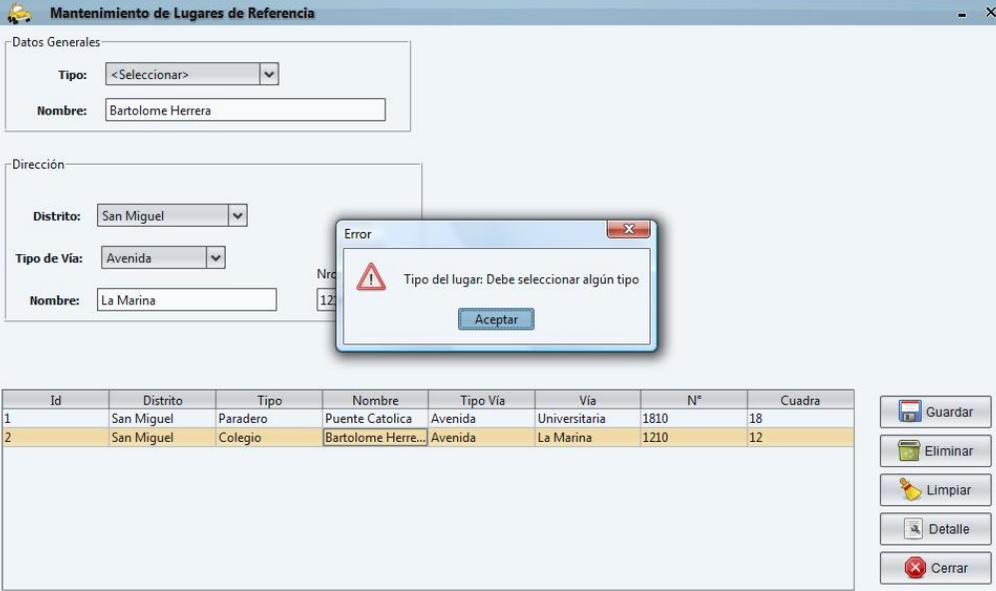
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
BCFHJM P	<Seleccionar>, Bartolomé Herrera San Miguel, Avenida, La Marina 1210, 12	Mensaje: "Tipo de Lugar: debe seleccionar algún tipo."		<ol style="list-style-type: none"> Se ingresa a la opción "Lugar de Referencia" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.24 Mantener Lugares de Referencia.

2.2.8. Configurar Vías

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ACEHKNQ TW	Nodo 1, Nodo 39, 120, 0,0,0,0,0,0	Mensaje: "Configuración de vía guardado con éxito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Vías -> Configurar Vías" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

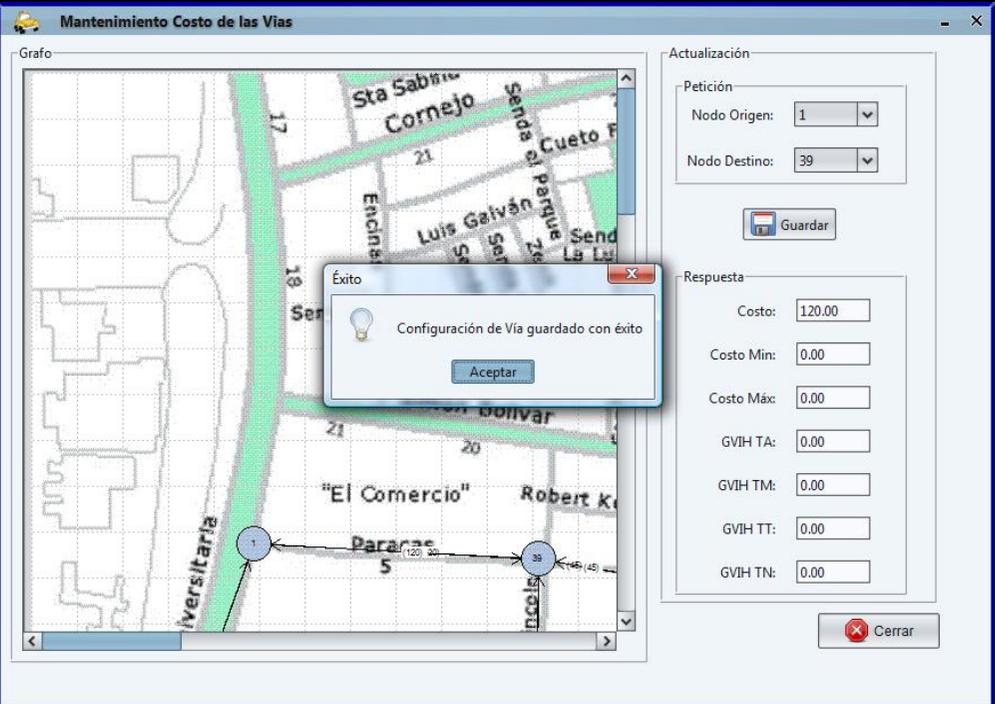
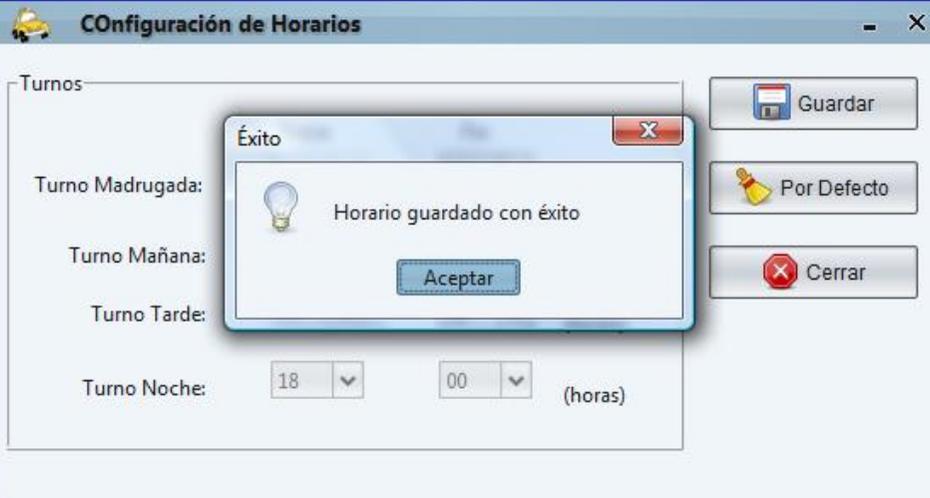
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ACEHKNQ TW	Nodo 1, Nodo 39, 120, 0,0,0,0,0,0	Mensaje: "Configuración de vía guardado con éxito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Vías -> Configurar Vías" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

Tabla C.25 Configurar Vías.

2.2.9. Configurar Horarios de operación

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ACEG	06:00, 12:00, 18:00, 00:00	Mensaje: "Horario guardado con éxito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Vías -> Configurar Horario" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

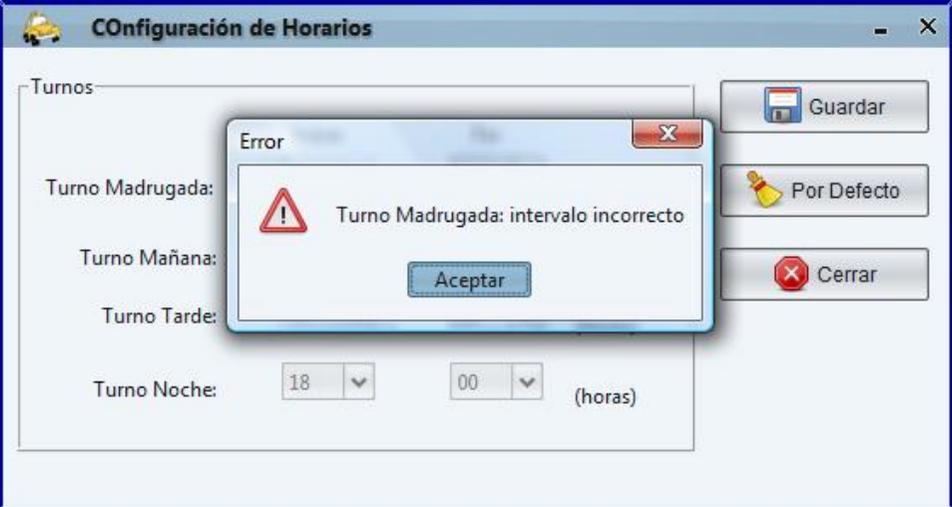
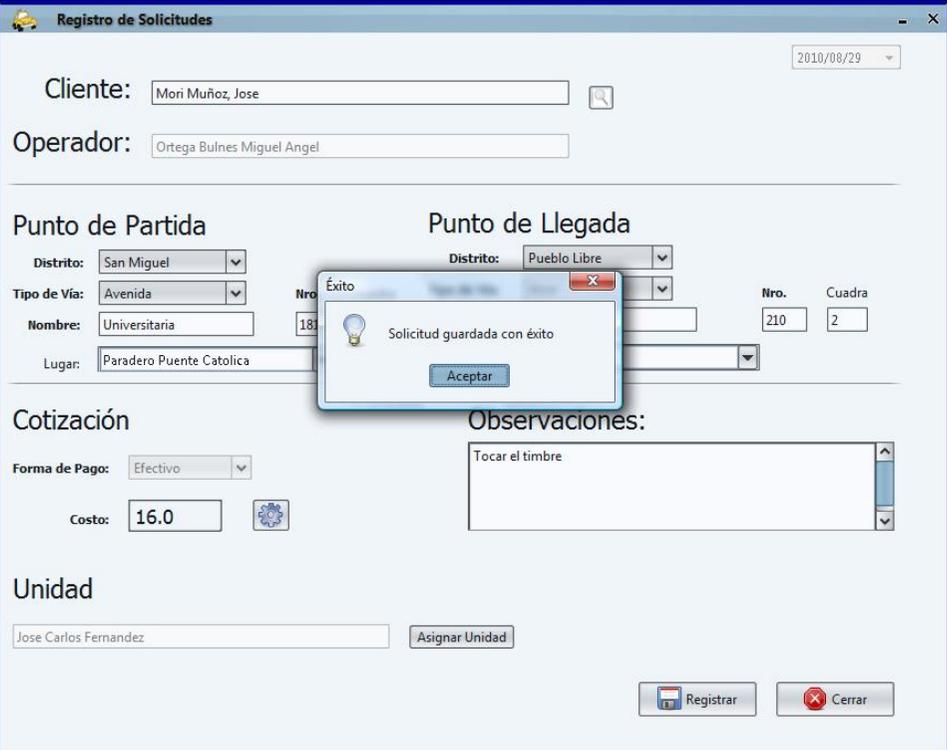
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
BCEG	00:00, 12:00, 18:00, 00:00	Mensaje: "Turno Madrugada: intervalo incorrecto"		<ol style="list-style-type: none"> Se ingresa a la opción "Vías -> Configurar Horario" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.26 Configurar Horarios de operación.

2.3. Módulo de Recepción y Asignación de Servicios

2.3.1. Registrar solicitudes de servicio

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ADGILORUW YB1E1H1J1 M1P1	Mori Muñoz José, Ortega Bulnes Miguel Ángel, <Seleccionar>, <Seleccionar>, Cadena Vacía, Cadena Vacía, Cadena Vacía, <Seleccionar>, <Seleccionar>, Cadena Vacía, Cadena Vacía, Cadena Vacía, efectivo, 16.00, José Carlos Fernández, Tocar el timbre	Mensaje: "Solicitud guardada con éxito"		<ol style="list-style-type: none"> Se ingresa a la opción "Registrar Solicitud" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.

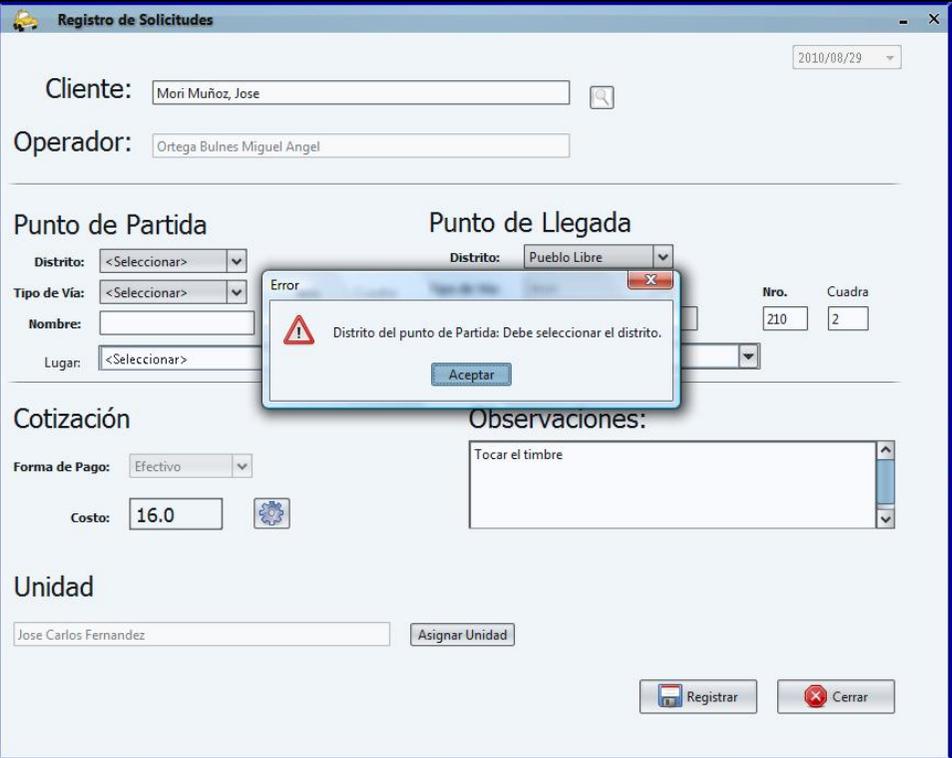
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
ADHILORUW YB1E1H1J1 M1P1	Mori Muñoz José, Ortega Bulnes Miguel Ángel, San Miguel, Avenida, <i>Universitaria</i> , 1810, 18, Pueblo Libre, Jirón, <i>Paracas</i> , 210, 2, efectivo, 16.00, José Carlos Fernández, Tocar el timbre	Mensaje: "Distrito de Partida: Debe seleccionar el distrito"		<ol style="list-style-type: none"> 1. Se ingresa a la opción "Registrar Solicitud" en el sistema. 2. Se ingresa los valores determinados. 3. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.27 Registrar solicitudes de servicio.

2.4. Módulo de Apoyo a la gestión

2.4.1. Generar Reporte Informativo

Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción																												
AEG	Todos, 01/08/2010, 31/08/2010	Listado de solicitudes	 <table border="1"> <thead> <tr> <th>IdSolicitud</th> <th>Fecha Registro</th> <th>Dir. Origen</th> <th>Dir. Destino</th> <th>Costo</th> <th>Cliente</th> <th>Conductor</th> </tr> </thead> <tbody> <tr> <td>17</td> <td>29/08/10 6:49 PM</td> <td>Jiron Paracas 210, C...</td> <td>Avenida Universitari...</td> <td>16.0</td> <td>LuisLockMendez</td> <td>Jose Carlos Fernand...</td> </tr> <tr> <td>16</td> <td>29/08/10 6:46 PM</td> <td>Jiron Paracas 210, C...</td> <td>Avenida Universitari...</td> <td>16.0</td> <td>Nora La Serna</td> <td>Jose Carlos Fernand...</td> </tr> <tr> <td>15</td> <td>29/08/10 6:13 PM</td> <td>Avenida Universitari...</td> <td>Jiron Paracas 210, C...</td> <td>16.0</td> <td>LuisLockMendez</td> <td>Mario Lopez Albuja...</td> </tr> </tbody> </table>	IdSolicitud	Fecha Registro	Dir. Origen	Dir. Destino	Costo	Cliente	Conductor	17	29/08/10 6:49 PM	Jiron Paracas 210, C...	Avenida Universitari...	16.0	LuisLockMendez	Jose Carlos Fernand...	16	29/08/10 6:46 PM	Jiron Paracas 210, C...	Avenida Universitari...	16.0	Nora La Serna	Jose Carlos Fernand...	15	29/08/10 6:13 PM	Avenida Universitari...	Jiron Paracas 210, C...	16.0	LuisLockMendez	Mario Lopez Albuja...	<ol style="list-style-type: none"> Se ingresa a la opción "Reporte de Solicitud" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de éxito esperado.
IdSolicitud	Fecha Registro	Dir. Origen	Dir. Destino	Costo	Cliente	Conductor																											
17	29/08/10 6:49 PM	Jiron Paracas 210, C...	Avenida Universitari...	16.0	LuisLockMendez	Jose Carlos Fernand...																											
16	29/08/10 6:46 PM	Jiron Paracas 210, C...	Avenida Universitari...	16.0	Nora La Serna	Jose Carlos Fernand...																											
15	29/08/10 6:13 PM	Avenida Universitari...	Jiron Paracas 210, C...	16.0	LuisLockMendez	Mario Lopez Albuja...																											

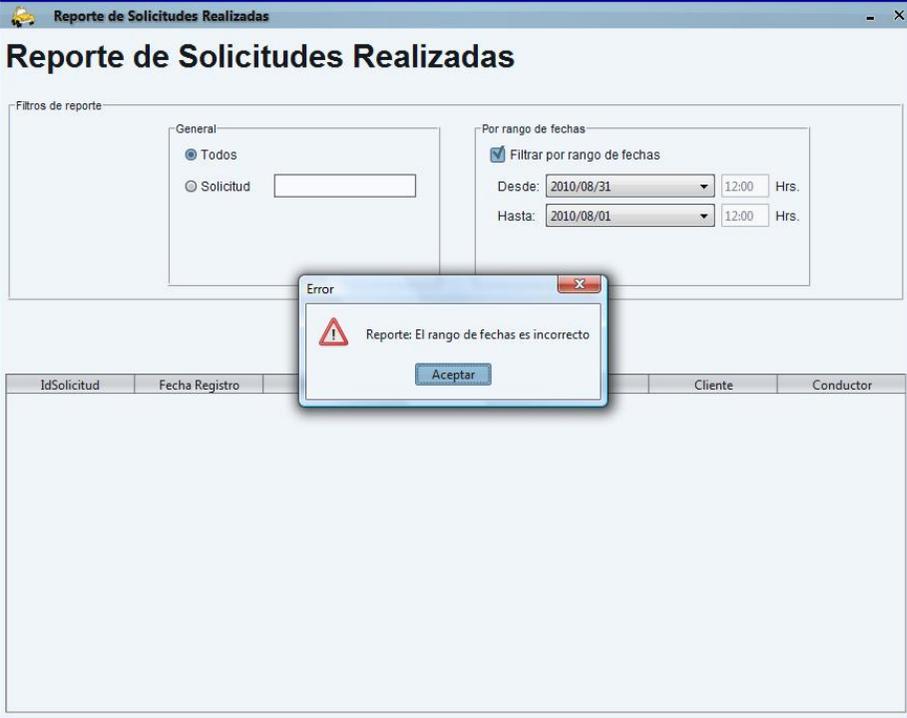
Clases	Valores	Valor Esperado	Pantalla	Secuencia de pasos	Resultado / Descripción
AFH	Todos, 31/08/2010, 01/08/2010	Mensaje: "Reporte: El rango de fechas es incorrecto"		<ol style="list-style-type: none"> Se ingresa a la opción "Reporte de Solicitud" en el sistema. Se ingresa los valores determinados. Se hace clic en el botón Guardar. 	Se muestra el mensaje de error esperado.

Tabla C.28 Generar Reporte Informativo.

Experimentación Numérica
Algoritmo Ant Colony System

Caso # 1											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	740	0
2	1	200	-	1	2	0.1	0.1	0.9	1	740	0
3	1	200	-	1	2	0.1	0.1	0.9	1	780	40
4	1	200	-	1	2	0.1	0.1	0.9	1	790	50
5	5	200	-	1	2	0.1	0.1	0.9	1	740	0
6	5	200	-	1	2	0.1	0.1	0.9	1	740	0
7	5	200	-	1	2	0.1	0.1	0.9	1	740	0
8	5	200	-	1	2	0.1	0.1	0.9	1	740	0

Caso # 1
Variable de estudio: Distancia total recorrida
Nodo Origen: 103
Nodo Destino: 12
Distancia mínima: 740
Tamaño de la muestra: 12

Caso # 2											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	780	50
2	1	200	-	1	2	0.1	0.1	0.9	1	730	0
3	1	200	-	1	2	0.1	0.1	0.9	1	730	0
4	1	200	-	1	2	0.1	0.1	0.9	1	760	30
5	5	200	-	1	2	0.1	0.1	0.9	1	760	30
6	5	200	-	1	2	0.1	0.1	0.9	1	780	50
7	5	200	-	1	2	0.1	0.1	0.9	1	760	30
8	5	200	-	1	2	0.1	0.1	0.9	1	730	0
9	5	200	-	1	2	0.02	0.5	0.9	1	730	0
10	5	200	-	1	2	0.02	0.5	0.9	1	730	0
11	5	200	-	1	2	0.02	0.5	0.9	1	730	0
12	5	200	-	1	2	0.02	0.5	0.9	1	730	0

Caso # 2
Variable de estudio: Distancia total recorrida
Nodo Origen: 12
Nodo Destino: 103
Distancia mínima: 730
Tamaño de la muestra: 28

Caso # 3											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	1200	210
2	1	200	-	1	2	0.1	0.1	0.9	1	1120	130
3	1	200	-	1	2	0.1	0.1	0.9	1	1050	60
4	1	200	-	1	2	0.1	0.1	0.9	1	1240	250
5	5	200	-	1	2	0.1	0.1	0.9	1	1200	210
6	5	200	-	1	2	0.1	0.1	0.9	1	1030	40
7	5	200	-	1	2	0.1	0.1	0.9	1	1170	180
8	5	200	-	1	2	0.1	0.1	0.9	1	1200	210
9	5	200	-	1	2	0.02	0.5	0.9	1	1200	210
10	5	200	-	1	2	0.02	0.5	0.9	1	1000	10
11	5	200	-	1	2	0.02	0.5	0.9	1	990	0
12	5	200	-	1	2	0.02	0.5	0.9	1	990	0
13	10	200	-	1	2	0.02	0.5	0.9	1	1170	180
14	10	200	-	1	2	0.02	0.5	0.9	1	990	0
15	10	200	-	1	2	0.02	0.5	0.9	1	990	0
16	10	200	-	1	2	0.02	0.5	0.9	1	990	0
17	20	200	2000	1	2	0.02	0.5	0.9	1	990	0
18	20	200	2000	1	2	0.02	0.5	0.9	1	990	0
19	20	200	2000	1	2	0.02	0.5	0.9	1	990	0
20	20	200	2000	1	2	0.02	0.5	0.9	1	990	0

Caso # 3
Variable de estudio: Distancia total recorrida
Nodo Origen: 161
Nodo Destino: 107
Distancia mínima: 990
Tamaño de la muestra: 20

Caso # 4											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	990	0
2	1	200	-	1	2	0.1	0.1	0.9	1	1195	205
3	1	200	-	1	2	0.1	0.1	0.9	1	1050	60
4	1	200	-	1	2	0.1	0.1	0.9	1	1055	65
5	5	200	-	1	2	0.1	0.1	0.9	1	990	0
6	5	200	-	1	2	0.1	0.1	0.9	1	1120	130
7	5	200	-	1	2	0.1	0.1	0.9	1	1050	60
8	5	200	-	1	2	0.1	0.1	0.9	1	1105	115
9	5	200	-	1	2	0.02	0.5	0.9	1	990	0
10	5	200	-	1	2	0.02	0.5	0.9	1	990	0
11	5	200	-	1	2	0.02	0.5	0.9	1	990	0
12	5	200	-	1	2	0.02	0.5	0.9	1	1160	170
13	10	200	-	1	2	0.02	0.5	0.9	1	990	0

Caso # 4
Variable de estudio: Distancia total recorrida
Nodo Origen: 107
Nodo Destino: 161
Distancia mínima: 990
Tamaño de la muestra: 20

14	10	200	-	1	2	0.02	0.5	0.9	1	990	0
15	10	200	-	1	2	0.02	0.5	0.9	1	990	0
16	10	200	-	1	2	0.02	0.5	0.9	1	1120	130
17	20	200	2000	1	2	0.02	0.5	0.9	1	990	0
18	20	200	2000	1	2	0.02	0.5	0.9	1	990	0
19	20	200	2000	1	2	0.02	0.5	0.9	1	990	0
20	20	200	2000	1	2	0.02	0.5	0.9	1	990	0

Caso # 5											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	1170	240
2	1	200	-	1	2	0.1	0.1	0.9	1	1135	205
3	1	200	-	1	2	0.1	0.1	0.9	1	1225	295
4	1	200	-	1	2	0.1	0.1	0.9	1	1345	415
5	5	200	-	1	2	0.1	0.1	0.9	1	1135	205
6	5	200	-	1	2	0.1	0.1	0.9	1	1185	255
7	5	200	-	1	2	0.1	0.1	0.9	1	1130	200
8	5	200	-	1	2	0.1	0.1	0.9	1	1120	190
9	5	200	-	1	2	0.02	0.5	0.9	1	930	0
10	5	200	-	1	2	0.02	0.5	0.9	1	1135	205
11	5	200	-	1	2	0.02	0.5	0.9	1	1240	310
12	5	200	-	1	2	0.02	0.5	0.9	1	1080	150
13	10	200	-	1	2	0.02	0.5	0.9	1	1185	255
14	10	200	-	1	2	0.02	0.5	0.9	1	1120	190
15	10	200	-	1	2	0.02	0.5	0.9	1	930	0
16	10	200	-	1	2	0.02	0.5	0.9	1	1170	240
17	20	200	2000	1	2	0.02	0.5	0.9	1	930	0
18	20	200	2000	1	2	0.02	0.5	0.9	1	1135	205
19	20	200	2000	1	2	0.02	0.5	0.9	1	930	0
20	20	200	2000	1	2	0.02	0.5	0.9	1	1060	130
21	30	300	2000	1	2	0.02	0.5	0.9	1	930	0
22	30	300	2000	1	2	0.02	0.5	0.9	1	930	0
23	30	300	2000	1	2	0.02	0.5	0.9	1	960	30
24	30	300	2000	1	2	0.02	0.5	0.9	1	1120	190
25	40	300	2000	1	2	0.02	0.5	0.9	1	930	0
26	40	300	2000	1	2	0.02	0.5	0.9	1	930	0
27	40	300	2000	1	2	0.02	0.5	0.9	1	930	0
28	40	300	2000	1	2	0.02	0.5	0.9	1	930	0

Caso # 5
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 103
 Nodo Destino: 170
 Distancia mínima: 930
 Tamaño de la muestra: 68

Caso # 6											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	880	225
2	1	200	-	1	2	0.1	0.1	0.9	1	965	310
3	1	200	-	1	2	0.1	0.1	0.9	1	820	165
4	1	200	-	1	2	0.1	0.1	0.9	1	1145	490
5	5	200	-	1	2	0.1	0.1	0.9	1	675	20
6	5	200	-	1	2	0.1	0.1	0.9	1	760	105
7	5	200	-	1	2	0.1	0.1	0.9	1	675	20
8	5	200	-	1	2	0.1	0.1	0.9	1	795	140
9	40	300	2000	1	2	0.02	0.5	0.9	1	655	0
10	40	300	2000	1	2	0.02	0.5	0.9	1	675	20
11	40	300	2000	1	2	0.02	0.5	0.9	1	675	20
12	40	300	2000	1	2	0.02	0.5	0.9	1	675	20
13	40	400	2000	1	2	0.02	0.5	0.9	1	655	0
14	40	400	2000	1	2	0.02	0.5	0.9	1	675	20
15	40	400	2000	1	2	0.02	0.5	0.9	1	655	0
16	40	400	2000	1	2	0.02	0.5	0.9	1	655	0
17	50	400	2000	1	2	0.02	0.5	0.9	1	675	0
18	50	400	2000	1	2	0.02	0.5	0.9	1	655	0
19	50	400	2000	1	2	0.02	0.5	0.9	1	655	0
20	50	400	2000	1	2	0.02	0.5	0.9	1	655	0

Caso # 6
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 93
 Nodo Destino: 17
 Distancia mínima: 655
 Tamaño de la muestra: 72

Caso # 7											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	790	0
2	1	200	-	1	2	0.1	0.1	0.9	1	820	30
3	1	200	-	1	2	0.1	0.1	0.9	1	880	90
4	1	200	-	1	2	0.1	0.1	0.9	1	815	25
5	5	200	-	1	2	0.1	0.1	0.9	1	790	0
6	5	200	-	1	2	0.1	0.1	0.9	1	790	0

Caso # 7
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 0
 Nodo Destino: 61
 Distancia mínima: 790
 Tamaño de la muestra: 12



7	5	200	-	1	2	0.1	0.1	0.9	1	790	0
8	5	200	-	1	2	0.1	0.1	0.9	1	815	25
9	50	400	2000	1	2	0.02	0.5	0.9	1	790	0
10	50	400	2000	1	2	0.02	0.5	0.9	1	790	0
11	50	400	2000	1	2	0.02	0.5	0.9	1	790	0
12	50	400	2000	1	2	0.02	0.5	0.9	1	790	0

Caso # 8											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	960	220
2	1	200	-	1	2	0.1	0.1	0.9	1	860	120
3	1	200	-	1	2	0.1	0.1	0.9	1	875	135
4	1	200	-	1	2	0.1	0.1	0.9	1	1005	265
5	5	200	-	1	2	0.1	0.1	0.9	1	870	130
6	5	200	-	1	2	0.1	0.1	0.9	1	975	235
7	5	200	-	1	2	0.1	0.1	0.9	1	800	60
8	5	200	-	1	2	0.1	0.1	0.9	1	915	175
9	50	400	2000	1	2	0.02	0.5	0.9	1	740	0
10	50	400	2000	1	2	0.02	0.5	0.9	1	740	0
11	50	400	2000	1	2	0.02	0.5	0.9	1	740	0
12	50	400	2000	1	2	0.02	0.5	0.9	1	740	0

Caso # 8
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 33
 Nodo Destino: 87
 Distancia mínima: 740
 Tamaño de la muestra: 72

Caso # 9											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	630	0
2	1	200	-	1	2	0.1	0.1	0.9	1	790	160
3	1	200	-	1	2	0.1	0.1	0.9	1	700	70
4	1	200	-	1	2	0.1	0.1	0.9	1	835	205
5	5	200	-	1	2	0.1	0.1	0.9	1	630	0
6	5	200	-	1	2	0.1	0.1	0.9	1	750	120
7	5	200	-	1	2	0.1	0.1	0.9	1	630	0
8	5	200	-	1	2	0.1	0.1	0.9	1	630	0
9	50	400	2000	1	2	0.02	0.5	0.9	1	630	0
10	50	400	2000	1	2	0.02	0.5	0.9	1	630	0
11	50	400	2000	1	2	0.02	0.5	0.9	1	630	0
12	50	400	2000	1	2	0.02	0.5	0.9	1	630	0

Caso # 9
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 4
 Nodo Destino: 107
 Distancia mínima: 630
 Tamaño de la muestra: 16

Caso # 10											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	940	90
2	1	200	-	1	2	0.1	0.1	0.9	1	875	25
3	1	200	-	1	2	0.1	0.1	0.9	1	850	0
4	1	200	-	1	2	0.1	0.1	0.9	1	880	30
5	5	200	-	1	2	0.1	0.1	0.9	1	850	0
6	5	200	-	1	2	0.1	0.1	0.9	1	875	25
7	5	200	-	1	2	0.1	0.1	0.9	1	875	25
8	5	200	-	1	2	0.1	0.1	0.9	1	850	0
9	50	400	2000	1	2	0.02	0.5	0.9	1	850	0
10	50	400	2000	1	2	0.02	0.5	0.9	1	850	0
11	50	400	2000	1	2	0.02	0.5	0.9	1	850	0
12	50	400	2000	1	2	0.02	0.5	0.9	1	850	0

Caso # 10
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 61
 Nodo Destino: 161
 Distancia mínima: 850
 Tamaño de la muestra: 32

Caso # 11											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	1105	225
2	1	200	-	1	2	0.1	0.1	0.9	1	1175	295
3	1	200	-	1	2	0.1	0.1	0.9	1	1085	205
4	1	200	-	1	2	0.1	0.1	0.9	1	1055	175
5	5	200	-	1	2	0.1	0.1	0.9	1	1190	310
6	5	200	-	1	2	0.1	0.1	0.9	1	1000	120
7	5	200	-	1	2	0.1	0.1	0.9	1	1055	175
8	5	200	-	1	2	0.1	0.1	0.9	1	1100	220
9	50	400	2000	1	2	0.02	0.5	0.9	1	880	0
10	50	400	2000	1	2	0.02	0.5	0.9	1	880	0
11	50	400	2000	1	2	0.02	0.5	0.9	1	880	0
12	50	400	2000	1	2	0.02	0.5	0.9	1	920	40
13	60	400	2000	1	2	0.02	0.5	0.9	1	880	0
14	60	400	2000	1	2	0.02	0.5	0.9	1	880	0
15	60	400	2000	1	2	0.02	0.5	0.9	1	880	0

Caso # 11
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 167
 Nodo Destino: 106
 Distancia mínima: 880
 Tamaño de la muestra: 72

*750

16	60	400	2000	1	2	0.02	0.5	0.9	1	880	0
----	----	-----	------	---	---	------	-----	-----	---	-----	---

Caso # 12											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	1000	295
2	1	200	-	1	2	0.1	0.1	0.9	1	910	205
3	1	200	-	1	2	0.1	0.1	0.9	1	850	145
4	1	200	-	1	2	0.1	0.1	0.9	1	910	205
5	5	200	-	1	2	0.1	0.1	0.9	1	705	0
6	5	200	-	1	2	0.1	0.1	0.9	1	910	205
7	5	200	-	1	2	0.1	0.1	0.9	1	790	85
8	5	200	-	1	2	0.1	0.1	0.9	1	820	115
9	50	400	2000	1	2	0.02	0.5	0.9	1	705	0
10	50	400	2000	1	2	0.02	0.5	0.9	1	705	0
11	50	400	2000	1	2	0.02	0.5	0.9	1	705	0
12	50	400	2000	1	2	0.02	0.5	0.9	1	705	0
13	60	400	2000	1	2	0.02	0.5	0.9	1	705	0
14	60	400	2000	1	2	0.02	0.5	0.9	1	705	0
15	60	400	2000	1	2	0.02	0.5	0.9	1	705	0
16	60	400	2000	1	2	0.02	0.5	0.9	1	705	0

Caso # 12
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 95
 Nodo Destino: 163
 Distancia mínima: 705
 Tamaño de la muestra: 72

Caso # 13											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	600	0
2	1	200	-	1	2	0.1	0.1	0.9	1	600	0
3	1	200	-	1	2	0.1	0.1	0.9	1	600	0
4	1	200	-	1	2	0.1	0.1	0.9	1	600	0
5	5	200	-	1	2	0.1	0.1	0.9	1	600	0
6	5	200	-	1	2	0.1	0.1	0.9	1	600	0
7	5	200	-	1	2	0.1	0.1	0.9	1	600	0
8	5	200	-	1	2	0.1	0.1	0.9	1	600	0
9	60	400	2000	1	2	0.02	0.5	0.9	1	600	0
10	60	400	2000	1	2	0.02	0.5	0.9	1	600	0
11	60	400	2000	1	2	0.02	0.5	0.9	1	600	0
12	60	400	2000	1	2	0.02	0.5	0.9	1	600	0

Caso # 13
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 35
 Nodo Destino: 0
 Distancia mínima: 600
 Tamaño de la muestra: 8

Caso # 14											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	520	0
2	1	200	-	1	2	0.1	0.1	0.9	1	555	35
3	1	200	-	1	2	0.1	0.1	0.9	1	550	30
4	1	200	-	1	2	0.1	0.1	0.9	1	670	150
5	5	200	-	1	2	0.1	0.1	0.9	1	520	0
6	5	200	-	1	2	0.1	0.1	0.9	1	520	0
7	5	200	-	1	2	0.1	0.1	0.9	1	520	0
8	5	200	-	1	2	0.1	0.1	0.9	1	520	0
9	60	400	2000	1	2	0.02	0.5	0.9	1	520	0
10	60	400	2000	1	2	0.02	0.5	0.9	1	520	0
11	60	400	2000	1	2	0.02	0.5	0.9	1	520	0
12	60	400	2000	1	2	0.02	0.5	0.9	1	520	0

Caso # 14
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 49
 Nodo Destino: 15
 Distancia mínima: 520
 Tamaño de la muestra: 12

Caso # 15											
N°	N° Iteraciones	N° de Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q_0	τ_0	Distancia (metros)	Variación (+)
1	1	200	-	1	2	0.1	0.1	0.9	1	915	120
2	1	200	-	1	2	0.1	0.1	0.9	1	1295	500
3	1	200	-	1	2	0.1	0.1	0.9	1	985	190
4	1	200	-	1	2	0.1	0.1	0.9	1	855	60
5	5	200	-	1	2	0.1	0.1	0.9	1	855	60
6	5	200	-	1	2	0.1	0.1	0.9	1	855	60
7	5	200	-	1	2	0.1	0.1	0.9	1	855	60
8	5	200	-	1	2	0.1	0.1	0.9	1	855	60
9	60	400	2000	1	2	0.02	0.5	0.9	1	795	0
10	60	400	2000	1	2	0.02	0.5	0.9	1	795	0
11	60	400	2000	1	2	0.02	0.5	0.9	1	795	0
12	60	400	2000	1	2	0.02	0.5	0.9	1	795	0

Caso # 15
 Variable de estudio: Distancia total recorrida
 Nodo Origen: 174
 Nodo Destino: 79
 Distancia mínima: 795
 Tamaño de la muestra: 72

N° Caso	Nodo Origen	Nodo Destino	Distancia Mín.	Parámetros de Optimización								
				N° Iteraciones	N° Agentes	N° para Reiniciar el Rastro de Feromona	α	β	ϕ	ρ	q0	τ_0
1	103	12	740	5	200	-	1	2	0.1	0.1	0.9	1
2	12	103	730	5	200	-	1	2	0.02	0.5	0.9	1
3	161	107	990	20	200	2000	1	2	0.02	0.5	0.9	1
4	107	161	990	20	200	2000	1	2	0.02	0.5	0.9	1
5	103	170	930	40	300	2000	1	2	0.02	0.5	0.9	1
6	93	17	655	50	400	2000	1	2	0.02	0.5	0.9	1
7	0	61	790	50	400	2000	1	2	0.02	0.5	0.9	1
8	33	87	740	50	400	2000	1	2	0.02	0.5	0.9	1
9	4	107	630	50	400	2000	1	2	0.02	0.5	0.9	1
10	61	161	850	50	400	2000	1	2	0.02	0.5	0.9	1
11	167	106	880	60	400	2000	1	2	0.02	0.5	0.9	1
12	95	163	705	60	400	2000	1	2	0.02	0.5	0.9	1
13	35	0	600	60	400	2000	1	2	0.02	0.5	0.9	1
14	49	15	520	60	400	2000	1	2	0.02	0.5	0.9	1
15	174	79	795	60	400	2000	1	2	0.02	0.5	0.9	1

Parámetros de Optimización	Valor
N° Iteraciones	60
N° Agentes	400
N° Reinicio Feromona	2000
Alfa	1
Beta	2
Phi	0.02
Rho	0.5
Valor Aleatorio Proporcional	0.9
Feromona Inicial	1

ANEXO E: Tecnologías utilizadas

1. Tecnologías

Esta sección contiene una presentación de las tecnologías más representativas utilizadas en el desarrollo del sistema de información propuesto.

1.1. Descripción

1.1.1. iReports

Nombre : iReports
Tipo : Software
Uso : Libre
Versión : 3.7.2
Función : Permitir diseñar los reportes informativos y estadísticos del sistema.

1.1.2. Grafos

Nombre : Grafos
Tipo : Software
Uso : Libre
Versión : 1.2.9
Función : Permitir construir un grafo con información de las zonas y calles de ciudades.

1.1.3. GrahML File Format

Nombre : GraphML
Tipo : Formato de archivo
Uso : Libre
Versión : 1.0

Función : Permitir representar la estructura de un grafo y sus propiedades en un archivo basado en XML.

1.2. Presentación

1.2.1. iReports



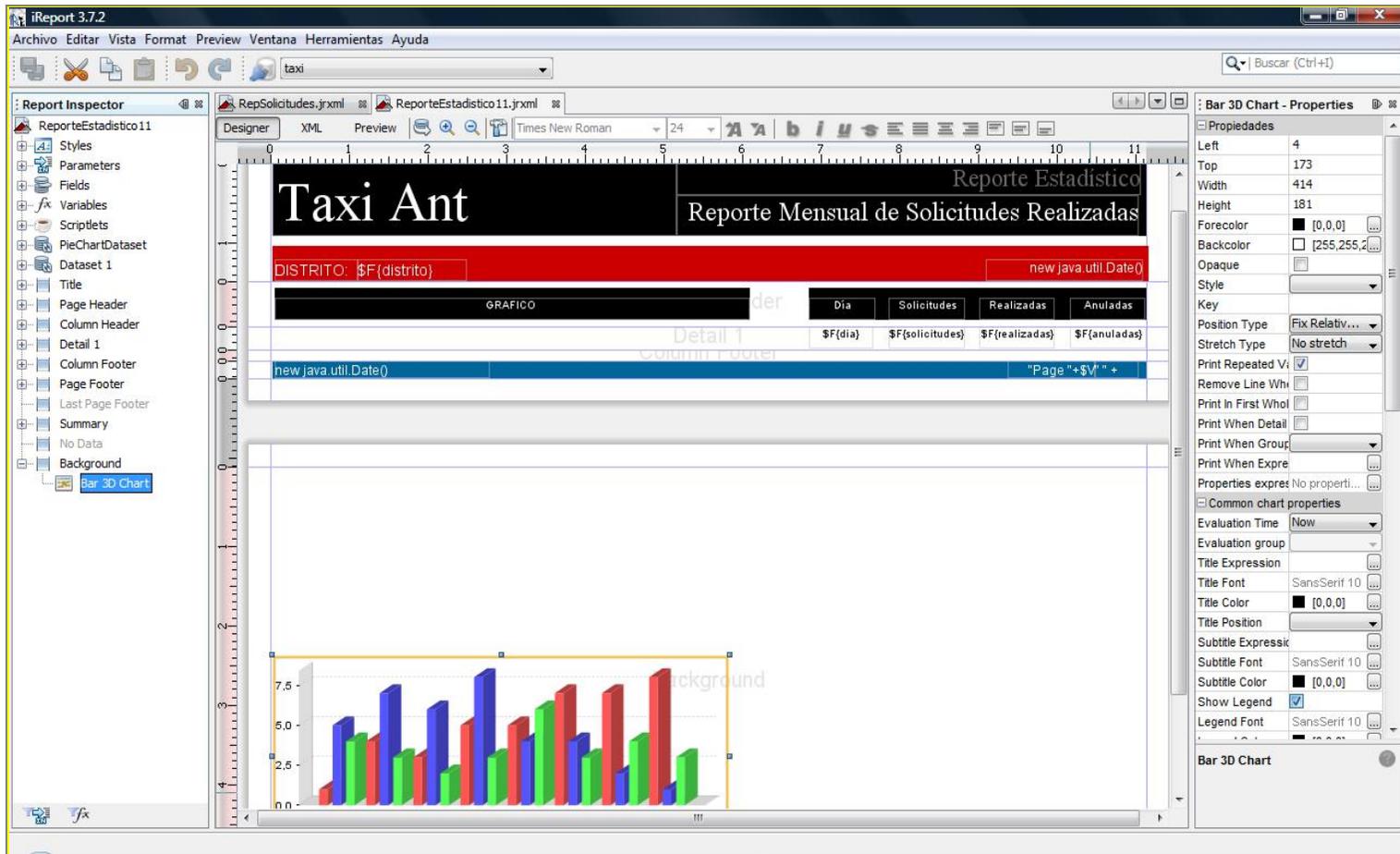


Figura E.1 Software iReports.

1.2.2. Grafos

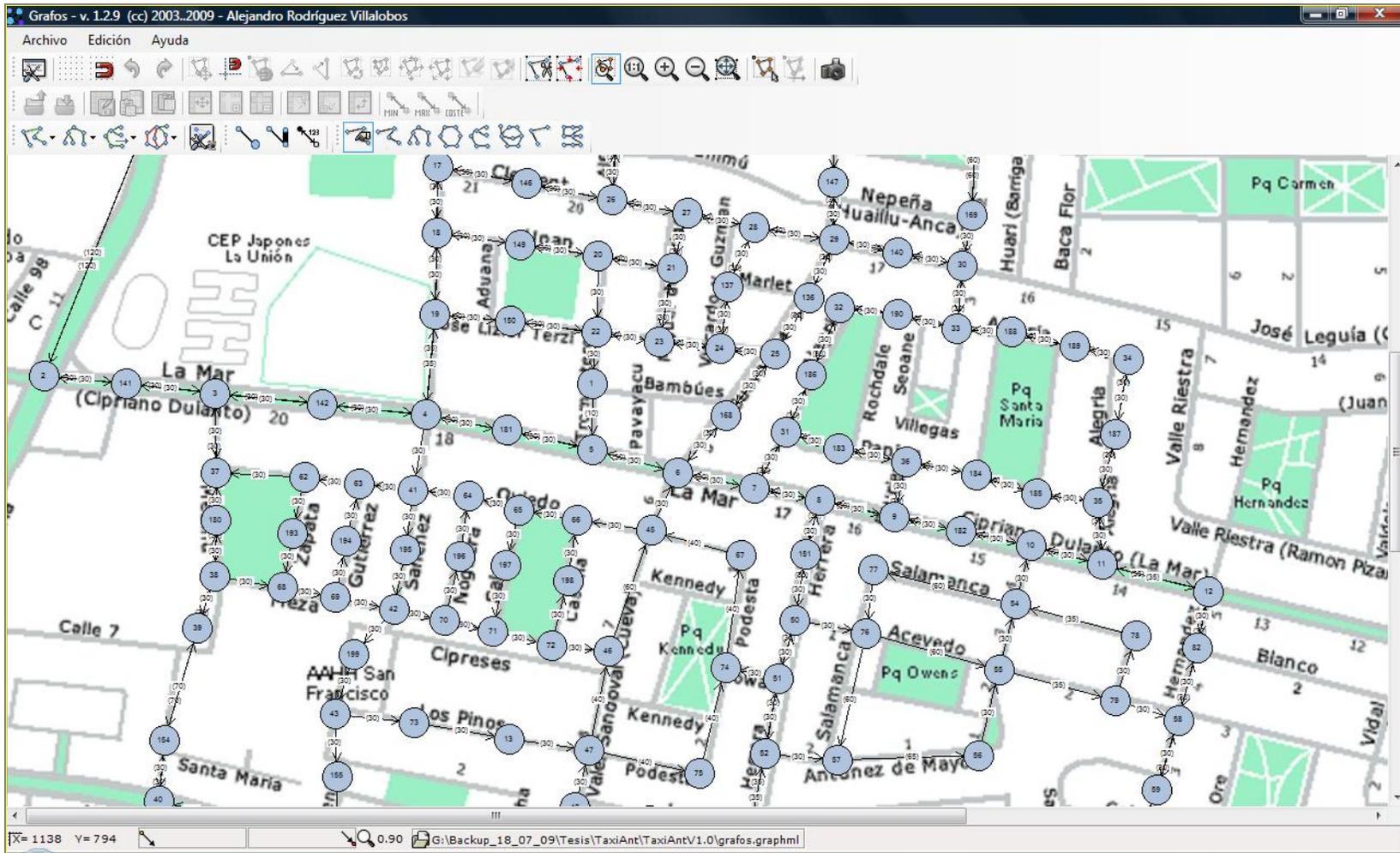
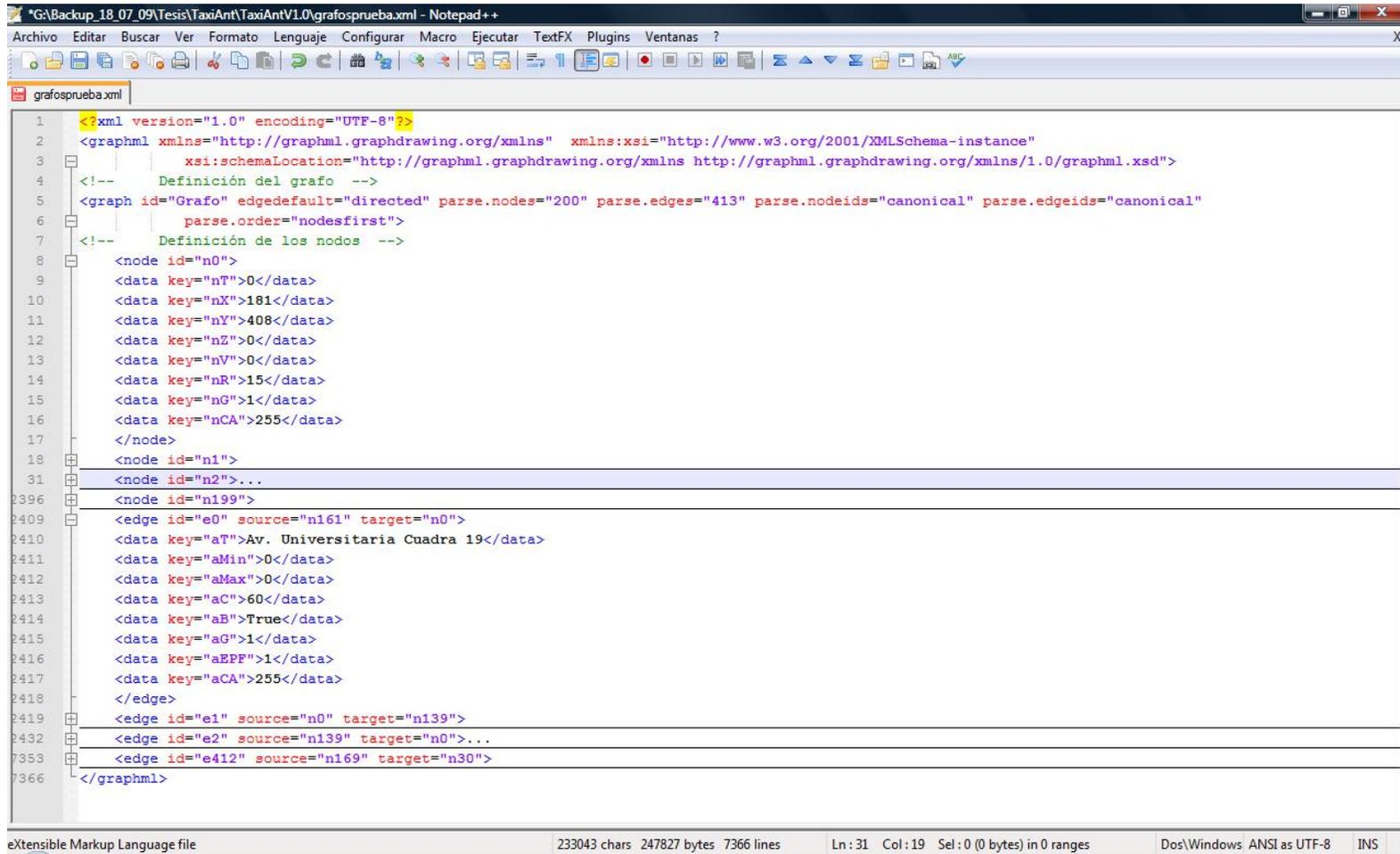


Figura E.2 Software Grafos.

1.2.3. GrahML File Format



```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <graphml xmlns="http://graphml.graphdrawing.org/xmlns" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3      xsi:schemaLocation="http://graphml.graphdrawing.org/xmlns http://graphml.graphdrawing.org/xmlns/1.0/graphml.xsd">
4      <!-- Definición del grafo -->
5      <graph id="Grafo" edgedefault="directed" parse.nodes="200" parse.edges="413" parse.nodeids="canonical" parse.edgeids="canonical"
6          parse.order="nodesfirst">
7          <!-- Definición de los nodos -->
8          <node id="n0">
9              <data key="nT">0</data>
10             <data key="nX">181</data>
11             <data key="nY">408</data>
12             <data key="nZ">0</data>
13             <data key="nV">0</data>
14             <data key="nR">15</data>
15             <data key="nG">1</data>
16             <data key="nCA">255</data>
17         </node>
18         <node id="n1">
19             <data key="nT">0</data>
20             <data key="nX">181</data>
21             <data key="nY">408</data>
22             <data key="nZ">0</data>
23             <data key="nV">0</data>
24             <data key="nR">15</data>
25             <data key="nG">1</data>
26             <data key="nCA">255</data>
27         </node>
28         <node id="n2">...
29         <node id="n199">
30             <data key="nT">0</data>
31             <data key="nX">181</data>
32             <data key="nY">408</data>
33             <data key="nZ">0</data>
34             <data key="nV">0</data>
35             <data key="nR">15</data>
36             <data key="nG">1</data>
37             <data key="nCA">255</data>
38         </node>
39         <edge id="e0" source="n161" target="n0">
40             <data key="aT">Av. Universitaria Cuadra 19</data>
41             <data key="aMin">0</data>
42             <data key="aMax">0</data>
43             <data key="aC">60</data>
44             <data key="aB">True</data>
45             <data key="aG">1</data>
46             <data key="aEPF">1</data>
47             <data key="aCA">255</data>
48         </edge>
49         <edge id="e1" source="n0" target="n139">
50             <data key="aT">Av. Universitaria Cuadra 19</data>
51             <data key="aMin">0</data>
52             <data key="aMax">0</data>
53             <data key="aC">60</data>
54             <data key="aB">True</data>
55             <data key="aG">1</data>
56             <data key="aEPF">1</data>
57             <data key="aCA">255</data>
58         </edge>
59         <edge id="e2" source="n139" target="n0">...
60         <edge id="e412" source="n169" target="n30">
61             <data key="aT">Av. Universitaria Cuadra 19</data>
62             <data key="aMin">0</data>
63             <data key="aMax">0</data>
64             <data key="aC">60</data>
65             <data key="aB">True</data>
66             <data key="aG">1</data>
67             <data key="aEPF">1</data>
68             <data key="aCA">255</data>
69         </edge>
70     </graph>
71 </graphml>
  
```

eXtensible Markup Language file 233043 chars 247827 bytes 7366 lines Ln: 31 Col: 19 Sel: 0 (0 bytes) in 0 ranges Dos\Windows ANSI as UTF-8 INS

Figura E.3 Formato de archivo GraphML