Contenido

[CAPITULO I 2](#_Toc392679352)

[PERFIL 2](#_Toc392679353)

[INTRODUCCION 2](#_Toc392679354)

[ANTECEDENTES 2](#_Toc392679355)

[DESCRIPCION DEL PROBLEMA 3](#_Toc392679356)

[OBJETIVOS 4](#_Toc392679357)

[Objetivo General 4](#_Toc392679358)

[Objetivo Especifico 4](#_Toc392679359)

[JUSTIFICACION 5](#_Toc392679360)

[Mediante el presente proyecto se pretende mejorar y optimizar la atención al cliente ofreciendo un servicio más cómodo y rápido, tanto en la venta de boletos, como en el envío y recepción de encomiendas y cargas, también se tendrá un mejor control del personal que trabaja en la empresa. 5](#_Toc392679361)

[ALCANCE 5](#_Toc392679362)

[MODULO VIAJE 5](#_Toc392679363)

[MODULO DOCUMENTOS 5](#_Toc392679364)

[MODULO SISTEMA Y SEGURIDAD 5](#_Toc392679365)

[METODOLOGIA 6](#_Toc392679366)

[CAPITULO II 7](#_Toc392679367)

[MARCO TEORICO 7](#_Toc392679368)

[EMPRESA DE TRANSPORTES “EL CHAQUEÑO” 7](#_Toc392679369)

[HERRAMIENTAS CASE 7](#_Toc392679370)

[Rational Rose 8](#_Toc392679371)

[LENGUAJE DE PROGRAMACION 8](#_Toc392679372)

[Java 8](#_Toc392679373)

[GESTOR DE BASE DE DATOS 9](#_Toc392679374)

[MySQL 9](#_Toc392679375)

[CAPITULO III 10](#_Toc392679376)

[CAPTURA DE REQUISITOS 10](#_Toc392679377)

[INTRODUCCION 10](#_Toc392679378)

[REQUISITOS FUNCIONALES 10](#_Toc392679379)

[REQUISITOS NO FUNCIONALES 11](#_Toc392679380)

[IDENTIFICAR CLASES DE DOMINIO 11](#_Toc392679381)

[DIAGRAMA DE DOMINIO INICIAL 12](#_Toc392679382)

[IDENTIFICAR CASOS DE USO 12](#_Toc392679383)

[IDENTIFICAR ACTORES 12](#_Toc392679384)

[DESCRIPCION DE LOS CASOS DE USO 13](#_Toc392679385)

[**1.** **Caso de uso – INICIAR SESION** 13](#_Toc392679386)

[**2.** **Caso de uso - REGISTRAR BOLETOS** 14](#_Toc392679387)

[**3.** **Caso de uso - REGISTRAR ENCOMIENDA** 15](#_Toc392679388)

[**4.** **Caso de uso - REGISTRAR GIRO** 16](#_Toc392679389)

[**5.** **Caso de uso - REGISTRAR PLANILLA DE VIAJE** 17](#_Toc392679390)

[**6.** **Caso de uso - REGISTRAR CHOFER** 18](#_Toc392679391)

[**7.** **Caso de uso - REGISTRAR BUSES** 19](#_Toc392679392)

[**8.** **Caso de uso - GENERAR REPORTES** 20](#_Toc392679393)

[**9.** **Caso de uso - REGISTRAR USUARIOS** 21](#_Toc392679394)

[**10.** **Caso de uso - GESTIONAR RESPALDO** 22](#_Toc392679395)

[DIAGRAMA GENERAL DEL MODELO DE CASO DE USO 23](#_Toc392679396)

[CAPITULO IV 24](#_Toc392679397)

[ANALISIS DE SISTEMA 24](#_Toc392679398)

[ANALISIS DE ARQUITECURA 24](#_Toc392679399)

[DIAGRAMA DE PAQUETES 25](#_Toc392679400)

[DIAGRAMAS DE COLABORACION – ANALISIS 26](#_Toc392679401)

[**1.** **Diagrama de colaboración - INICIAR SESION** 26](#_Toc392679402)

[**2.** **Diagrama de colaboración - REGISTRAR BOLETOS** 26](#_Toc392679403)

[**3.** **Diagrama de colaboración - REGISTRAR ENCOMIENDA** 26](#_Toc392679404)

[**4.** **Diagrama de colaboración - REGISTRAR GIRO** 27](#_Toc392679405)

[**5.** **Diagrama de colaboración - REGISTRAR PLANILLA DE VIAJE** 27](#_Toc392679406)

[**6.** **Diagrama de colaboración - REGISTRAR CHOFERES** 27](#_Toc392679407)

[**7.** **Diagrama de colaboración - REGISTRAR BUSES** 28](#_Toc392679408)

[**8.** **Diagrama de colaboración - GENERAR REPORTES** 28](#_Toc392679409)

[**9.** **Diagrama de colaboración - REGISTRAR USUARIOS** 28](#_Toc392679410)

[**10.** **Diagrama de colaboración - GESTIONAR RESPALDO** 29](#_Toc392679411)

[CAPITULO V 30](#_Toc392679412)

[DIAGRAMAS FINALES 30](#_Toc392679413)

[DIAGRAMA DE CLASES 30](#_Toc392679414)

[DIAGRAMA DE SECUENCIA DE DISEÑO 31](#_Toc392679415)

[**1.** **Diagrama de secuencia - INICIAR SESION** 31](#_Toc392679416)

[**2.** **Diagrama de secuencia - REGISTRAR BOLETOS** 31](#_Toc392679417)

[**3.** **Diagrama de secuencia - REGISTRAR ENCOMIENDA** 32](#_Toc392679418)

[**4.** **Diagrama de secuencia - REGISTRAR GIRO** 32](#_Toc392679419)

[**5.** **Diagrama de secuencia - REGISTRAR PLANILLA DE VIAJE** 33](#_Toc392679420)

[**6.** **Diagrama de secuencia - REGISTRAR CHOFERES** 33](#_Toc392679421)

[**7.** **Diagrama de secuencia - REGISTRAR BUSES** 34](#_Toc392679422)

[**8.** **Diagrama de secuencia - GENERAR REPORTES** 34](#_Toc392679423)

[**9.** **Diagrama de secuencia - REGISTRAR USUARIOS** 35](#_Toc392679424)

[**10.** **Diagrama de secuencia - GESTIONAR RESPALDO** 35](#_Toc392679425)

[CAPITULO VI 36](#_Toc392679426)

[DESARROLLO DEL SISTEMA 36](#_Toc392679427)

[INTERFACES 36](#_Toc392679428)

[LOGIN 36](#_Toc392679429)

[PANEL DE ENCARGADO 36](#_Toc392679430)

[VENTA DE PASAJES 37](#_Toc392679431)

[ENCOMIENDAS 37](#_Toc392679432)

[PANEL DE ADMINISTRACION 38](#_Toc392679433)

[PANEL DE REGISTRO DE USUARIOS 38](#_Toc392679434)

[BASE DE DATOS 39](#_Toc392679435)

# CAPITULO I

# PERFIL

## INTRODUCCION

Desde hace mucho años atrás la información a sido el pilar fundamental para la toma de decisiones basado en un histórico de datos almacenados con el transcurrir el tiempo se fue mejorando la manipulando de información mediante inventario el cual se manejaba en hoja de cuadernos, además que la información resguardada proporciona un si fin de resultado útiles para la persona que manejan la misma.

Los sistemas de Información administrativa están volviéndose indispensables a gran velocidad para la planificación, la toma de decisiones y el control de información de datos. La velocidad y exactitud con que lo directivo pueden recibir información sobre lo que esta funcionando bien o lo que está funcionando mal determinaran, en gran medida, la eficacia que tendrá lo sistemas de control.

El transporte utilizando buses ha llegado a convertirse en unos de los principales medio de comunicación, el cual oferta diversos servicios, tales como envíos y recepción de encomienda pagadas o por pagar, venta y reserva de pasaje y otros, tales práctica pueden utilizarse en otro medio de transporte.

En un medio como el transporte es indispensable el resguardado y un buen manejo de la información, porque en base a esa se puede generar reporte en cualquier momento de estado actual de la empresa de transporte.

## ANTECEDENTES

la empresa el Transporte “EL CHAQUEÑO” se fundó el 1 de mayo de 1960 teniendo como propietario al señor Isaac Ortuño quien empezó a prestar servicio con dos Buses. En la actualidad tiene como propietario su hijo el señor Julio Ortuño que es dueño de 6 buses, lo cuales prestan servicios con nueves buses los cuales tres prestan servicio de Camiri a Santa Cruz y viceversa.

El servicio que presta es a la ciudad de Santa Cruz de la Sierra, Yacuiba, Monteagudo y a la ciudad de Sucre.

La oficina central se ubica en la Calle Comercio entre Av. Bolívar y Av. Bush, ubicándose en la actualidad en la terminal de Buses “ISAAC ORTUÑO” en la ciudad de Camiri, Provincia Cordillera del Departamento de Santa Cruz.

## DESCRIPCION DEL PROBLEMA

La empresa de Transportes “EL CHAQUEÑO” presenta los siguientes inconvenientes:

* Mantiene todos los datos de los viajes y pasajero en hojas tradicionales, pudiendo perderse o borrarse en cualquier momento.
* La búsqueda de ciertos datos es morosa, lo cual con lleva a que el encargado utilice tiempo extra en la búsqueda.
* La planilla de encomiendas se maneja en hojas convencionales, que puede ser sobre escrita, alterada o dañada, causando confusión.
* La elaboración de la planilla de viaje para entregar a la autoridad de transito es generada manualmente, lo que con lleva a usar aun más tiempo del que se puede dar a un cliente en espera.
* El control del personal tanto administrativo como los choferes, son controlados de manera informal.

## OBJETIVOS

### Objetivo General

Desarrollar un sistema de información de administración de Viaje en la empresa de Transporte “EL CHAQUEÑO” en la ciudad de Camiri.

### Objetivo Especifico

• Recopilar la información necesaria para conocer el funcionamiento y la actividad del estudio jurídico, usando entrevistas, encuestas.

* Definir los requerimientos del sistema, utilizando herramientas para la recolección de información, como ser entrevistas, encuestas, observaciones.

• Análisis de los requisitos del sistema, se procederá al análisis de estos requisitos, tratando de enfocar las soluciones informáticas que estén a la altura de estos requisitos.

• Diseñar un sistema de información que cumpla los requisitos, utilizando las herramientas del Proceso Unificado de Desarrollo de Software.

• Desarrollo del sistema de información y escritura de código, se utilizará un lenguaje de programación y un gestor de base de datos que pueda satisfacer los requerimientos del sistema.

• Pruebas y Simulación al sistema

## JUSTIFICACION

## Mediante el presente proyecto se pretende mejorar y optimizar la atención al cliente ofreciendo un servicio más cómodo y rápido, tanto en la venta de boletos, como en el envío y recepción de encomiendas y cargas, también se tendrá un mejor control del personal que trabaja en la empresa.

## ALCANCE

El sistema de Información de Administración de Viaje tendrá los siguientes modulo de trabajo.

## MODULO VIAJE

* GESTIONAR BOLETOS
* REGISTRAR ENCOMIENDA
* REGISTRAR GIRO
* REGISTRAR PLANILLA DE VIAJE

## MODULO DOCUMENTOS

* GESTIONAR ENCARGADOS
* GESTIONAR CHOFERES
* GESTIONAR BUSES
* GENERAR REPORTES

## MODULO SISTEMA Y SEGURIDAD

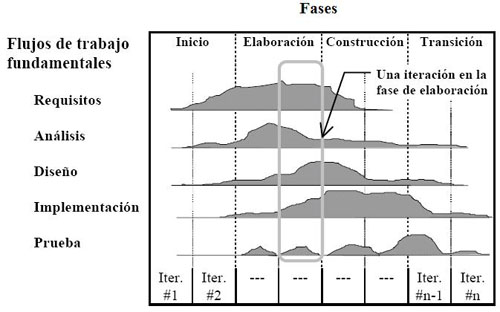
* INICIAR SESION
* GESTIONAR USUARIO
* GESTIONAR RESPALDO

## METODOLOGIA

Se utilizara el Proceso Unificado de Desarrollo de Software en la implementación del Sistema de Información para el Estudio Jurídico “Miguel Morón” este proceso tiene un conjunto de actividades necesarias para l transformar los requisitos del cliente en software.

El Proceso Unificado tiene las fases de inicio, elaboración, construcción, transición en las cuales se realizan iteraciones que abarcan los flujos de trabajo, requisitos, análisis, diseño implementación y prueba.

Se utilizara los artefactos que proporciona UML (Lenguaje Unificado de Modelado) para el modelado del sistema.



# 

# CAPITULO II

# MARCO TEORICO

## EMPRESA DE TRANSPORTES “EL CHAQUEÑO”

Los orígenes del siguiente negocio fue creado en el año 1960, teniendo como propietario a Don Isaac Ortuño un señor que tenia muy bien definidos sus ueños y metas por o cual decidió crecer en el ámbito del transporte con la integración de su propio negocio dedicado a prestar servicio de viajes a las diferentes ciudades.

Poco a poco su negocio fue creciendo llegando a comprar mas buses y en el año 2005 ya tenia éxito rotundo en el ámbito del transporte camireño dándose a conocer como uno de los principales buses de Camiri que brinda sus servicios de viajes a las ciudades de Sure, Monteagudo, Santa Cruz y Viceversa

## HERRAMIENTAS CASE

Las herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y dinero. Estas herramientas no pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, calculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.

Sistema de software que intenta proporcionar ayuda automatizada a las actividades del proceso de software.

En tal caso para el desarrollo e implementación del Sistema para el Estudio Jurídico “Miguel Moron” se utilizara la siguiente herramienta case:

### Rational Rose

Rational Software es actualmente conocida como una familia del software de IBM para el despliegue, diseño, construcción, pruebas, administración de proyectos en el proceso de desarrollo de software.

## LENGUAJE DE PROGRAMACION

### Java

Java es un lenguaje de programación con el que podemos realizar cualquier tipo de programa. En la actualidad es un lenguaje muy extendido y cada vez cobra más importancia tanto en el ámbito de Internet como en la informática en general. Está desarrollado por la compañía Sun Microsystems con gran dedicación y siempre enfocado a cubrir las necesidades tecnológicas más punteras.

Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Eso quiere decir que si hacemos un programa en Java podrá funcionar en cualquier ordenador del mercado. Es una ventaja significativa para los desarrolladores de software, pues antes tenían que hacer un programa para cada sistema operativo, por ejemplo Windows, Linux, Apple, etc. Esto lo consigue porque se ha creado una Máquina de Java para cada sistema que hace de puente entre el sistema operativo y el programa de Java y posibilita que este último se entienda perfectamente.

La independencia de plataforma es una de las razones por las que Java es interesante para Internet, ya que muchas personas deben tener acceso con ordenadores distintos. Pero no se queda ahí, Java está desarrollándose incluso para distintos tipos de dispositivos además del ordenador como móviles, agendas y en general para cualquier cosa que se le ocurra a la industria.

## GESTOR DE BASE DE DATOS

### MySQL

MySQL es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. Así, MySQL no es más que una aplicación que permite gestionar archivos llamados de bases de datos.

Existen muchos tipos de bases de datos, desde un simple archivo hasta sistemas relacionales orientados a objetos. MySQL, como base de datos relacional, utiliza multiples tablas para almacenar y organizar la información. MySQL fue escrito en C y C++ y destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interactuación con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos.

También es muy destacable, la condición de open source de MySQL, que hace que su utilización sea gratuita e incluso se pueda modificar con total libertad, pudiendo descargar su código fuente. Esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones, para hacer de MySQL una de las herramientas más utilizadas por los programadores orientados a Internet.

# CAPITULO III

# CAPTURA DE REQUISITOS

## INTRODUCCION

La captura de requisitos es el pilar fundamental para el desarrollo de un sistema, por ser la base, el inicio de un trabajo que ir evolucionando según los resultados de este flujo de trabajo, nos ayuda a encontrar, priorizar, detallar y estructurar el modelo de los casos de uso. Obteniendo resultados según requisitos funcionales de los usuarios del software.

## REQUISITOS FUNCIONALES

La captura de los requisitos funcionales se realiza por medio de entrevistas al personal de la empresa de transportes “El Chaqueño” de acuerdo a la actividad que realiza cada persona (actor), y a las necesidades de cada actor tiene el desempeño de su actividad laboral. A Continuación se presenta la lista de requisitos:

|  |  |
| --- | --- |
| **REQUISITOS** | **ACTOR** |
| INICIAR SESIÓN | TODOS |
| REGISTRAR BOLETOS | ENCARGADO |
| REGISTRAR ENCOMIENDA | ENCARGADO |
| REGISTRAR GIRO | ENCARGADO |
| REGISTRAR PLANILLA DE VIAJE | ENCARGADO |
| REGISTRAR CHOFERES | GERENTE |
| REGISTRAR BUSES | GERENTE |
| GENERAR REPORTES | GERENTE |
| REGISTRAR USUARIOS | ADMINISTRADOR |
| GESTIONAR RESPALDO | ADMINISTRADOR |

## REQUISITOS NO FUNCIONALES

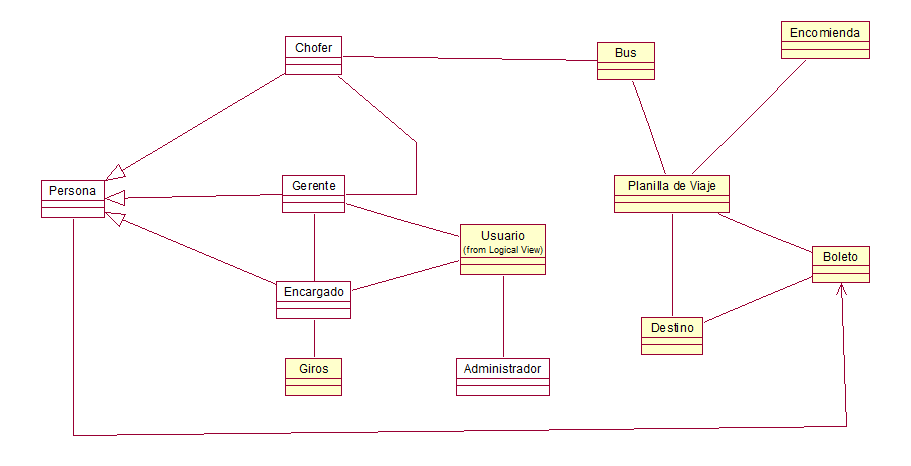
Los usuarios del sistema requieren de un software que cumpla con los requisitos no funcionales; los cuales se describen a continuación:

* El sistema debe funcionar correctamente para proporcionar a los usuarios eficiencia en el registro de datos y recuperación de datos en reportes (corrección).
* El software debe poseer fiabilidad para asegurar la veracidad de los datos especialmente los datos que comprenden datos personales y documentos de los clientes del estudio jurídico (fiabilidad).
* Debe existir eficiencia en el sistema realizando creación dinámica de formularios para ahorrar recursos de memoria (eficiencia).
* El software debe ser desarrollado con una documentación que pueda ser entendida para realizar cambios o mejoras que se requieran en un futuro; el sistema debe ser desarrollado con un estándar de codificación (mantenimiento).
* Realizar un control de usuario para el ingreso al sistema según su perfil de usuario.

## IDENTIFICAR CLASES DE DOMINIO

En la base de requisitos encontrados se procede a identificar las clases del dominio del sistema las cuales son backup,

## DIAGRAMA DE DOMINIO INICIAL



## IDENTIFICAR CASOS DE USO

Los casos de uso del sistema fueron encontrados según los requisitos de los diferentes usuarios del sistema en la base a los datos que requieren guardar o adquirir, los cuales se detallan en el cuadro.

## IDENTIFICAR ACTORES

Los actores que se encontraron en el desarrollo del software se especifica a continuación:

* **PERSONA.-** Es la agrupación de actores, o la minima expresión de ellos.
* **ENCARGADO.-** Es el encargado de la recepción de archivos del abogado, al mismo tiempo tiene la función de registrar lo que este a su alcance.
* **GERENTE.-** Tiene el rol fundamental en la toma de decisiones de las actividades dentro de la empresa de transporte.
* **ADMINISTRADOR.-** El administrador tiene el rol de administrar el sistema siendo la única persona que tiene acceso a sistema**.**

## DESCRIPCION DE LOS CASOS DE USO

1. **Caso de uso – INICIAR SESION**

****

|  |  |
| --- | --- |
| **CASO DE USO** | Iniciar Sesion |
| **ACTORES** | Todos |
| **ACTOR INICIADOR** | Todos |
| **PROPOSITO** | Por seguridad para acceder al sistema se deberá acreditar atreves de un Log In |
| **RESUMEN** | Permite realizar el control de los usuarios tanto de los administradores como de los encargados |
| **FLUJO DE TRABAJO** | 1. Ingresar Login y Password. 2. El sistema procede a la validación de los datos. 3. El usuario accede al sistema con los respectivos privilegios asignados. |

1. **Caso de uso - REGISTRAR BOLETOS**

****

|  |  |
| --- | --- |
| **CASO DE USO** | Registrar Boletos |
| **ACTORES** | Encargado |
| **ACTOR INICIADOR** | Encargado |
| **PROPOSITO** | Controlara la asignación de boletos de pasajes a los distintos pasajeros |
| **RESUMEN** | El encargado podrá registrar boletos de viaje, cada vez que así lo desee. |
| **FLUJO DE TRABAJO** | 1. Registrar, Se presenta al encargado un formulario con la información que debe ser llenada para la venta de un boleto de viaje. 2. Modificar, permite que el Encargado pueda cambiar la fecha de viaje, el asiento, previas condiciones de políticas de la empresa. 3. Eliminar, permitirá eliminar un boleto por un determinado motivo. |

1. **Caso de uso - REGISTRAR ENCOMIENDA**

****

|  |  |
| --- | --- |
| **CASO DE USO** | Registrar Encomiendas |
| **ACTORES** | Encargado |
| **ACTOR INICIADOR** | Encargado |
| **PROPOSITO** | Crear un Registro sobre las encomiendas que se envían. |
| **RESUMEN** | El Encargado antes de enviar todos los datos procederá a la verificación de ellos. Porque por políticas de la empresa solo se permitirá añadir registros y no modificarlos. |
| **FLUJO DE TRABAJO** | 1. Registrar Encomienda, se debe llenar los datos necesarios en el formulario, y verificar si todos los datos son correctos. 2. Luego enviar y guardar los datos |

1. **Caso de uso - REGISTRAR GIRO**

****

|  |  |
| --- | --- |
| **CASO DE USO** | Registrar Giro |
| **ACTORES** | Encargado |
| **ACTOR INICIADOR** | Encargado |
| **PROPOSITO** | Crear un Registro sobre los Giros que se envían y reciben. |
| **RESUMEN** | El Encargado antes de enviar todos los datos procederá a la verificación de ellos. Porque por políticas de la empresa solo se permitirá añadir registros y no modificarlos. |
| **FLUJO DE TRABAJO** | 1. Registrar Giro, se debe llenar los datos necesarios en el formulario, y verificar si todos los datos son correctos. 2. Luego enviar y guardar |

1. **Caso de uso - REGISTRAR PLANILLA DE VIAJE**

****

|  |  |
| --- | --- |
| **CASO DE USO** | Registrar Planilla de Viaje |
| **ACTORES** | Encargado |
| **ACTOR INICIADOR** | Encargado |
| **PROPOSITO** | Se crea una lista de pasajeros de un bus determinado, para llevar un mejor control y organización. |
| **RESUMEN** | El Encargado utilizara esta planilla para enviarlas a las autoridades de tránsito y administración de la terminal, además para llevar un control interno de los pasajeros viajantes |
| **FLUJO DE TRABAJO** | 1. El Encargado programara el viaje. 2. Se llenan los campos de boletos, encomiendas y giros para ese viaje. 3. Se procede a registrar la planilla de viaje |

1. **Caso de uso - REGISTRAR CHOFER**

****

|  |  |
| --- | --- |
| **CASO DE USO** | Registrar Chofer |
| **ACTORES** | Gerente |
| **ACTOR INICIADOR** | Gerente |
| **PROPOSITO** | El Gerente estará a cargo añadir y actualizar datos de los empleados de la empresa. |
| **RESUMEN** | Utilizara este formulario para ingresar y modificar información relevante de los choferes de buses |
| **FLUJO DE TRABAJO** | 1. Solicitar datos al chofer 2. Registrar, utilizando el formulario indicado para esto. 3. Enviar y guardar 4. Modificar y actualizar datos cuando sea necesario. |

1. **Caso de uso - REGISTRAR BUSES**

****

|  |  |
| --- | --- |
| **CASO DE USO** | Registrar Bus |
| **ACTORES** | Gerente |
| **ACTOR INICIADOR** | Gerente |
| **PROPOSITO** | El Gerente estará a cargo añadir y actualizar datos de los buses con los que cuenta la empresa. |
| **RESUMEN** | Utilizara este formulario para ingresar y modificar información relevante de los buses |
| **FLUJO DE TRABAJO** | 1. Registrar, utilizando el formulario indicado para esto. 2. Enviar y guardar 3. Modificar y actualizar datos cuando sea necesario. |

1. **Caso de uso - GENERAR REPORTES**

****

|  |  |
| --- | --- |
| **CASO DE USO** | Generar Reportes |
| **ACTORES** | Gerente |
| **ACTOR INICIADOR** | Gerente |
| **PROPOSITO** | El Gerente estará a cargo de la toma de decisiones y llevar el control de la empresa para ello necesitara los reportes precisos del área que necesita. |
| **RESUMEN** | Utilizara este formulario para obtener información del sistema sobre la empresa |
| **FLUJO DE TRABAJO** | 1. Proveerá los datos necesarios para obtener los reportes que desee. 2. Generara los reportes 3. Mostrara los reportes 4. Imprimirá los Reportes a través de una impresora de ser necesario |

1. **Caso de uso - REGISTRAR USUARIOS**

****

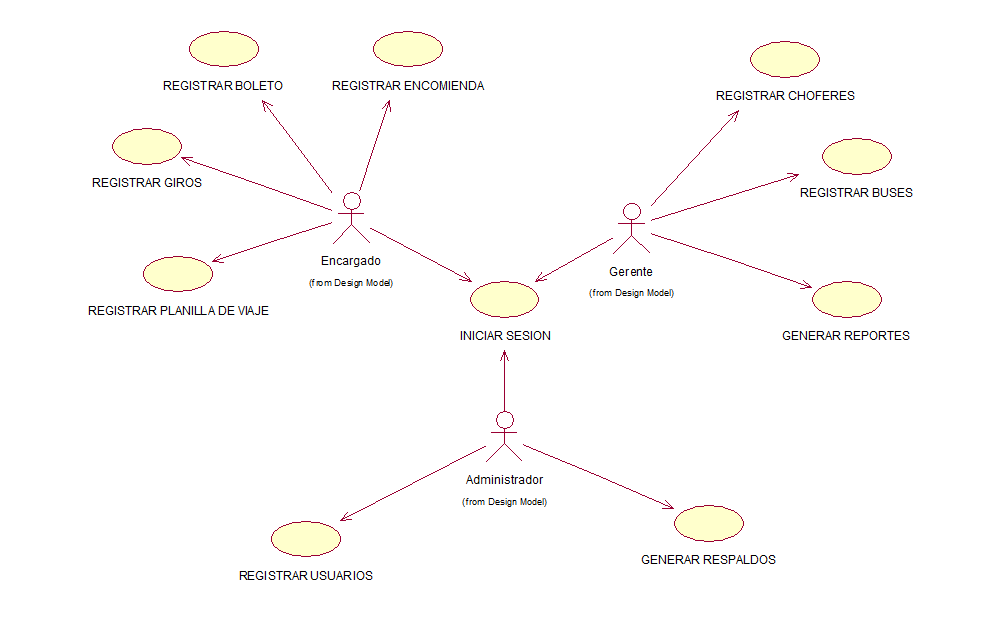
|  |  |
| --- | --- |
| **CASO DE USO** | Registrar Usuarios |
| **ACTORES** | Administrador |
| **ACTOR INICIADOR** | Administrador |
| **PROPOSITO** | El Administrador será el encargado de asignar los privilegios ya sea de Administrador o de Encargado a los usuarios de sistema. |
| **RESUMEN** | Es iniciado por el Administrador, cada vez que desee añadir un nuevo usuario y asignar privilegios que el Administrador no puede. |
| **FLUJO DE TRABAJO** | 1. Registrar, Se presenta al Administrador Un formulario para poder registrar nuevos usuarios con sus privilegios adecuados. 2. Modificar, el Administrador Podrá modificar datos y permisos relevantes sobre los usuarios del sistema. 3. Eliminar, el Administrador podrá inhabilitar a usuarios indebidos. |

1. **Caso de uso - GESTIONAR RESPALDO**

****

|  |  |
| --- | --- |
| **CASO DE USO** | Registrar Usuarios |
| **ACTORES** | Administrador |
| **ACTOR INICIADOR** | Administrador |
| **PROPOSITO** | El Administrador será el encargado de asignar los privilegios ya sea de Administrador o de Encargado a los usuarios de sistema. |
| **RESUMEN** | Es iniciado por el Administrador, cada vez que desee añadir un nuevo usuario y asignar privilegios que el Administrador no puede. |
| **FLUJO DE TRABAJO** | 1. Registrar, Se presenta al Administrador Un formulario para poder registrar nuevos usuarios con sus privilegios adecuados. 2. Modificar, el Administrador Podrá modificar datos y permisos relevantes sobre los usuarios del sistema. 3. Eliminar, el Administrador podrá inhabilitar a usuarios indebidos. |

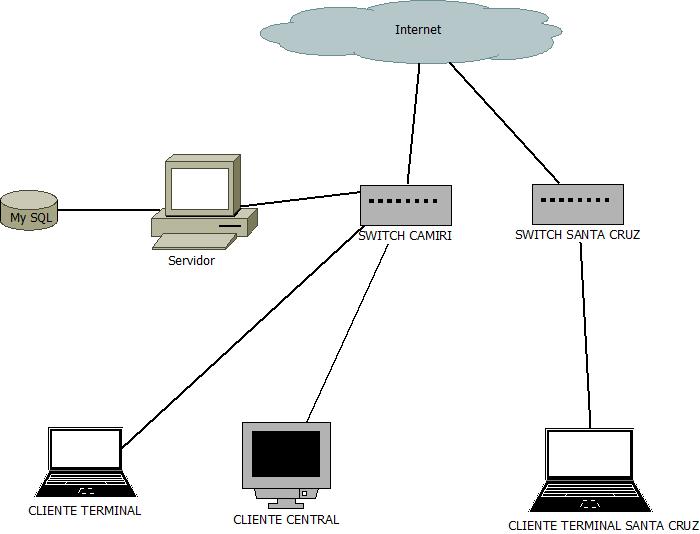
## DIAGRAMA GENERAL DEL MODELO DE CASO DE USO



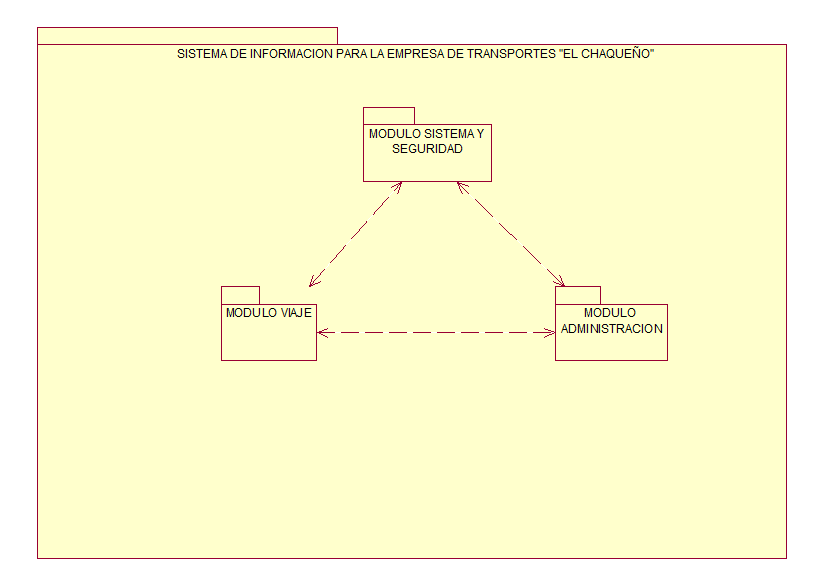
# CAPITULO IV

# ANALISIS DE SISTEMA

## ANALISIS DE ARQUITECURA



## DIAGRAMA DE PAQUETES



## DIAGRAMAS DE COLABORACION – ANALISIS

1. **Diagrama de colaboración - INICIAR SESION**

****

1. **Diagrama de colaboración - REGISTRAR BOLETOS**

****

1. **Diagrama de colaboración - REGISTRAR ENCOMIENDA**

****

1. **Diagrama de colaboración - REGISTRAR GIRO**

****

1. **Diagrama de colaboración - REGISTRAR PLANILLA DE VIAJE**

****

1. **Diagrama de colaboración - REGISTRAR CHOFERES**

****

1. **Diagrama de colaboración - REGISTRAR BUSES**

****

1. **Diagrama de colaboración - GENERAR REPORTES**

****

1. **Diagrama de colaboración - REGISTRAR USUARIOS**

****

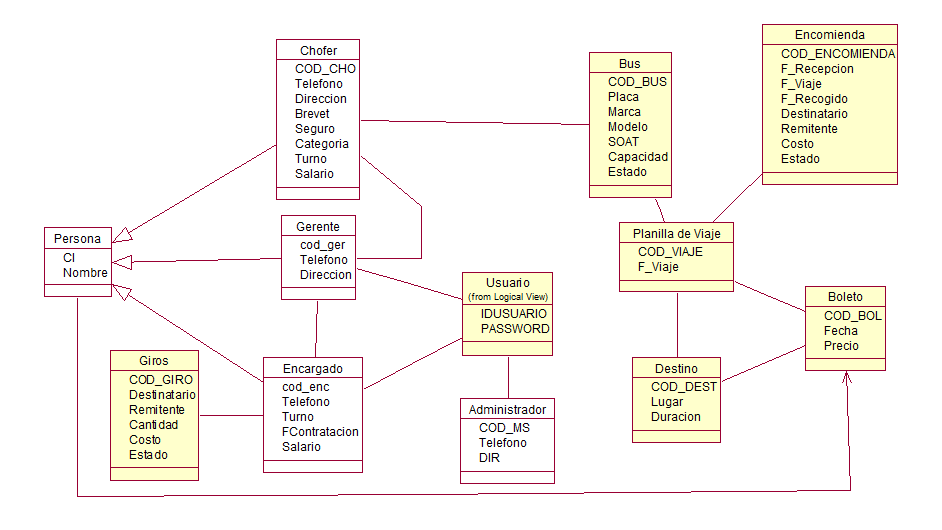
1. **Diagrama de colaboración - GESTIONAR RESPALDO**

****

# CAPITULO V

# DIAGRAMAS FINALES

# DIAGRAMA DE CLASES



# DIAGRAMA DE SECUENCIA DE DISEÑO

1. **Diagrama de secuencia - INICIAR SESION**

****

1. **Diagrama de secuencia - REGISTRAR BOLETOS**

****

1. **Diagrama de secuencia - REGISTRAR ENCOMIENDA**

****

1. **Diagrama de secuencia - REGISTRAR GIRO**

****

1. **Diagrama de secuencia - REGISTRAR PLANILLA DE VIAJE**

****

1. **Diagrama de secuencia - REGISTRAR CHOFERES**

****

1. **Diagrama de secuencia - REGISTRAR BUSES**

****

1. **Diagrama de secuencia - GENERAR REPORTES**

****

1. **Diagrama de secuencia - REGISTRAR USUARIOS**

****

1. **Diagrama de secuencia - GESTIONAR RESPALDO**

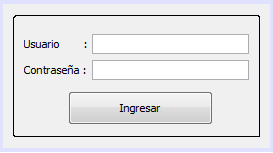
****

# CAPITULO VI

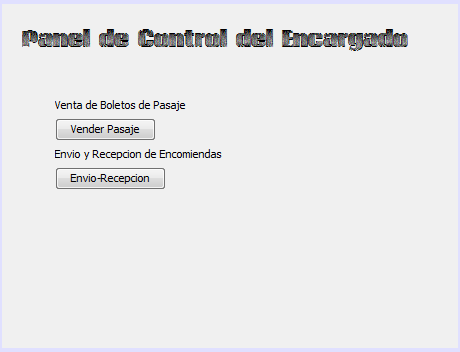
## DESARROLLO DEL SISTEMA

## INTERFACES

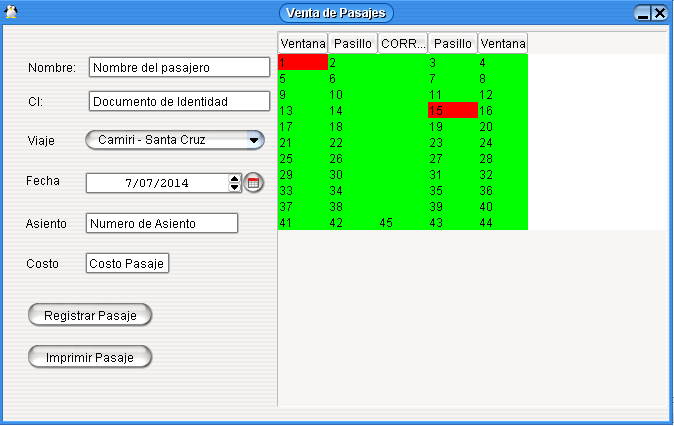
### LOGIN



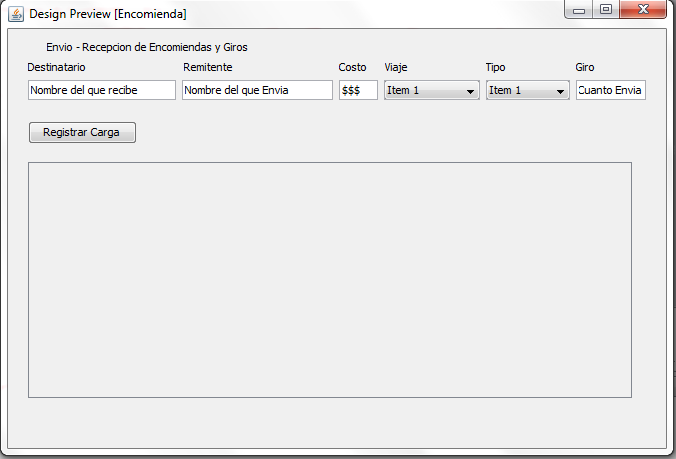
### PANEL DE ENCARGADO



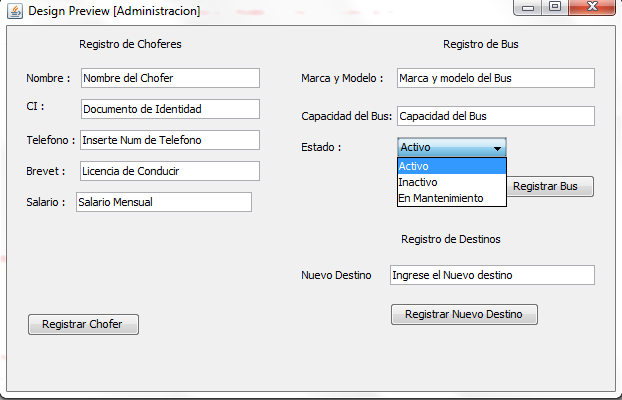
### VENTA DE PASAJES



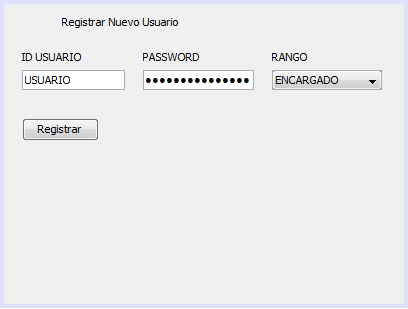
### ENCOMIENDAS



### PANEL DE ADMINISTRACION



### PANEL DE REGISTRO DE USUARIOS



## BASE DE DATOS

CREATE DATABASE /\*!32312 IF NOT EXISTS\*/`sistemadebuses`

USE `sistemadebuses`;

/\*Table structure for table `backup` \*/

DROP TABLE IF EXISTS `backup`;

CREATE TABLE `backup` (

`id\_back` int(11) NOT NULL AUTO\_INCREMENT,

`fecha` timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

`SQL` text NOT NULL,

PRIMARY KEY (`id\_back`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=latin1;

/\*Table structure for table `boleto` \*/

DROP TABLE IF EXISTS `boleto`;

CREATE TABLE `boleto` (

`id\_boleto` int(11) NOT NULL AUTO\_INCREMENT,

`ci` varchar(15) DEFAULT NULL,

`id\_viaje` int(11) DEFAULT NULL,

`costo` float DEFAULT NULL,

`asiento` int(2) NOT NULL,

PRIMARY KEY (`id\_boleto`),

KEY `FK\_boleto\_viaje` (`id\_viaje`),

KEY `FK\_boleto\_persona` (`ci`),

CONSTRAINT `FK\_boleto\_persona` FOREIGN KEY (`ci`) REFERENCES `persona` (`ci`) ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `FK\_boleto\_viaje` FOREIGN KEY (`id\_viaje`) REFERENCES `viaje` (`id\_viaje`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=latin1;

/\*Table structure for table `bus` \*/

DROP TABLE IF EXISTS `bus`;

CREATE TABLE `bus` (

`id\_bus` int(11) NOT NULL AUTO\_INCREMENT,

`placa` varchar(8) NOT NULL,

`marca/modelo` varchar(30) NOT NULL,

`capacidad` int(2) NOT NULL,

`estado` int(1) NOT NULL,

PRIMARY KEY (`id\_bus`)

) ENGINE=InnoDB AUTO\_INCREMENT=2 DEFAULT CHARSET=latin1;

/\*Table structure for table `chofer` \*/

DROP TABLE IF EXISTS `chofer`;

CREATE TABLE `chofer` (

`id\_chofer` int(11) NOT NULL AUTO\_INCREMENT,

`ci` varchar(15) NOT NULL,

`salario` float DEFAULT NULL,

`telefono` varchar(15) DEFAULT NULL,

`brevet` varchar(15) DEFAULT NULL,

PRIMARY KEY (`id\_chofer`,`ci`),

KEY `FK\_chofer\_persona` (`ci`),

CONSTRAINT `FK\_chofer\_persona` FOREIGN KEY (`ci`) REFERENCES `persona` (`ci`) ON DELETE CASCADE ON UPDATE NO ACTION

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=latin1;

/\*Table structure for table `destino` \*/

DROP TABLE IF EXISTS `destino`;

CREATE TABLE `destino` (

`id\_destino` int(2) NOT NULL AUTO\_INCREMENT,

`Descripcion` varchar(30) DEFAULT NULL,

PRIMARY KEY (`id\_destino`)

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=latin1;

/\*Table structure for table `encomienda` \*/

DROP TABLE IF EXISTS `encomienda`;

CREATE TABLE `encomienda` (

`id\_encomienda` int(11) NOT NULL AUTO\_INCREMENT,

`remitente` varchar(30) NOT NULL,

`destinatario` varchar(30) NOT NULL,

`id\_viaje` int(11) DEFAULT NULL,

`costo` float NOT NULL,

`tipo` int(2) NOT NULL,

`estado` int(1) NOT NULL,

`giro` int(6) DEFAULT NULL,

PRIMARY KEY (`id\_encomienda`),

KEY `FK\_encomienda` (`id\_viaje`),

CONSTRAINT `FK\_encomienda` FOREIGN KEY (`id\_viaje`) REFERENCES `viaje` (`id\_viaje`) ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=latin1;

/\*Table structure for table `persona` \*/

DROP TABLE IF EXISTS `persona`;

CREATE TABLE `persona` (

`ci` varchar(15) NOT NULL,

`nombre` varchar(30) DEFAULT NULL,

PRIMARY KEY (`ci`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/\*Table structure for table `usuario` \*/

DROP TABLE IF EXISTS `usuario`;

CREATE TABLE `usuario` (

`id\_user` varchar(15) NOT NULL,

`password` varchar(40) NOT NULL,

`ci` varchar(15) DEFAULT NULL,

`rango` int(2) DEFAULT NULL,

PRIMARY KEY (`id\_user`),

KEY `FK\_usuario` (`ci`),

CONSTRAINT `FK\_usuario` FOREIGN KEY (`ci`) REFERENCES `persona` (`ci`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/\*Table structure for table `viaje` \*/

DROP TABLE IF EXISTS `viaje`;

CREATE TABLE `viaje` (

`id\_viaje` int(11) NOT NULL AUTO\_INCREMENT,

`id\_destino` int(2) NOT NULL,

`fecha` timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

`observacion` text,

`id\_chofer` int(11) NOT NULL,

`id\_bus` int(11) NOT NULL,

PRIMARY KEY (`id\_viaje`),

KEY `FK\_viaje\_destino` (`id\_destino`),

KEY `FK\_viaje\_bus` (`id\_bus`),

KEY `FK\_viaje\_chofer` (`id\_chofer`),

CONSTRAINT `FK\_viaje\_bus` FOREIGN KEY (`id\_bus`) REFERENCES `bus` (`id\_bus`) ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `FK\_viaje\_chofer` FOREIGN KEY (`id\_chofer`) REFERENCES `chofer` (`id\_chofer`) ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `FK\_viaje\_destino` FOREIGN KEY (`id\_destino`) REFERENCES `destino` (`id\_destino`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=latin1;