



**Universidad Tecnológica del Centro de Veracruz**

**Programa Educativo: Ingeniería en Tecnologías de la Información**

**Reporte para obtener título de Ingeniería en Tecnologías de la  
Información**

**Proyecto de estadia realizado en la empresa Omnibus Centro Sur  
S.A de C.V.**

**Nombre del proyecto: Control de Tiempos en Trayectos**

**Presenta: Emma Vázquez Flores**



**Universidad Tecnológica del Centro de Veracruz**

**Programa Educativo: Tecnologías de la Información y  
Comunicación**

**Asesor Industrial: C.P Francisco Nieto Hernández.**

**Asesor Académico: MCC Lorena Alcudia Chagala**

**Presenta: Emma Vázquez Flores**

## RESUMEN

Aplicación de Escritorio Control de Tiempos en Trayectos es un proyecto que se propone después del análisis de requerimientos ayudando con el manejo de información de una manera más fácil y eficiente, utilizando conexión a base de datos, aplicando herramientas de Tecnologías de la Información como son PostgreSQL para el desarrollo de la base de datos. JAVAFXScene Builder 2.0 para la creación de las interfaces de la aplicación, en ésta no se maneja código. EclipseMars como entorno de desarrollo de software donde se maneja código para darle funcionalidad a la interfaz creada en JAVAFXScene Builder 2.0. Jaspersoft Studio 2.0 que es una herramienta de diseño de informes basada en Eclipse.

Con el uso de estas herramientas se logra desarrollar una Aplicación de Escritorio de calidad, eficiente y segura utilizando la metodología Ágile UP.

Esta aplicación va a proporcionar muchos beneficios a la empresa Omnibus Centro Sur, S.A de C.V., como es llevar de manera controlada la información del área de operaciones, seguridad en la información, información actualizada, también cuenta con reportes e inicio de sesión.

## Contenido

<b>CAPÍTULO 1 ANTECEDENTES</b> .....	5
1.1 Introducción .....	5
1.2 Justificación .....	7
1.3 Definición del problema .....	8
1.4 ANTECEDENTES DE LA EMPRESA. ....	10
1.4.1 Misión .....	10
1.4.2 Visión.....	10
1.4.3. Valores de la empresa.....	10
1.5 Objetivos .....	11
1.6 Alcances y limitaciones.....	11
1.6.1 Alcance .....	11
1.6.2 Limitaciones .....	13
<b>CAPÍTULO 2</b> .....	14
<b>Marco teórico</b> .....	14
<b>CAPÍTULO 3 METODOLOGÍA</b> .....	16
Metodología Ágile UP .....	16
3.1 Inicio.....	16
3.1.1 Definir el alcance del proyecto .....	17
3.1.2 Estimación de costes y Programación.....	19

3.1.3 Viabilidad (técnica, económica, social y operativa) del proyecto.....	19
3.1.4 Diagrama de caso de uso .....	21
3.2 Elaboración.....	37
3.1 interfaces.....	39
3.3 Construcción.....	42
CAPÍTULO 4. PROCEDIMIENTO .....	48
CAPÍTULO 5 EVALUACIÓN DE RESULTADOS.....	50
CAPÍTULO 7 REFERENCIAS.....	52
CAPÍTULO 8. ANEXOS.....	54

## Tabla de ilustraciones

Ilustración 1. Casos de uso registro del cliente .....	23
Ilustración 2 Caso de uso Iniciar sesión.....	24
Ilustración 3. Caso de uso gestionar terminales .....	27
Ilustración 4. Caso de uso gestionar unidades .....	29
Ilustración 5. Caso de uso gestionar operadores .....	31
Ilustración 6. Caso de uso gestionar recorridos.....	34
Ilustración 7. Caso de uso generar reportes .....	36
Ilustración 8. Diagrama del sistema .....	37
Ilustración 9. Diagrama de clases .....	38
Ilustración 10. Modelo relacional.....	38
Ilustración 11. Interfaz inicio de sesión .....	39
Ilustración 12. Interfaz de menú principal .....	39
Ilustración 13. Interfaz de Administración de usuarios .....	40
Ilustración 14. Interfaz de Administración de operadores.....	40
Ilustración 15. Interfaz de Administración de unidades. ....	41
Ilustración 16. Interfaz de Control de tiempos .....	41
Ilustración 17. Interfaz de Administración de terminales. ....	42
Ilustración 18. Encarpetado MVC .....	42
Ilustración 19. Funcionalidad del inicio de sesión .....	43
Ilustración 20. Funcionalidad del menú.....	44
Ilustración 21. Funcionalidad del módulo administración de usuarios.....	44
Ilustración 22. Funcionalidad del módulo administración de operadores .....	45
Ilustración 23. Funcionalidad del módulo administración de unidades.....	46
Ilustración 24. Funcionalidad del módulo control de tiempos .....	46
Ilustración 25. Funcionalidad del módulo administración de terminales.....	47
Ilustración 26. Entrevista .....	54

Tabla 1. Tecnologías utilizadas.....	48
--------------------------------------	----

---

## **CAPÍTULO 1 ANTECEDENTES**

---

### **1.1 Introducción**

Actualmente las tecnologías de la información son esenciales en las organizaciones; se utilizan de manera interna y externa en las empresas, mejorando la habilidad de los administradores para controlar y coordinar las actividades de las organizaciones.

Utilizar tecnologías de la información se ha convertido en una prioridad para muchas organizaciones que requieren realizar sus procesos de una manera rápida, eficiente, confiable y fácil. En la actualidad los sistemas de información, no solo son la base de cualquier organización sino también ayudan a facilitar la toma de decisiones de manera oportuna.

Las aplicaciones de escritorio son aquellas aplicaciones de software que se instalan y/o ejecuta en un ordenador para facilitar el trabajo a los empleados, facilitando la administración de la información.

La necesidad de tener ordenado y actualizados los datos en las empresas es uno de los principales objetivos de las tecnologías de la información, debido a ello, en la empresa Omnibus Centro Sur, S.A de C.V de la ciudad de Córdoba, Veracruz se requiere desarrollar una aplicación de escritorio para el control de tiempos en trayectos en el área de operaciones, en donde se llevan los datos de manera poco controlada dado que se maneja la información mediante formatos impresos y libros de Excel, lo que provoca pérdida de tiempo a la hora de capturar datos y buscar información.

Este documento presenta la información de la siguiente manera:

Capítulo I. Contiene una breve introducción del documento en general.

Capítulo II. Describe a detalle la problemática de la empresa Omnibus Centro Sur, S.A de C.V.

Capítulo III. Esta parte describe el objetivo general y los específicos, que son la base principal para poder desarrollar el proyecto debido a que estos ayudarán a solucionar la problemática.

Capítulo IIV En este apartado se describe brevemente el alcance de la aplicación, así como sus limitaciones.

Capítulo V. Justifica el porqué del desarrollo de la aplicación de escritorio, los beneficios que se obtendrán si se implementa dicha aplicación.

Capítulo VI. Se presentan las alternativas de solución, que en este caso son dos alternativas, también describe el estado de la práctica y la metodología y tecnologías que se emplearían para su desarrollo.

Capítulo VII. Describe la solución propuesta, junto con las características, justificación de la misma, marco teórico, descripción de producto o entregables y el cronograma.

Capítulo VIII. Presenta las referencias bibliográficas, que son los sitios de donde se obtuvo parte de la información.

## **1.2 Justificación**

La empresa Organismo de formación y actualización requiere darse a conocer en distintos sectores. Los usuarios interesados deben acudir personalmente a la empresa para obtener dicha información causando así pérdida de tiempo. Es por ello que se crea un sitio web en el cual se encontrará información actualizada acerca de la empresa así como de sus servicios, proporcionando al usuario mayor comodidad al consultar este sitio web desde donde esté y cuando lo requiera, sin la necesidad de acudir a la empresa, siendo así los beneficiados ambas partes ya que habrá más organización para el empresario y mayor facilidad en la consulta de información para el usuario.

### **1.3 Definición del problema**

La empresa Omnibus Centro Sur, S.A. de C.V., empresa dedicada al transporte de pasajeros, en donde la información especialmente del área de operaciones, se lleva en formatos impresos y en libros de Microsoft Excel.

El formato impreso se llama Control de Tiempos en Trayectos, que contiene datos referentes a cada terminal, como son: ruta, fecha, hora de salida, hora de arribo, entre otros; cabe mencionar que la empresa cuenta con 50 unidades que hacen varias rutas al día, lo que genera varios formatos, aproximadamente 100 registros al día; los formatos impresos son llenados manualmente y después son enviados a la empresa.

Como se mencionó anteriormente, la empresa acumula varios formatos, posteriormente los datos son capturados en libros de Microsoft Excel llamado reporte “retardos de operadores” en los cuales se manejan datos como lo son: nombre del operador, unidades, terminales, entre otros; debido a que esta información se toma en cada terminal para llevar un control de retrasos, este proceso se vuelve muy tedioso y es necesario dedicar más tiempo, aproximadamente 3 horas diarias para capturar la información del formato control de tiempos en trayectos al reporte retardo de operadores, para poder realizar un solo concentrado de datos, cabe mencionar que con toda la información final se genera un reporte para el gerente, el cual es enviado vía correo electrónico.

El llevar reportes en libros de Excel genera demasiados archivos, lo cual dificulta la disponibilidad de información en caso de ser requerido, además de que si una persona accede al equipo de cómputo con malas intenciones puede hacer modificaciones en la información.

Haciendo mención a lo anterior y tomando en cuenta los recursos que dispone la empresa se propone una aplicación de escritorio con conexión a una base de datos, la cual permita

sustituir los libros de Excel para el Reporte de retardo de operadores, que a su vez permita agilizar la búsqueda de los datos atrasados, insertar la información, realizar consultas y visualizar reportes acerca de toda la información del área de operaciones.

## **1.4 ANTECEDENTES DE LA EMPRESA.**

### **1.4.1 Misión**

Somos un grupo corporativo de Autotransporte de pasajeros, cuya finalidad es brindar un servicio de transporte seguro y eficiente a nuestros usuarios. Solo así generaremos los ingresos económicos suficientes que nos permitan retribuir equitativamente el esfuerzo de todos sus aportantes (Trabajadores y Accionistas) la reinversión constante y nuestro posicionamiento en las comunidades a las que servimos.

### **1.4.2 Visión**

Constituir una Empresa sólida que brinde un servicio con calidad en beneficio de nuestros usuarios y de esta manera lograr la aceptación, consolidación, rentabilidad y crecimiento de nuestra empresa.

### **1.4.3. Valores de la empresa**

**Lealtad:** Atender y servir incondicionalmente las necesidades del Grupo ADIC, aceptando su participación en el bienestar y mejoramiento de los miembros del corporativo y de la sociedad.

**Honestidad:** Hablar con la verdad, expresando clara, directa y abiertamente su parecer, congruente con lo que hace y con lo que piensa; solidario y leal con las mejores causas del Grupo y sus miembros sin buscar recompensa alguna.

**Unidad familiar:** Promover la unidad en todos los miembros del Grupo ADIC y sus familias sin permitir que le dañe ningún interés sectario o personal.

## **1.5 Objetivos**

### **Objetivo general**

Desarrollar una aplicación que permita administrar la información del área de operaciones con el fin de agilizar el control de tiempos en trayectos de las unidades Omnibus aplicando las tecnologías de la información

### **Objetivos específicos**

- Establecer los requisitos funcionales y no funcionales para el desarrollo de la aplicación utilizando en estándar IEEE 830.
- Implementar técnicas de seguridad para mayor control de acceso a la información.
- Establecer la estructura de los reportes a generar para facilitar la toma de decisiones.
- Implementar patrones visuales para la generación de las interfaces de usuario dentro de la aplicación.
- Desarrollar los módulos necesarios para agilizar los procesos dentro de la organización.

## **1.6 Alcances y limitaciones**

### **1.6.1 Alcance**

Desarrollar una aplicación de escritorio con el objetivo de llevar la administración de la información en el área de operaciones, el tiempo establecido para el desarrollo de la aplicación es de 15 semanas.

La aplicación contará con lo siguiente:

- Módulo de Registro de usuarios. En este módulo administrar las personas asignadas a introducir los datos de tiempos y retardos de los operadores.

- Módulo de Inicio de sesión. Este módulo permitirá brindarle acceso a la aplicación solo al personal autorizado, por lo cual será necesario tener un nombre de usuario y una contraseña para poder acceder a la aplicación.
- Módulo de Terminales. Este módulo permitirá administrar las terminales pertenecientes a la empresa Omnibus Centro Sur, S.A de C.V.
- Módulo de unidades. Este módulo permitirá administra las unidades.
- Módulo de operadores. En este módulo se administraran los operadores de las unidades.
- Módulo de tiempos. Este es el módulo central del sistema, dado que permitirá capturar la información de las rutas y los tiempos en que cada operador realiza las mismas. En este módulo se concentran la información de los operadores y terminales.
- Módulo de generación de reportes. Se podrán generar los siguientes reportes:
  - Tiempos de recorrido por operador: En este reporte se podrá visualizar la el retraso de los operadores por fecha y terminal.
  - Rotación de operadores: Este reporte podrá mostrar la información de los operadores, así como unidad asignada.
  - Estatus de unidad: Este reporte podrá mostrar los datos de las unidades, así como la información del estatus de la misma.

## 1.6.2 Limitaciones

Para el uso de la aplicación se mencionan las siguientes restricciones y/o limitaciones:

- La aplicación será utilizada por el personal de la empresa Omnibus Centro Sur, S.A. de C.V. especialmente para el área de operaciones, mientras cuenten con una cuenta (usuario y contraseña) para poder tener acceso a la aplicación.
- Toda la información que se manejará durante la construcción de la aplicación será ficticia debido a la privacidad de la empresa. Posteriormente personal encargado de del área realizará la inserción de los datos.
- La aplicación se desarrollará exclusivamente para sistema operativo Windows.

---

## CAPÍTULO 2 MARCO TEÓRICO

---

### Marco teórico

En este capítulo se establece un marco teórico fundamental, en el cual se proporcionan las definiciones más importantes para la comprensión del presente proyecto de tesina.

Java: es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes

JavaFX Scene Builder: es una herramienta de diseño visual que permite a los usuarios diseñar rápidamente interfaces de usuario de aplicaciones JavaFX, sin necesidad de programación.

PostgreSQL: es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales.

Jaspersoft: Studio es un diseñador de informes gratuito de código abierto basado en Eclipse para JasperReports y JasperReports Server. Crear diseños muy sofisticados que contienen gráficos, imágenes, subinformes, tablas de contingencia y mucho más.

Requisito: Es la descripción de los servicios y restricciones.

Funcionalidad: Descripción de lo que el software debe hacer.

Prototipos: Un prototipo es un borrador de un producto potencial o de una parte del mismo. Es una simulación de los requisitos.



### Metodología Ágile UP

El Proceso Unificado Ágil (Agile UP) es un enfoque al desarrollo de software basado en el Rational Unified Process (RUP) de IBM. El ciclo de vida de Agile UP es serial en lo grande e iterativo en lo pequeño, liberando entregables incrementales en el tiempo. (Ambysoft Inc, 2015)

### Fases de Agile UP

Las fases son implementadas de una forma serial a lo largo de un proyecto de Agile UP.

Estas fases son:

1. Inicio
2. Elaboración
3. Construcción
4. Transición

#### 3.1 Inicio

El objetivo principal de la fase de iniciación es archivar el consenso de los interesados del proyecto en relación a los objetivos del proyecto para obtener el financiamiento.

Para salir de la etapa de Iniciación su equipo de terminar el hito de Objetivos del Ciclo de Vida (LCO). El principal aspecto es hacer que el equipo entienda el alcance del proyecto y el esfuerzo requerido y cómo los usuarios patrocinarán el proyecto.

### 3.1.1 Definir el alcance del proyecto

Para definir el alcance del proyecto fue necesario el uso de algunas técnicas de recolección de datos, como lo fue la entrevista, que fue realizada al contador Francisco Nieto Hernández, quien fue el primer contacto en la empresa, posteriormente al gerente del área de operaciones y finalmente a la jefa de inspección.

Dentro de las principales preguntas que se le hicieron al contador fueron:

¿A qué área se desarrollará la aplicación?

La aplicación que se desarrollará ¿en que será alojada?

¿Cuántas personas ocuparán la aplicación?

Dentro de la encuesta realizada al área de operaciones se encuentran:

¿Cómo se lleva a cabo el proceso de control de tiempos respecto a los trayectos?

¿Qué datos se necesitan para dicho proceso?

¿Generan algún reporte?

¿Qué datos lleva dicho reporte?

Posteriormente de la recolección de datos se definió el alcance.

Dentro del alcance del proyecto se acordó que la aplicación contaría con lo siguiente:

- Módulo de Registro de usuarios. En este módulo se administrará las personas asignadas a introducir los datos de tiempos y retardos de los operadores.
- Módulo de Inicio de sesión. Este módulo permitirá brindarle acceso a la aplicación solo al personal autorizado, por lo cual será necesario tener un nombre de usuario y una contraseña para poder acceder a la aplicación.

- Módulo de Terminales. Este módulo permitirá administrar las terminales pertenecientes a la empresa Omnibus Centro Sur, S.A de C.V.
- Módulo de unidades. Este módulo permitirá administrar de las unidades.
- Módulo de operadores. En este módulo se administraran los operadores de las unidades.
- Módulo de recorridos. Este es el módulo central del sistema, dado que permitirá capturar la información de las rutas y los tiempos en que cada operador realiza las mismas. En este módulo se concentran la información de las unidades, operadores y terminales.
- Módulo de generación de reportes. Se podrán generar los siguientes reportes:
  - Tiempos de recorrido por operador: Este reporte se podrá generar por rango de fecha y por rutas.
  - Rotación de operadores: Este reporte podrá mostrar la información de los operadores, así como unidad asignada y clase de unidad.
  - Administración de terminales: Este reporte podrá mostrar los datos de las terminales.

### **3.1.2 Estimación de costes y Programación**

Debido a que el proyecto Control de Tiempos en Trayectos es desarrollado como proyecto de estadía y con tecnologías gratuitas, el adquirir e instalar esta aplicación no tendrá costo; es por ello que la empresa Omnibus podrá utilizar esta aplicación sin tener que pagar por ello.

En esta etapa también se definieron las metodologías a utilizar, dichas tecnologías fueron:

- Java JavaFX Scene Builder 2.0
- Lenguaje de programación Java.
- Eclipse Mars
- PostgreSQL
- Jaspersoft Studio

### **3.1.3 Viabilidad (técnica, económica, social y operativa) del proyecto**

#### Viabilidad técnica

Para realizar la Aplicación de Escritorio Control de Tiempos en Proyectos existe la tecnología necesaria para desarrollar la aplicación como son PostgreSQL para realizar una base de datos, JavaFX Scene Builder 2.0 para diseñar las ventanas, eclipse para programar las funciones de la aplicación y JasperSoft Studio para el diseño y creación de los reportes. A esta tecnología es fácil acceder debido a que es software libre.

#### Viabilidad económica

La Aplicación se desarrolla como proyecto de estadía por ello no tendrá ningún costo y por las herramientas tecnológicas no hay inconveniente debido a que es software libre.

Una vez puesta en marcha la aplicación se capacitará al personal una semana para que puedan utilizar de manera correcta la aplicación y que puedan obtener buenos resultados.

Para la aplicación no existen proveedores alternativos debido a que esta aplicación fue desarrollada a medida de la empresa, es decir, de acuerdo a las necesidades de la empresa Omnibus Centro Sur.

#### Viabilidad social

Debido a que este proyecto es de beneficio a la empresa Omnibus Centro Sur, esta empresa no deberá gastar en recurso financiero para su desarrollo e implementación, dejando entendido a los interesados (al área de operaciones) que las herramientas a utilizar son totalmente gratis, es por ello que la empresa no aportará recursos financieros.

#### Viabilidad operativa

Debido a que va ser una aplicación de acuerdo a las necesidades de la empresa (software a la medida), los usuarios del área de operaciones harán uso de ella, otra de las razones por la cual los empleados le darán uso a la aplicación es porque es una aplicación fácil de usar e interactiva y con una interfaz amigable para los usuarios. Debido a que el personal recibirá capacitación será muy fácil usarla.

### 3.1.4 Diagrama de caso de uso

1. Caso de Uso: Registro de usuario.

#### 1.1 Breve descripción

El cliente debe registrarse para poder ingresar al sistema.

Flujo de eventos.

#### 1.2.1 Flujo básico

1.- El caso de uso inicia cuando el cliente solicita su registro al sistema.

2.- El sistema le muestra una interfaz donde se le muestra un formulario con información básica para su registro

3.- El cliente ingresa los datos que se le piden en el formulario.

4.- El cliente elige la opción 'Registrar' para enviar sus datos.

5.- El sistema valida que no quede ningún campo vacío y cumpla con lo especificado.

6.- El sistema valida los datos ingresados y genera un nuevo código de cliente con los datos ingresados y le asigna tipo de nivel (cliente).

7.- El caso de uso termina cuando el cliente queda registrado y se le muestra la primera interfaz del sistema.

Flujo alternativo.

1.- El cliente ya existe.

Si el sistema detecta que el cliente ya existe en el sistema, envía un mensaje "El usuario ya existe" y el flujo alternativo finaliza.

2.- Formulario con campos vacíos.

Cuando el cliente solicita enviar sus datos, si el sistema detecta que un campo de datos se encuentra vacío, envía un mensaje de alerta “El campo (nombre del campo) está vacío, ingrese los datos.” El flujo alternativo finaliza.

3.- Los datos ingresados no cumplen el formato establecido.

Cuando el cliente solicita enviar sus datos, si el sistema detecta que los datos no son válidos envía un mensaje de alerta “Favor de introducir datos correctos al formulario” y el flujo alternativo finaliza.

Requerimientos especiales

Ninguno.

1.4 Condiciones previas.

El solicitante no ha sido registrado como cliente del sistema.

Postcondiciones.

1.- El solicitante ha sido registrado, es cliente y puede hacer uso de las funciones del sistema.

Puntos de extensión.

Ninguno.

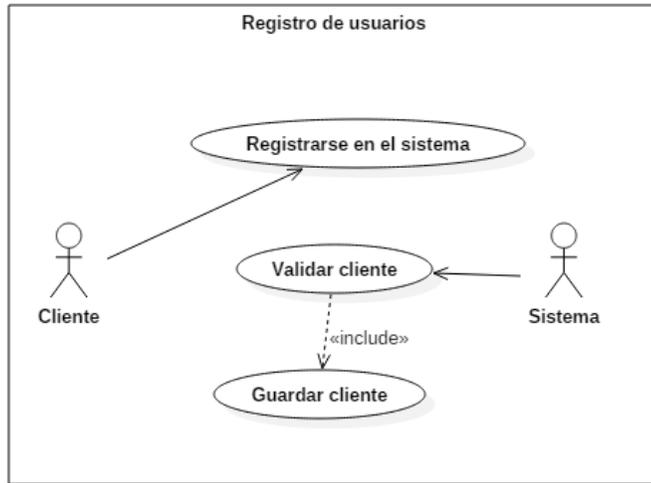


Ilustración 1. Casos de uso registro del cliente

## Caso de uso: Iniciar sesión.

### 2.1 Breve descripción.

El cliente debe ingresar al sistema con su nombre de usuario y contraseña para poder iniciar sesión.

### 2.2 Flujo de eventos.

#### 2.2.1 Flujo básico.

1. El caso de uso inicia cuando el cliente ingresa a la aplicación y ésta le pide su nombre de usuario y contraseña.

2. Después de que el cliente haya ingresado sus datos, el sistema validará que se encuentre registrado y que sus datos sean correctos, y sólo así lo dejará utilizar la aplicación.

#### 2.2.1 Flujo alternativo.

Ninguno

### 2.3 Requerimientos especiales.

Ninguno.

2.4 Condiciones previas.

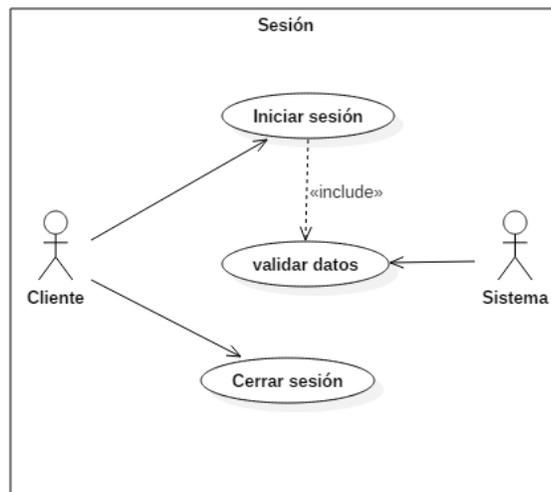
El cliente está registrado en la base de datos y puede utilizar la aplicación.

2.5 Postcondiciones.

El cliente ha puede ingresar o consultar información sin ningún problema.

2.6 Puntos de extensión.

Ninguna.



**Ilustración 2 Caso de uso Iniciar sesión**

Caso de uso: Gestionar terminales

3.1 Breve descripción.

El cliente puede insertar, editar o eliminar terminales en la aplicación.

3.2 Flujo de eventos.

3.2.1 Flujo básico.

1. El caso de uso inicia cuando el usuario inicia sesión con su nombre de usuario y contraseña.

2. El sistema le muestra la interfaz de inicio a la aplicación.

3. El cliente se dirige al botón terminales.

4. El sistema le muestra diferentes opciones entre ellas 'ingresar, editar o eliminar' terminales.

5. El cliente solicita al sistema alguna opción.

6. El sistema realiza la acción sin problemas.

### 3.2.2 Flujo alternativo

1.- La contraseña o usuario son incorrectos.

El sistema validó que la contraseña o el nombre de usuario no son correctos, el sistema envía un mensaje con el dato que es incorrecto, el usuario tendrá la opción de volver a introducir el dato que no incorrecto.

2.- Ingresar Terminal

El sistema envía una interfaz con los datos referente a las terminales, el usuario le da clic en el botón Nuevo, el sistema le mostrará los campos que debe llenar con información referente a las terminales, cuando termine de llenar éstos le tendrá que dar clic en insertar, el sistema le mostrará un mensaje confirmando que los datos han sido insertados correctamente.

3.- Editar Terminal

El sistema envía una interfaz con los datos referente a las terminales, el usuario le da clic al registro que desea editar, el sistema llenará las cajas de texto de acuerdo a la

información del registro seleccionado, el usuario podrá editar la información y dar clic en modificar.

#### 4.- Eliminar Terminal

El sistema envía una interfaz con los datos referente a las terminales, el usuario le da clic al registro que desea eliminar, el sistema llenará las cajas de texto de acuerdo al registro seleccionado, el usuario le dará clic en eliminar, el sistema le mandará un alerta que si está seguro de eliminar la información y si el usuario desea eliminar la información le podrá dar clic en aceptar, el sistema le mandará un mensaje de terminal eliminada; de lo contrario le dará clic en cancelar.

#### 3.3 Requerimientos especiales.

Ninguno.

#### 3.4 Condiciones previas.

El usuario debe iniciar sesión para realizar las diferentes opciones del sistema.

#### 3.5 Postcondiciones.

El usuario puede utilizar libremente las diferentes secciones del sistema

#### 3.6 Puntos de extensión.

Ninguno

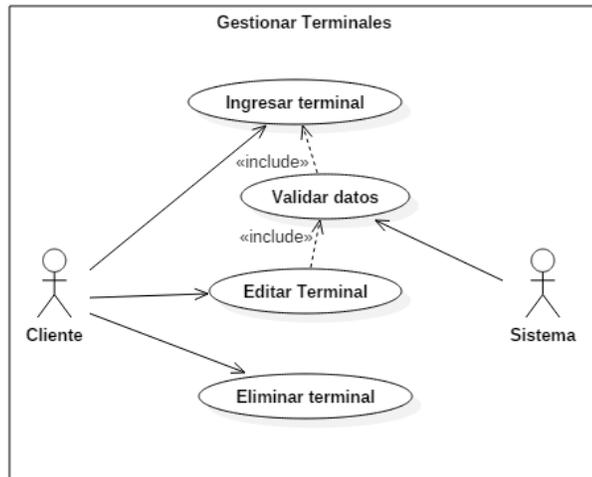


Ilustración 3. Caso de uso gestionar terminales

## Caso de uso: Gestionar unidades

### 4.1 Breve descripción.

El cliente ingresa al sistema para insertar, editar o eliminar unidades en la aplicación.

### 4.2 Flujo de eventos.

#### 4.2.1 Flujo básico.

1. El caso de uso inicia cuando el usuario inicia sesión con su nombre de usuario y contraseña.
2. El sistema le muestra la interfaz de inicio a la aplicación.
3. El cliente se dirige al botón unidades.
4. El sistema le muestra diferentes opciones entre ellas ‘ingresar, editar o eliminar’ unidad.
5. El cliente solicita al sistema alguna opción.
6. El sistema realiza la acción sin problemas.

#### 4.2.2 Flujo alternativo

##### 1.- Ingresar Unidad

El sistema envía una interfaz con los datos referente a las unidades, el usuario le da clic en el botón Nuevo, el sistema le mostrará los campos que debe llenar con información referente a las unidades, cuando termine de llenar éstos le tendrá que dar clic en insertar, el sistema le mostrará un mensaje confirmando que los datos han sido insertados correctamente.

##### 2.- Editar Unidad

El sistema envía una interfaz con los datos referente a las unidades, el usuario le da clic al registro que desea editar, el sistema llenará las cajas de texto de acuerdo a la información del registro seleccionado, el usuario podrá editar la información y dar clic en modificar.

##### 3.- Eliminar Unidad

El sistema envía una interfaz con los datos referente a las unidades, el usuario le da clic al registro que desea eliminar, el sistema llenará las cajas de texto de acuerdo al registro seleccionado, el usuario le dará clic en eliminar, el sistema le mandará un alerta que si está seguro de eliminar la información y si el usuario desea eliminar la información le podrá dar clic en aceptar, el sistema le mandará un mensaje de unidad eliminada; de lo contrario le dará clic en cancelar.

#### 4.3 Requerimientos especiales.

Ninguno.

#### 4.4 Condiciones previas.

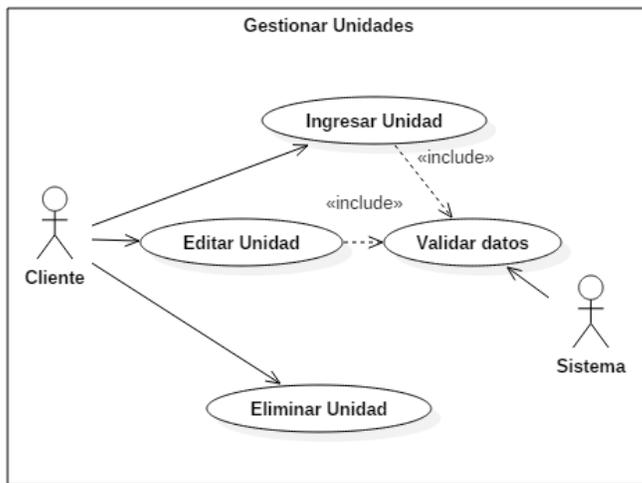
El usuario debe iniciar sesión para realizar las diferentes opciones del sistema.

4.5 Postcondiciones.

El usuario puede utilizar libremente los diferentes módulos del sistema

4.6 Puntos de extensión.

Ninguno



**Ilustración 4. Caso de uso gestionar unidades**

### **Caso de uso: Gestionar operadores**

5.1 Breve descripción.

El cliente ingresa al sistema para insertar, editar o eliminar operadores en la aplicación.

5.2 Flujo de eventos.

5.2.1 Flujo básico.

1. El caso de uso inicia cuando el usuario inicia sesión con su nombre de usuario y contraseña.

2. El sistema le muestra la interfaz de inicio a la aplicación.

3. El cliente se dirige al botón operadores.
4. El sistema le muestra diferentes opciones entre ellas ‘ingresar, editar o eliminar’ operadores.
5. El cliente solicita al sistema alguna opción.
6. El sistema realiza la acción sin problemas.

#### 5.2.2 Flujo alternativo

##### 1.- Ingresar Operador

El sistema envía una interfaz con los datos referente a los operadores, el usuario le da clic en el botón Nuevo, el sistema le mostrará los campos que debe llenar con información referente a los operadores, cuando termine de llenar éstos le tendrá que dar clic en insertar, el sistema le mostrará un mensaje confirmando que los datos han sido insertados correctamente.

##### 2.- Editar Operador

El sistema envía una interfaz con los datos referente a los operadores, el usuario le da clic al registro que desea editar, el sistema llenará las cajas de texto de acuerdo a la información del registro seleccionado, el usuario podrá editar la información y dar clic en modificar.

##### 3.- Eliminar Operador

El sistema envía una interfaz con los datos referente a los operadores, el usuario le da clic al registro que desea eliminar, el sistema llenará las cajas de texto de acuerdo al registro seleccionado, el usuario le dará clic en eliminar, el sistema le mandará un alerta que si está seguro de eliminar la información y si el usuario desea eliminar la información le

podrá dar clic en aceptar, el sistema le mandará un mensaje de operador eliminado; de lo contrario le dará clic en cancelar.

5.3 Requerimientos especiales.

Ninguno.

5.4 Condiciones previas.

El usuario debe iniciar sesión para realizar las diferentes opciones del sistema.

5.5 Postcondiciones.

El usuario puede utilizar libremente los diferentes módulos del sistema

5.6 Puntos de extensión.

Ninguno.

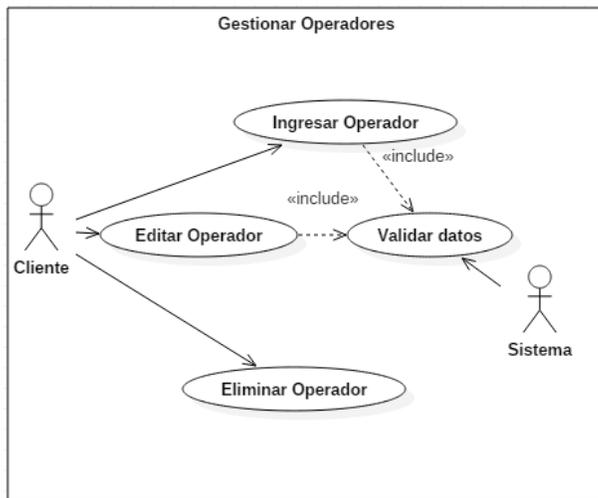


Ilustración 5. Caso de uso gestionar operadores

## Caso de uso: Tiempos

6.1 Breve descripción.

El cliente ingresa al sistema para insertar, editar o eliminar recorridos en la aplicación.

## 6.2 Flujo de eventos.

### 6.2.1 Flujo básico.

1. El caso de uso inicia cuando el usuario inicia sesión con su nombre de usuario y contraseña.

2. El sistema le muestra la interfaz de inicio a la aplicación.

3. El cliente se dirige al botón recorrido.

4. El sistema le muestra diferentes opciones entre ellas "ingresar, editar o eliminar" recorrido.

5. El cliente solicita al sistema alguna opción.

6. El sistema realiza la acción sin problemas.

### 6.2.2 Flujo alternativo

#### 1.- Ingresar Recorrido

El sistema envía una interfaz con los datos referente a los recorridos, el usuario le da clic en el botón Nuevo, el sistema le mostrará los campos que debe llenar con información referente a los recorridos, cuando termine de llenar éstos le tendrá que dar clic en insertar, el sistema le mostrará un mensaje confirmando que los datos han sido insertados correctamente.

#### 2.- Editar Recorrido

El sistema envía una interfaz con los datos referente a los recorridos, el usuario le da clic al registro que desea editar, el sistema llenará las cajas de texto de acuerdo a la

información del registro seleccionado, el usuario podrá editar la información y dar clic en modificar.

### 3.- Eliminar Recorrido

El sistema envía una interfaz con los datos referente a los recorridos, el usuario le da clic al registro que desea eliminar, el sistema llenará las cajas de texto de acuerdo al registro seleccionado, el usuario le dará clic en eliminar, el sistema le mandará un alerta que si está seguro de eliminar la información y si el usuario desea eliminar la información le podrá dar clic en aceptar, el sistema le mandará un mensaje de recorrido eliminado; de lo contrario le dará clic en cancelar.

### 6.3 Requerimientos especiales.

Ninguno.

### 6.4 Condiciones previas.

El usuario debe iniciar sesión para realizar las diferentes opciones del sistema.

### 6.5 Postcondiciones.

El usuario puede utilizar libremente los diferentes módulos del sistema

### 6.6 Puntos de extensión.

Ninguno.

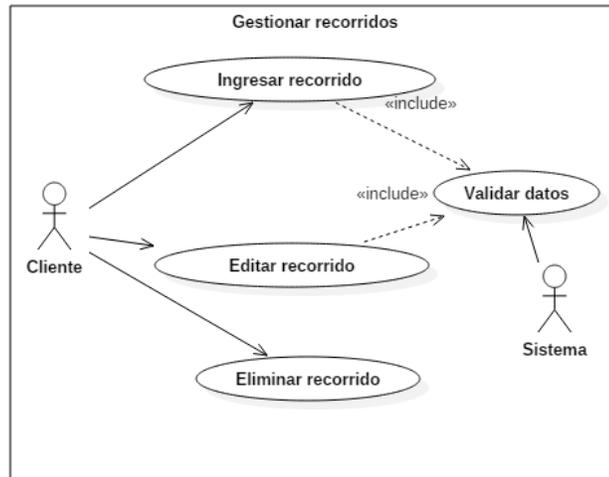


Ilustración 6. Caso de uso gestionar recorridos

### Caso de uso: Generar reporte.

#### 7.1 Breve descripción.

El cliente ingresa al sistema para generar o guardar reportes de los datos insertados anteriormente a la aplicación.

#### 7.2 Flujo de eventos.

##### 7.2.1 Flujo básico.

1. El caso de uso inicia cuando el usuario inicia sesión con su nombre de usuario y contraseña.
2. El sistema le muestra la interfaz de inicio a la aplicación.
3. El cliente se dirige al botón generar reporte.
4. El sistema le muestra diferentes opciones de los reportes que puede generar.
5. El cliente solicita al sistema alguna opción.
6. El sistema realiza la acción sin problemas.

## 7.2.2 Flujo alternativo

### 1.- Generar reporte

El sistema envía una interfaz con los diferentes reportes que puede generar, el usuario le da clic en alguna opción, el sistema pedirá los periodos de tiempo de los cuales desea adquirir la información, cuando termine de llenar éstos le tendrá que dar clic en generar, el sistema le mostrará el reporte de dicha opción.

### 2.- Guardar reporte

Después de haber generado el reporte el usuario podrá guardar el reporte dando clic en la imagen guardar que aparecerá dentro de las opciones del reporte y el reporte se guardará en donde lo especifique el usuario.

### 3.- Imprimir reporte

Después de haber generado el reporte el usuario podrá imprimir el reporte dando clic en la imagen imprimir que aparecerá dentro de las opciones del reporte y el reporte se imprimirá en unos segundos (dependiendo de la funcionalidad de la impresora).

## 7.3 Requerimientos especiales.

Ninguno.

## 7.4 Condiciones previas.

El usuario debe iniciar sesión para realizar las diferentes opciones del módulo generar reportes.

## 7.5 Postcondiciones.

El usuario puede utilizar libremente los diferentes módulos del sistema

## 7.6 Puntos de extensión.

Ninguno.

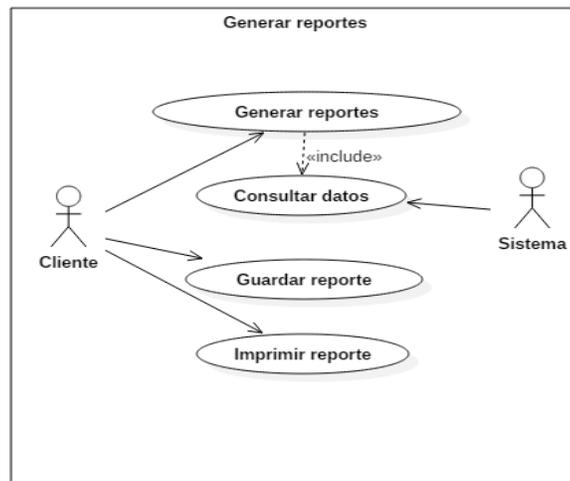


Ilustración 7. Caso de uso generar reportes

## 3.2 Elaboración

El principal objetivo de la fase de elaboración es probar la arquitectura del sistema a desarrollar. El punto es asegurar que se puede desarrollar un sistema que satisfaga los requisitos, y la mejor manera de hacerlo que es la construcción de extremo a extremo del esqueleto de trabajo del sistema conocido como "prototipo de la arquitectura" que su significado es software funcional de alto nivel, el cual incluye varias casos de uso de alto riesgos (a partir de un punto de vista técnico) para demostrar que el sistema es técnicamente factible.

En esta fase se realizaron los diagramas de:

- Sistemas
- Clases
- Modelo relacional de datos

Diagrama del sistema.

Este diagrama muestra las clases e interfaces que contiene el sistema.

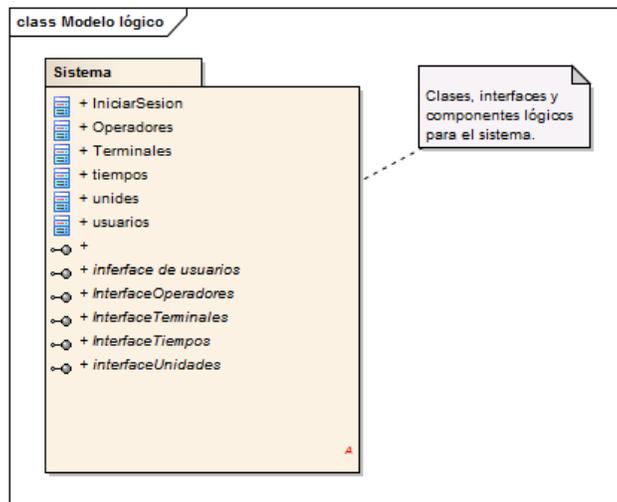


Ilustración 8. Diagrama del sistema

Diagrama de clases

Este diagrama muestra las principales clases que se realizaron para poder dar funcionalidad al sistema.

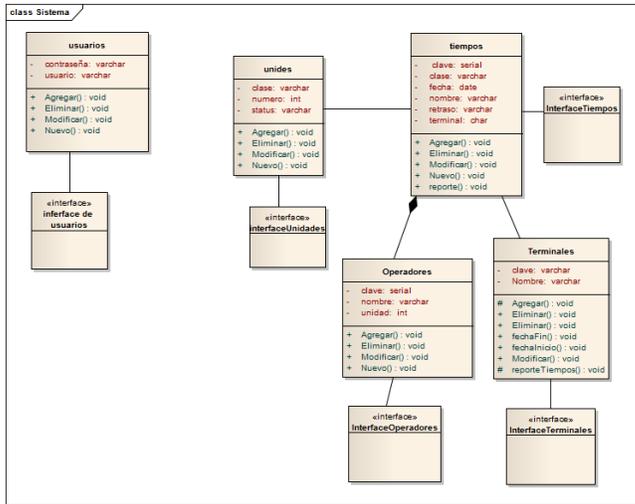


Ilustración 9. Diagrama de clases

## Modelo relacional

Este diagrama muestra las tablas y nombre de los atributos que contiene las tablas en la base de datos.

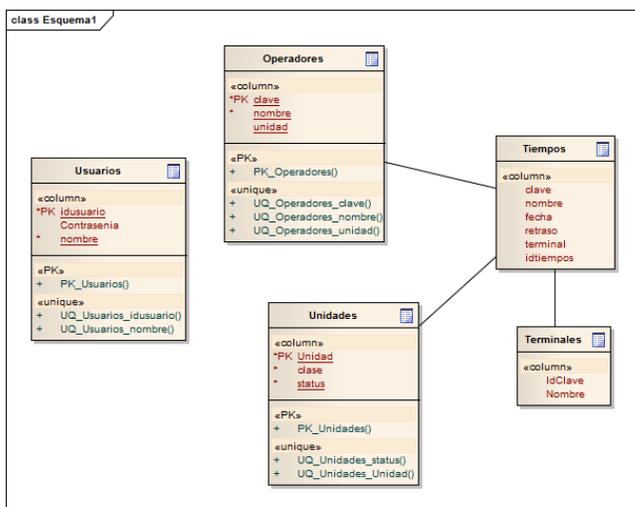


Ilustración 10. Modelo relacional

### 3.1 interfaces

#### Inicio de sesión



Ilustración 11. Interfaz inicio de sesión

#### Interface de menú principal

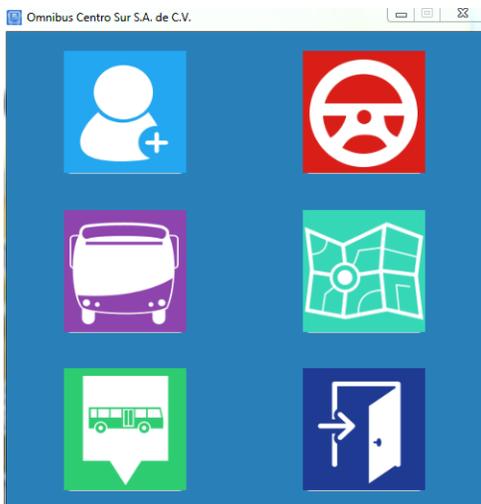
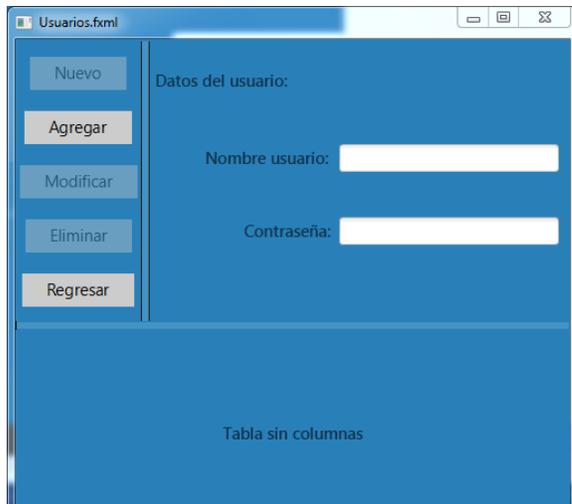


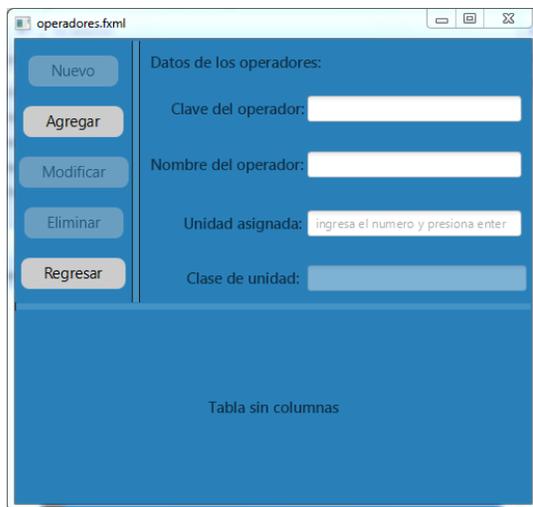
Ilustración 12. Interfaz de menú principal

#### Interface de Administración de usuarios



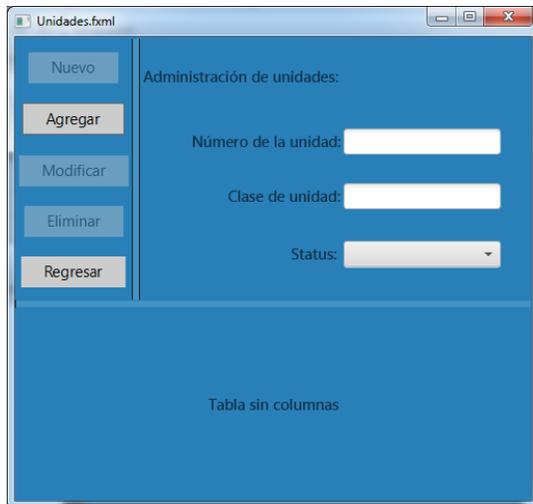
**Ilustración 13. Interfaz de Administración de usuarios**

Interface de Administración de operadores



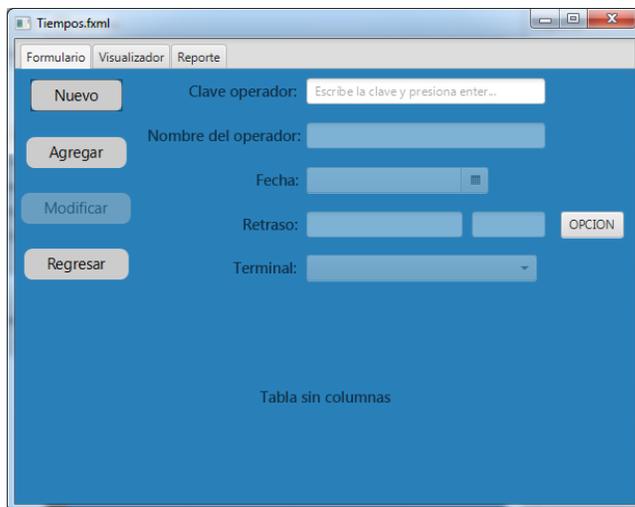
**Ilustración 14. Interfaz de Administración de operadores**

Interface de Administración de unidades.



**Ilustración 15. Interfaz de Administración de unidades.**

Interface de Control de tiempos.



**Ilustración 16. Interfaz de Control de tiempos**

Interface de Administración de terminales.

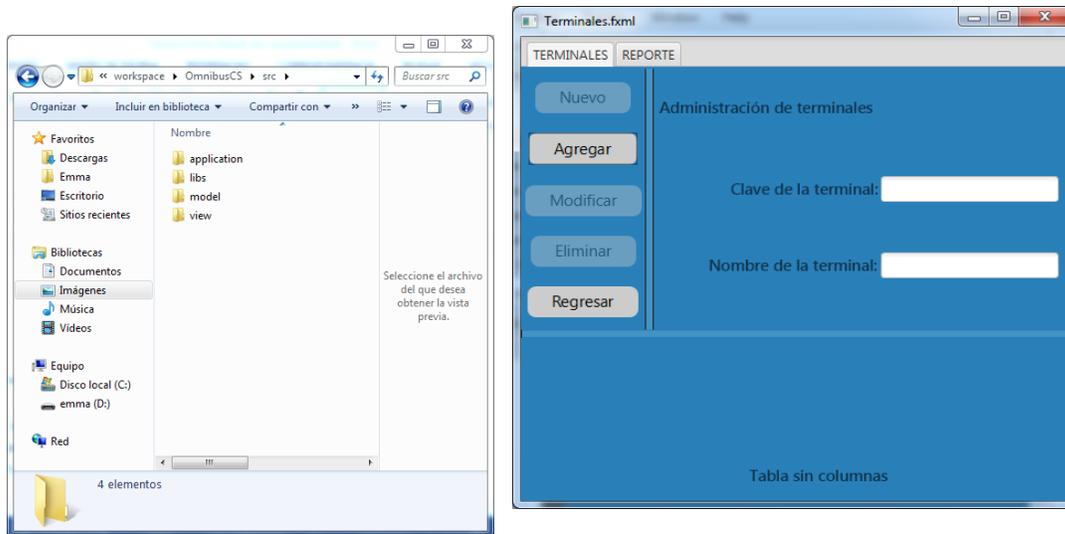


Ilustración 17. Interfaz de Administración de terminales.

### 3.3 Construcción.

En esta fase se empezó a programar la funcionalidad al sistema.

Antes de iniciar con la programación se crearon las carpetas pertenecientes, implementado el patrón de arquitectura de software MVC (Modelo-Vista-Controlador) para posteriormente hacer el código fuente que le daría la funcionalidad al sistema.

Ilustración 18. Encarpetado MVC

Después del encarpetao se le dio funcionalidad a los módulos

- iniciar sesión.

De acuerdo al encarpetao se fueron haciendo las clases.

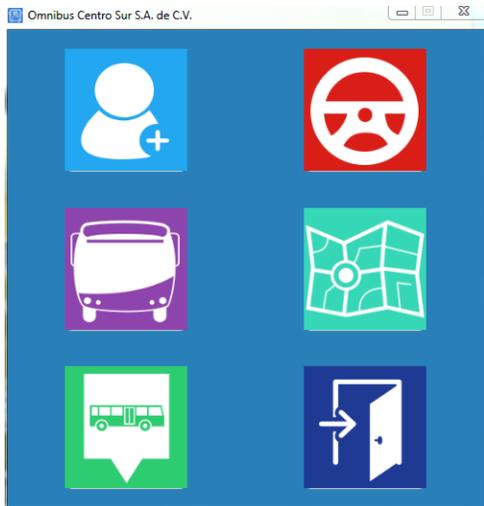
En el model se crearon las clase csConexion y csLogin, en View se ingresaron las interfaces correspondientes, y en controller se creó la clase ctrlLogin, todas con sus métodos pertenecientes, una vez finalizado se le hicieron sus respectivas pruebas y se realizaron algunas modificaciones, dando así por terminado el módulo iniciar sesión.



**Ilustración 19. Funcionalidad del inicio de sesión**

- Módulo menú

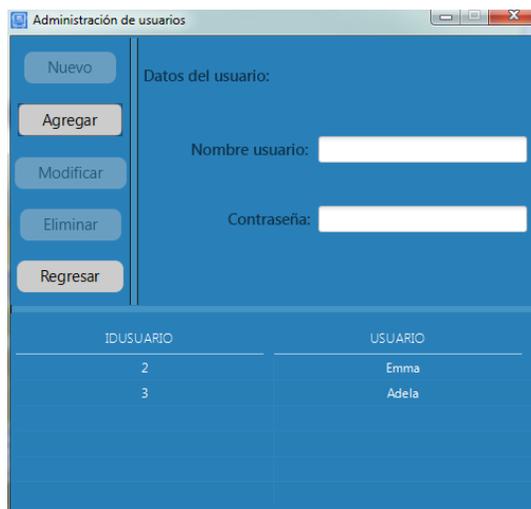
Para realizar este módulo se creó la clase ctrlMenu, realizando la codificación necesaria para darle funcionalidad al menú. Una vez terminado se le realizaron las pruebas pertenecientes y se dio por finalizado el módulo.



**Ilustración 20. Funcionalidad del menú**

- Módulo Administración de usuarios

Para realizar este módulo se crearon las clases csUsuarios y ctrlUsuarios, en cada una de ellas se codificaron sus métodos pertenecientes. Una vez terminado se le realizaron las pruebas pertenecientes y se dio por finalizado el módulo.



**Ilustración 21. Funcionalidad del módulo administración de usuarios**

- Módulo Administración de operadores

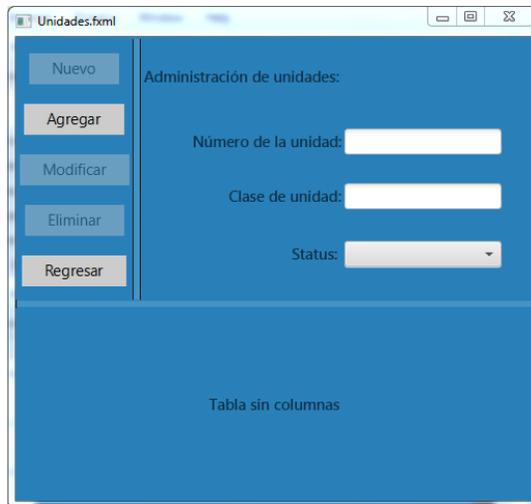
Para dar funcionalidad a este módulo se crearon las clases `csOperadores` y `ctrlOperadores` en cada una de ellas se codificaron sus métodos pertenecientes. Una vez terminado se le realizaron las pruebas pertenecientes y se dio por finalizado el módulo.



**Ilustración 22. Funcionalidad del módulo administración de operadores**

- Módulo Administración de unidades

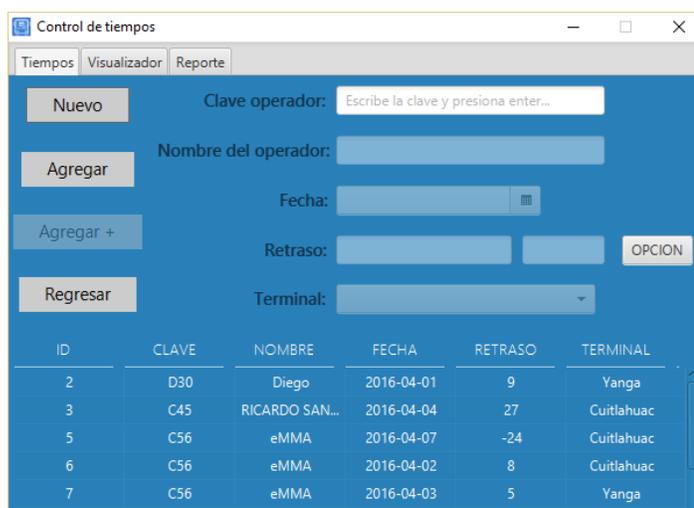
Para dar funcionalidad a este módulo se crearon las clases `csUnidad` y `ctrlUnidad` en cada una de ellas se codificaron sus métodos pertenecientes. Una vez terminado se le realizaron las pruebas pertenecientes como que no dejara escribir letras donde va el número de unidad y viceversa, y finalmente se dio por finalizado el módulo.



**Ilustración 23. Funcionalidad del módulo administración de unidades**

- Módulo Control de tiempos

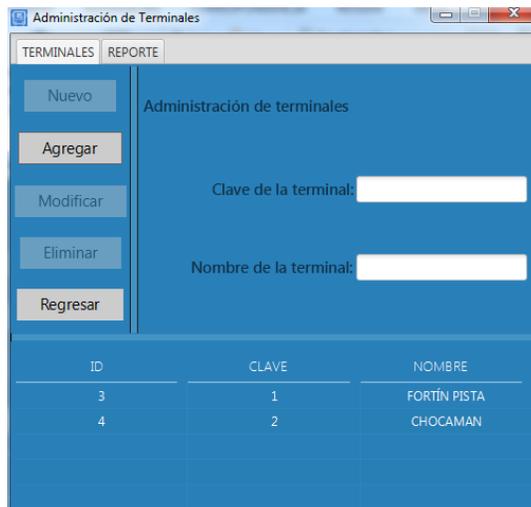
Para dar funcionalidad a este módulo se crearon las clases `csTiempos` y `ctrlTiempos` en cada una de ellas se codificaron sus métodos pertenecientes. Una vez terminado se le realizaron las pruebas pertenecientes, y finalmente se dio por finalizado el módulo.



**Ilustración 24. Funcionalidad del módulo control de tiempos**

- Módulo Administración de terminales

Para dar funcionalidad a este módulo se crearon las clases `csTerminal` y `ctrlTerminal`, en cada una de ellas se codificaron sus métodos pertenecientes. Una vez terminado se le realizaron las pruebas pertenecientes, y finalmente se dio por finalizado el módulo.



**Ilustración 25. Funcionalidad del módulo administración de terminales**

Cabe mencionar que en la carpeta controller también se hizo la clase `TipoDatos`, esta con el fin de validar que se introduzcan los tipos de datos que le corresponde a cada cuadro de texto.

Finalmente se diseñaron los reportes y también se hizo la clase `ctrlReportes` la cual hace referencia a los reportes y

---

## CAPÍTULO 4. PROCEDIMIENTO

---

Para poder realizar el sistema primero se llevó a cabo el levantamiento de requisitos que en la primera etapa de la metodología se describen los procesos que se llevaron a cabo para poder determinar la funcionalidad del sistema, tomando en cuenta ello, se empezaron a realizar los diagramas, para ello fue necesario el uso de otra aplicación llamada Enterprise Archite con el fin de diseñar los diagramas.

Las tecnologías utilizadas para el desarrollo de la aplicación fueron:

Tabla 1. Tecnologías utilizadas

Tecnología	Uso
Postgres	Realizar la base de datos
JavaFX	Diseño de las interfaces
Java	Entorno de desarrollo
JasperSoft Report	Diseño de los reportes

Con el programa JavaFX se diseñaron las interfaces y en seguida se empezó a codificar las clases en el entorno de Java para dar la funcionalidad a la aplicación; cabe mencionar que cada módulo que se fue desarrollando se le realizó pruebas para validar que se insertaran los datos correspondientes; también el usuario final le realizó diferentes pruebas.

Una vez terminados todos los módulos se empezó a diseñar los reportes, terminando éstos se dio por finalizada la aplicación y se realizó el manual de usuario en donde se le explica detalladamente al usuario el uso de la aplicación.

Finalmente se integró toda la información para tener un solo documento de todo lo que se debió hacer para desarrollar la aplicación.

---

## **CAPÍTULO 5 EVALUACIÓN DE RESULTADOS**

---

Habiendo realizado todos los módulos y reportes como se fue requerido por el usuario final, se puede decir que el objetivo se ha logrado ya que el uso de esta aplicación aporta los beneficios que esperaba la empresa Omnibus Centro Sur, S.A de C.V.

---

---

## **CAPÍTULO 6. CONCLUSIONES**

El desarrollo del proyecto Control de Tiempos en Trayectos es un proyecto desarrollado a la medida, debido a que todas sus funcionalidades fueron desarrolladas en base a las necesidades del área de operaciones de la empresa Omnibus. Es por ello que con apoyo de los usuarios finales se ha logrado desarrollar una aplicación muy eficiente.

---

## CAPÍTULO 7 REFERENCIAS

---

Molina, J. (2015, 11). HTML5. HTML5 - HTML|MDN. Obtenido 11, 2015, de <https://developer.mozilla.org/es/docs/HTML/HTML5>

Mtx, R. (2015, 08). CSS3. CSS3 | MD5. Obtenido 11, 2015, de <https://developer.mozilla.org/es/docs/Web/CSS/CSS3>

Iván, P. (2015, 08). JavaScript. JavaScript | MDN. Obtenido 11, 2015, de <https://developer.mozilla.org/es/docs/Web/JavaScript>

Delgado Reyes, M. (2000, 11). Diseño y construcción de un sistema de control de tiempos de recorrido en los buses de transporte público. Repositorio Digital EPN. Obtenido 11, 2015, de <http://bibdigital.epn.edu.ec/handle/15000/5126>

Alpizar Naranjo, D. (2006, 05). The Agile Unified Process. The Agile Unified Process. Obtenido 11, 2015, de <http://www.cc.una.ac.cr/AUP/index.html>

Foundation, T. (2015, 01). Eclipse. *Eclipse Downloads*. Obtenido 11, 2015, de <https://eclipse.org/downloads/>

Oracle, C. (2015, 01). JavaFX Scene Builder. JavaFX Scene Builder 2.0 Download. Obtenido 11, 2015, de <http://www.oracle.com/technetwork/java/javase/downloads/sb2download-2177776.html>

TIBCO Software, Inc., (2015, 01). Jaspersoft® Studio. Jaspersoft® Studio | Jasper Community. Obtenido 11, 2015, de <http://community.jaspersoft.com/project/jaspersoft-studio>

Oracle, (2015, 01). ¿Qué es la tecnología Java y para qué la necesito? ¿Qué es la tecnología Java y para qué la necesito? Obtenido 11, 2015, de [https://www.java.com/es/download/faq/whatis\\_java.xml](https://www.java.com/es/download/faq/whatis_java.xml)

Oracle, (2015, 01). JavaFX Scene Builder. JavaFX Scene Builder Information. Obtenido 11, 2015, de <http://www.oracle.com/technetwork/java/javase/downloads/javafxscenebuilder-info-2157684.html>

Postgresql, (2010, 10). Sobre Postgresql Sobre Postgresql Obtenido 11, 2015, de [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql)

---

## CAPÍTULO 8. ANEXOS

---

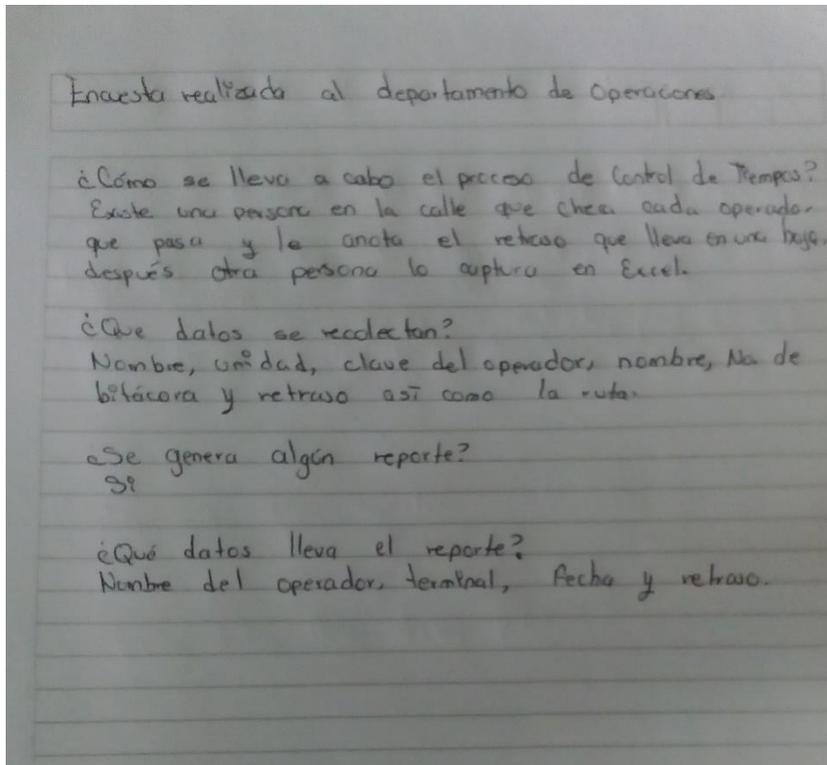


Ilustración 26. Entrevista