Sistema de  
Venta de Automóviles

Dedico este trabajo a mis padres.

INDICE

RESUMEN 5

INTRODUCCIÓN 6

ALCANCES Y LIMITES 7

REQUERIMIENTOS 8

Requerimientos Funcionales 8

Requerimientos no Funcionales 8

ANALISIS DE REQUERIMIENTOS 9

ACTORES 9

CASOS DE USO 10

DISEÑO DE REQUERIMIENTOS 11

CU001 - Logueo 11

Caso de Uso 11

Especificación del Caso de Uso 11

Prototipo 12

Diagrama E-R 12

Diagrama de Secuencia 13

El Servicio 13

CU002 - Alta de una ciudad 15

Caso de Uso 15

MODELO DE LA BASE DE DATOS 16

Diagrama E-R 16

PROCESO ALTA DE UN CLIENTE 17

Caso de Uso 17

Especificación del Caso de Uso 17

Prototipo 18

REGISTRAR VENTA 19

Caso de Uso 19

Especificación del Caso de Uso 19

Prototipo 20

CRUD DE CLIENTES 21

Análisis 21

Diseño 21

# RESUMEN

# INTRODUCCIÓN

# ALCANCES Y LIMITES

# REQUERIMIENTOS

## Requerimientos Funcionales

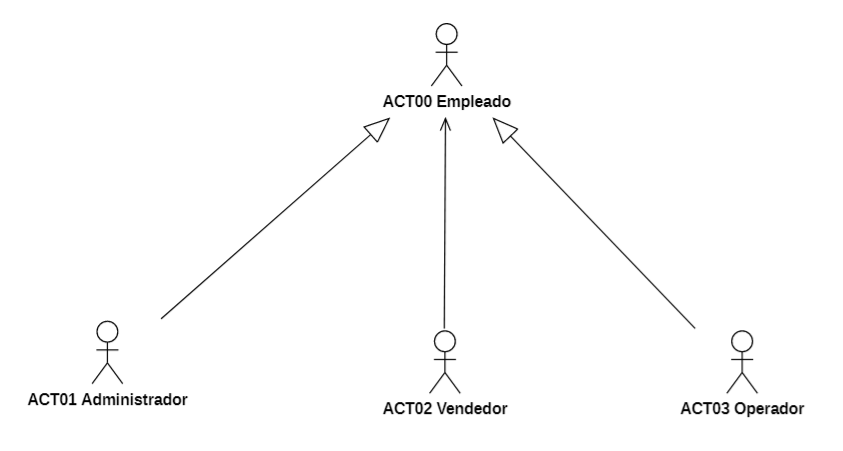
1. Acceso al sistema mediante usuario y clave
2. Gestión de ciudades destino
3. Gestión de rutas
4. Programación de viajes
5. Gestión de ventas de pasajes
6. Gestión de clientes
7. Gestión de trabajadores (Choferes, azafatas, etc.)

## Requerimientos no Funcionales

1. Los permisos de acceso al sistema podrán ser cambiados solamente por el administrador de acceso a datos.
2. Si se identifican ataques de seguridad o brecha del sistema, el mismo no continuará operando hasta ser desbloqueado por un administrador de seguridad.
3. Desarrollado en Lenguaje Java con base de datos SQL Server.
4. Facilidad de uso (Usabilidad).
5. Tiempo de respuesta adecuado.
6. Sistema concurrente con por lo menos 100 usuarios simultáneos.

# ANALISIS DE REQUERIMIENTOS

## ACTORES



Los actores son usuarios del sistema.

|  |  |  |
| --- | --- | --- |
| **CODIGO** | **NOMBRE** | **DESCRIPCION** |
| **ACT00** | Empleado | Representa un actor abstracto, en este caso, se refiere a cualquier empleado de la empresa. |
| **ACT01** | Administrador | Es el encargado de gestionar las rutas, buses, trabajadores, programar los viajes y asignar tribulación por viaje. |
| **ACT02** | Vendedor | Representa a todos los vendedores.  Es el responsable de gestionar las ventas, que implica: registrar, una venta, gestionar cambios de horarios de los boletos, etc.. |
| **ACT03** | Operador | Representa a todos los empleados que serán usuarios del sistema, y se encargarán de realizar tareas del negocio, como por ejemplo consultas y reportes. |

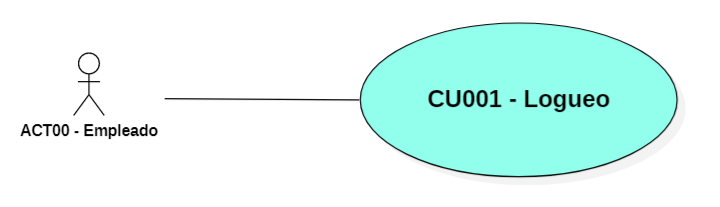
## CASOS DE USO

|  |  |  |
| --- | --- | --- |
| **CODIGO** | **NOMBRE** | **DESCRIPCION** |
| **CU001** | Logueo | Representa el proceso de inicio de sesión de los usuarios de la aplicación. |
| **CU002** | Alta de una ciudad | Este proceso permite registrar una nueva ciudad en la base de datos. |
| **CU003** | Creación de nueva ruta |  |
| **CU004** | Programación de viaje. |  |
| **CU005** | Asignar tripulación a viaje |  |
| **CU005** | Venta de pasajes |  |
|  |  |  |

# DISEÑO DE REQUERIMIENTOS

## CU001 - Logueo

### Caso de Uso



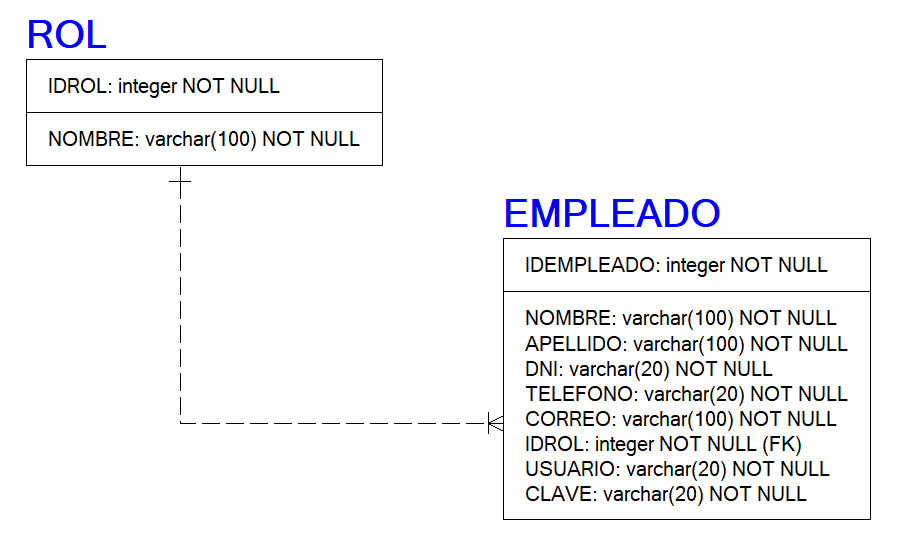
### Especificación del Caso de Uso

|  |  |
| --- | --- |
| **Identificador** | **CU0001** |
| **Descripción** | Representa el proceso de inicio de sesión de los usuarios de la aplicación. Es necesario que exista el usuario en la base de datos y se encuentre habilitado. |
| **Actor** | ACT00 |
| **Pre Condiciones** | El empleado debe tener su usuario y clave, y además debe estar activo. |
| **Flujo Básico** | Para poder iniciar este caso de uso, el empleado debe tener su usuario y clave a la mano.  El proceso a ejecutar es el siguiente:   1. El empleado ejecuta la aplicación. 2. La aplicación muestra la ventana de inicio de sesión o logueo. 3. El empleado ingresa su nombre de usuario. 4. El empleado ingresa su clave. 5. El usuario ejecuta el formulario. 6. Si los datos no son correctos, muestra el mensaje **"Datos incorrectos"** y regresa al punto 4. 7. Si los datos son correctos, la aplicación muestra el formulario principal. |
| **Flujos Alternativos** | No aplica |
| **Escenarios Claves** | Si el empleado no cuenta con usuario y clave, debe informar y solicitar sus credenciales al área respectiva. |
| **Post Condiciones** | Después de iniciar sesión correctamente, el empleado puede ejecutar las tareas que le corresponden según su perfil. |
| **Puntos de extensión** | No aplica. |
| **Requerimientos especiales** | La aplicación debe estar funcionando correctamente. |
| **Información adicional** | La aplicación solo puede ejecutarse desde la red interna de la empresa o utilizando una VPN. |
| **Requerimientos Relacionados** | No aplica. |

### Prototipo



### Diagrama E-R



### Diagrama de Secuencia



### El Servicio

#### Diseño



En la clase **LogonService** se implementa el servicio validar, este servicio recibe como datos de entrada el **usuario** y la **clave**, si existen en la base de datos retorna un objeto de tipo **EmpleadoDto** con los datos del usuario.

En caso de error o que no exista datos, genera una excepción de tipo **RuntimeException**.

#### Implementación

public EmpleadoDto validarUsuario(String usuario, String clave) {

EmpleadoDto empleadoDto = null;

String query = "select IDEMPLEADO, NOMBRE, APELLIDO, DNI, "

+ "TELEFONO, CORREO, IDROL, USUARIO, '\*\*\*\*\*\*' CLAVE "

+ "from empleado "

+ "where usuario=? and clave=?";

PreparedStatement pstm = null;

ResultSet rs = null;

Connection cn = null;

try {

cn = AccesoDB.getConnection();

pstm = cn.prepareStatement(query);

pstm.setString(1, usuario);

pstm.setString(2, clave);

rs = pstm.executeQuery();

if (rs.next()) {

empleadoDto = new EmpleadoDto();

empleadoDto.setIdempleado(rs.getInt("IDEMPLEADO"));

empleadoDto.setNombre(rs.getString("NOMBRE"));

empleadoDto.setApellido(rs.getString("APELLIDO"));

empleadoDto.setDni(rs.getString("DNI"));

empleadoDto.setTelefono(rs.getString("TELEFONO"));

empleadoDto.setCorreo(rs.getString("CORREO"));

empleadoDto.setIdrol(rs.getInt("IDROL"));

empleadoDto.setUsuario(rs.getString("USUARIO"));

empleadoDto.setClave(rs.getString("CLAVE"));

}

rs.close();

pstm.close();

if (empleadoDto == null) {

throw new SQLException("Datos incorrectos.");

}

} catch (SQLException e) {

throw new RuntimeException(e.getMessage());

} catch (Exception e) {

throw new RuntimeException("Error en el proceso, intenteo de nuevo.");

} finally {

try {

cn.close();

} catch (Exception e) {

}

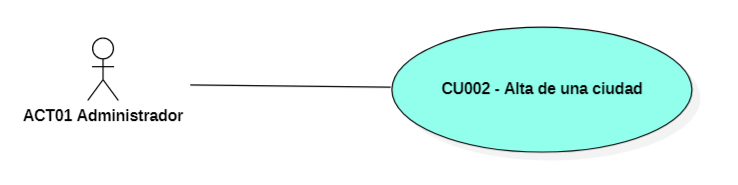
}

return empleadoDto;

}

## CU002 - Alta de una ciudad

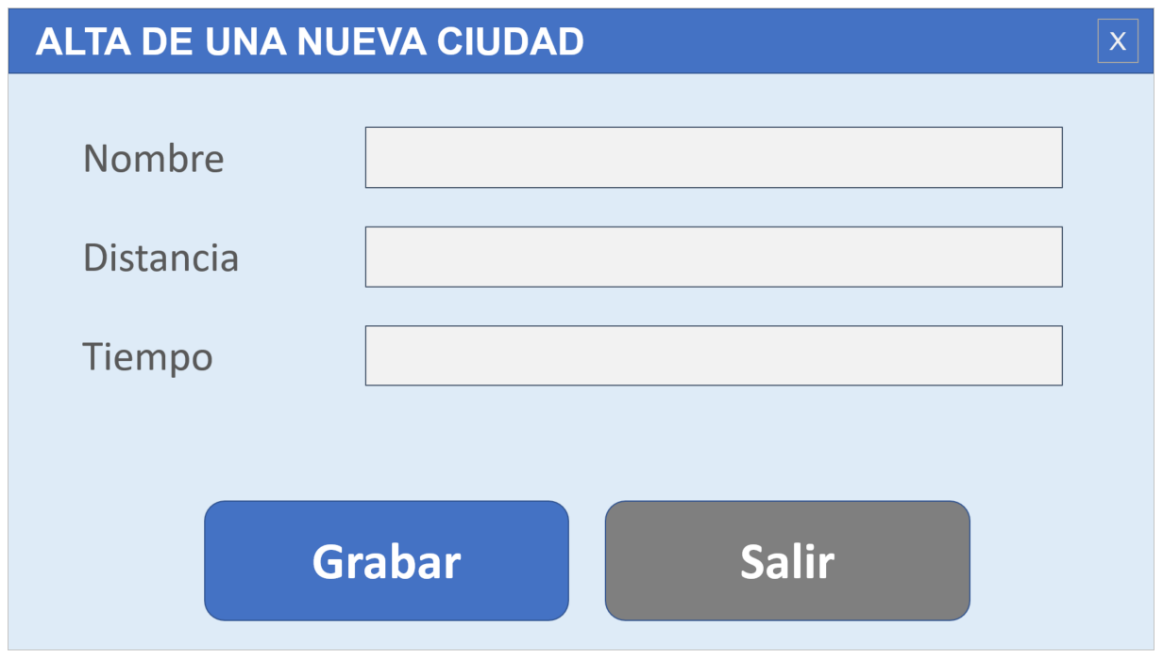
### Caso de Uso



### Especificación del Caso de Uso

|  |  |
| --- | --- |
| **Identificador** | **CU0002** |
| **Descripción** | Este proceso permite registrar una nueva ciudad en la base de datos. |
| **Actor** | ACT01 |
| **Pre Condiciones** | Es necesario que el usuario ya se encuentre contactado con el sistema. |
| **Flujo Básico** | El proceso a ejecutar es el siguiente:   1. El empleado ejecuta la aplicación. 2. El usuario ejecuta la opción para registrar nueva ciudad. 3. El usuario completa los campos del formulario don los datos de la nueva ciudad. 4. El usuario ejecuta el botón grabar para que se registre la nueva ciudad en la base de datos. 5. Si en el paso anterior se genera un error, pasa al paso 3 con datos correctos. 6. El sistema responde con el código de la nueva ciudad. |
| **Flujos Alternativos** |  |
| **Escenarios Claves** | Si el empleado no cuenta con usuario y clave, debe informar y solicitar sus credenciales al área de …….. |
| **Post Condiciones** | Después de registrar la nueva ciudad correctamente, ya se pueden crear nuevas rutas de transporte. |
| **Puntos de extensión** | No aplica. |
| **Requerimientos especiales** | La aplicación debe estar funcionando correctamente. |
| **Información adicional** | La aplicación solo puede ejecutarse desde la red interna de la empresa o utilizando una VPN. |
| **Requerimientos Relacionados** | No aplica. |

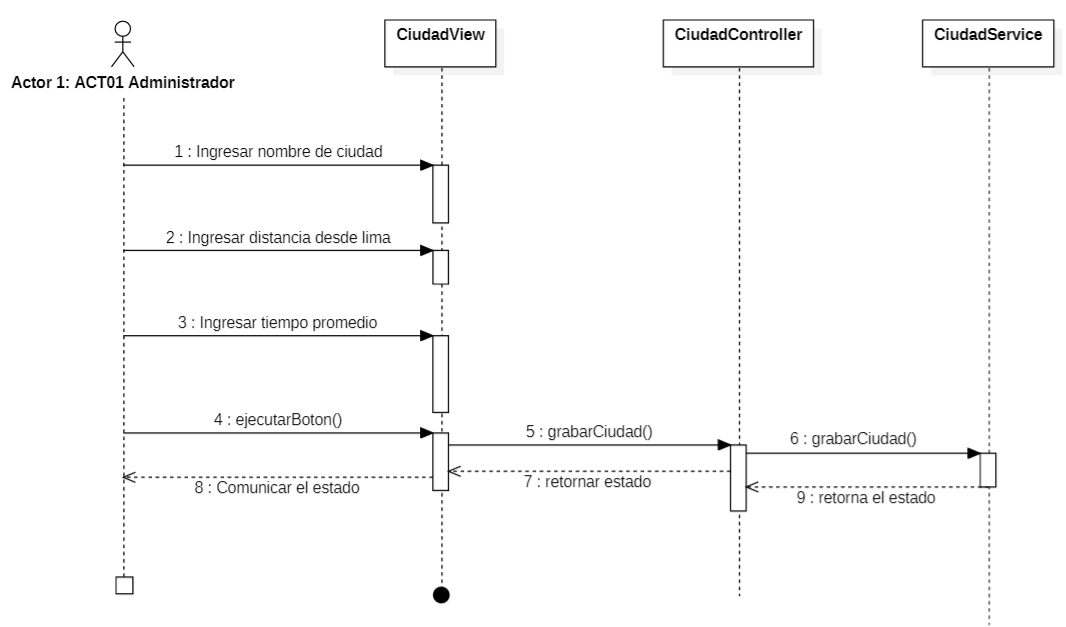
### Prototipo



### Diagrama E-R



### Diagrama de Secuencia



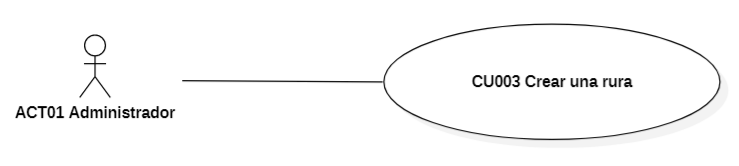
### Servicio

#### Diseño

#### Implementación

## CU003 – Crear un Ruta

### Caso de Uso



### Especificación del Caso de Uso

|  |  |
| --- | --- |
| **Identificador** | **CU0003** |
| **Nombre** | Crear una ruta |
| **Descripción** | Este proceso permite crear una nueva ruta. |
| **Actor** | ACT01 |
| **Pre Condiciones** | Es necesario que el usuario ya haya iniciado sesión en el sistema. |
| **Flujo Básico** | El proceso a ejecutar es el siguiente:   1. El usuario ejecuta el formulario para la creación de nuevas rutas. 2. El usuario establece la ciudad origen. 3. El usuario establece la ciudad destino. 4. El usuario establece los precios o tarifas respectivas según el tipo y piso del asiento. 5. El usuario ejecuta el botón grabar para que se registre la nueva ruta en la base de datos. 6. El sistema responde con el código de la nueva ciudad. |
| **Flujos Alternativos** |  |
| **Escenarios Claves** | Si el usuario no cuenta con el rol para crear nuevas rutas debe consultar con el gerente del área respectiva. |
| **Post Condiciones** | Después de registrar la nueva ruta, ya se puede programar salida utilizando esta nueva ruta. |
| **Puntos de extensión** | No aplica. |
| **Requerimientos especiales** | La aplicación debe estar funcionando correctamente. |
| **Información adicional** | La aplicación solo puede ejecutarse desde la red interna de la empresa o utilizando una VPN. |
| **Requerimientos Relacionados** | No aplica. |

### Prototipo



### Diagrama E-R



### Diagrama de Secuencia

### Servicio

#### Diseño

#### Implementación

## CU004 - Programación de viaje.

### Caso de Uso

### Especificación del Caso de Uso

### Prototipo

### Diagrama E-R

### Diagrama de secuencia

### El Servicio

#### Diseño

#### Implementación

## CU005 - Asignar Tripulación

### Caso de Uso

### Especificación del Caso de Uso

### Prototipo

### Diagrama E-R

### Diagrama de secuencia

### El Servicio

#### Diseño

#### Implementación

## CU006 – Venta de Pasajes

### Caso de Uso

### Especificación del Caso de Uso

### Prototipo

### Diagrama E-R

### Diagrama de secuencia

### El Servicio

#### Diseño

#### Implementación

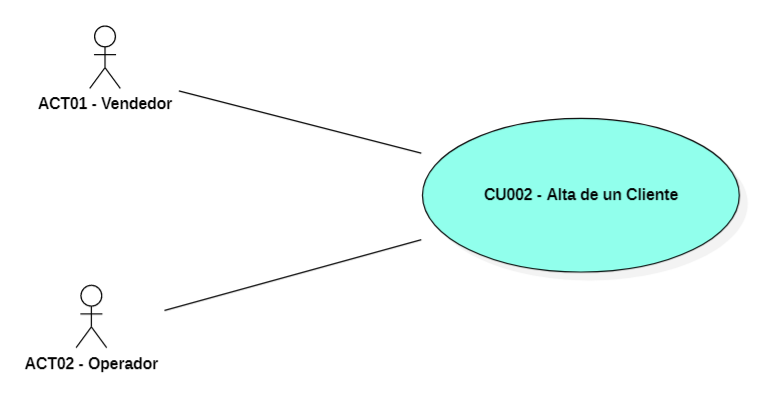
# MODELO DE LA BASE DE DATOS

## Diagrama E-R



# PROCESO ALTA DE UN CLIENTE

## Caso de Uso



## Especificación del Caso de Uso

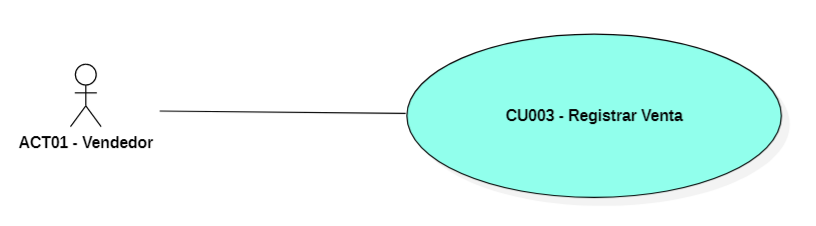
|  |  |
| --- | --- |
| **Identificador** | **CU0002** |
| **Descripción** | Este proceso permite dar de alta a un cliente en el sistema, esto es necesario para registrar la venta.  Este proceso lo puede hacer un vendedor o un operador. |
| **Actor** | ACT01, ACT02 |
| **Pre Condiciones** | El usuario debe haber iniciado sesión en el sistema. |
| **Flujo Básico** | Para poder iniciar este caso de uso, el usuario debe haber iniciado sesión en el sistema.  El proceso a ejecutar es el siguiente:   1. El usuario abre la ventana o formulario "ALTA DE UN CLIENTE". 2. Ingresa los datos del nuevo cliente:    * Nombres y apellidos    * Documento de identidad    * Dirección    * Celular    * Correo 3. Luego el usuario ejecuta el formulario. 4. Si los datos no son correctos, muestra el mensaje **"Datos incorrectos"** y regresa al punto 2. 5. Si los datos son correctos, la aplicación muestra el mensaje "**Proceso ejecutado correctamente**". 6. El usuario acepta el mensaje, luego el formulario se limpia para el ingreso de un nuevo cliente. |
| **Flujos Alternativos** | No aplica |
| **Escenarios Claves** | Si el empleado no cuenta con usuario y clave, debe informar y solicitar sus credenciales al área respectiva. |
| **Post Condiciones** | Después de registrar un cliente, el usuario puede optar por cerrar el sistema o ingresar un nuevo cliente. |
| **Puntos de extensión** | No aplica. |
| **Requerimientos especiales** | La aplicación debe estar funcionando correctamente. |
| **Información adicional** | La aplicación solo puede ejecutarse desde la red interna de la empresa. |
| **Requerimientos Relacionados** | No aplica. |

## Prototipo



# REGISTRAR VENTA

## Caso de Uso

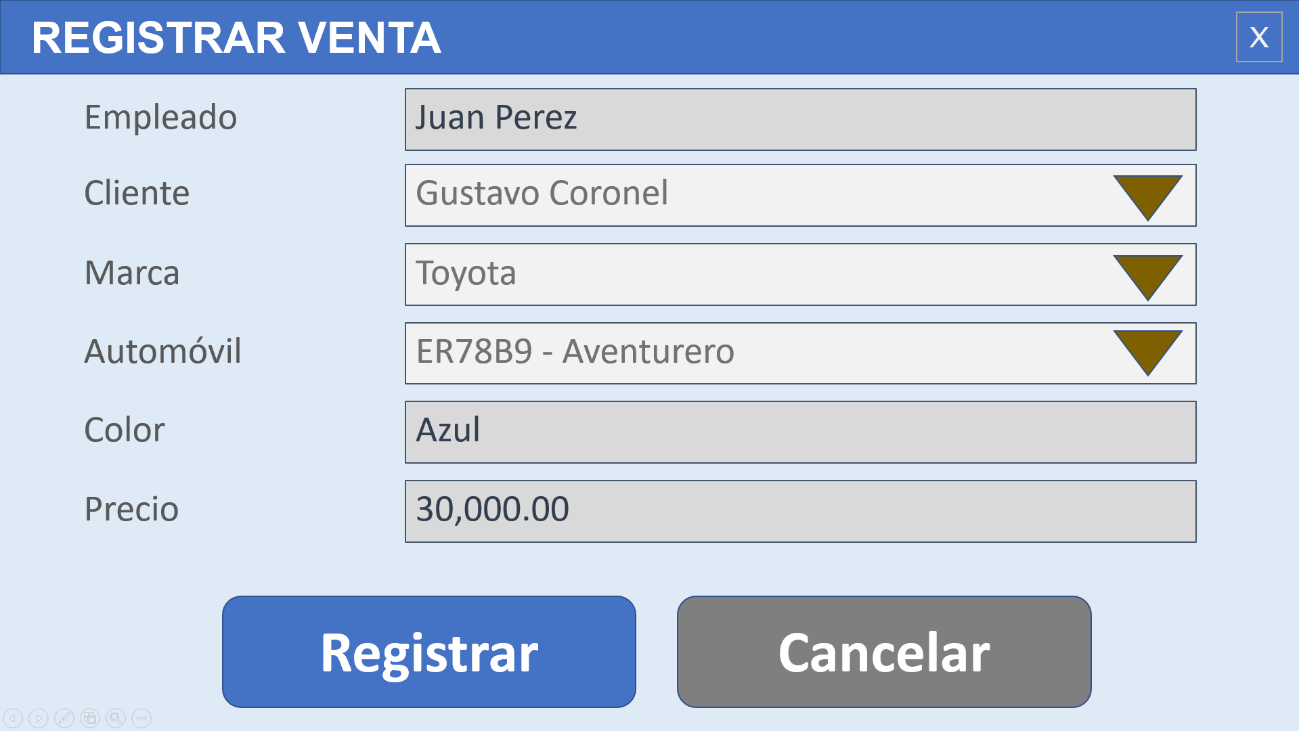


## Especificación del Caso de Uso

Falta concluir.

|  |  |
| --- | --- |
| **Identificador** | **CU0003** |
| **Descripción** | Este proceso permite registrar una venta.  En cada venta, un cliente solo puede comprar un automóvil.  Se el cliente quiere comprar 3 automóviles, serian 3 ventas. |
| **Actor** | ACT01 |
| **Pre Condiciones** | El usuario debe haber iniciado sesión en el sistema.  Es necesario que el cliente se encuentre registrado en el sistema. |
| **Flujo Básico** | Para poder iniciar este caso de uso, el usuario debe haber iniciado sesión en el sistema.  El proceso a ejecutar es el siguiente:   1. El usuario abre la ventana o formulario "REGISTRAR VENTA". 2. Selecciona el cliente que está comprando el automóvil. 3. Selecciona la marca del automóvil. 4. Selecciona el automóvil, se mostrará automáticamente su color y precio de venta. 5. Luego el usuario ejecuta el formulario. 6. Si los datos no son correctos, muestra el mensaje **"Datos incorrectos"** y regresa al punto 2. 7. Si los datos son correctos, la aplicación muestra el mensaje "**Proceso ejecutado correctamente**". 8. El usuario acepta el mensaje, luego el formulario se limpia para el registro de una nueva venta. |
| **Flujos Alternativos** | No aplica |
| **Escenarios Claves** | Si el empleado no cuenta con usuario y clave, debe informar y solicitar sus credenciales al área respectiva. |
| **Post Condiciones** | Después de registrar la venta, el cliente debe acercarse a caja para pagar el importe de la venta, y continuar con el proceso de entrega del automóvil. |
| **Puntos de extensión** | No aplica. |
| **Requerimientos especiales** | La aplicación debe estar funcionando correctamente. |
| **Información adicional** | La aplicación solo puede ejecutarse desde la red interna de la empresa. |
| **Requerimientos Relacionados** | No aplica. |

## Prototipo



# CRUD DE CLIENTES

## Análisis



## Diseño

