Universidad Nacional de La Matanza

**Año 2024**

**Proyecto:**

**Sistema Automotor - MotorLink**

**Número de Equipo: 02**

**Integrantes Equipo de Proyecto**

| DNI | Nombre | E-Mail |
| --- | --- | --- |
| 42656604 | Jorge Luis Lopez | jorgellopez@alumno.unlam.edu.ar |
| 39066715 | Federico Nicolas Loiero | floiero@alumno.unlam.edu.ar |
| 37804899 | Federico Maximiliano Rossendy | frossendy@alumno.unlam.edu.ar |
|  |  |  |
|  |  |  |

**ÍNDICE**

[1.Visión Preliminar 4](#_Toc171031487)

[1.1 Alcance 4](#_Toc171031488)

[1.2 Límites / Restricciones 5](#_Toc171031489)

[2. Especificación de Requerimientos 5](#_Toc171031490)

[2.1 Requerimientos Funcionales 5](#_Toc171031491)

[2.1.1.Requerimiento Funcional: Agregar Automotor 5](#_Toc171031492)

[2.1.2. Requerimiento Funcional: Modificar Automotor 6](#_Toc171031493)

[2.1.3. Requerimiento Funcional: Eliminar Automotor 7](#_Toc171031494)

[2.1.4. Requerimiento Funcional: Realizar Pago 7](#_Toc171031495)

[2.1.5. Requerimiento Funcional: Visualizar Facturas Pagadas 7](#_Toc171031496)

[2.1.6. Requerimiento Funcional: Exportar Facturas Pagadas 8](#_Toc171031497)

[2.1.7. Requerimiento Funcional: Visualizar Comprobante 8](#_Toc171031498)

[2.1.8. Requerimiento Funcional: Generar Factura 8](#_Toc171031499)

[2.1.9. Requerimiento Funcional: Automotores Con Deuda 9](#_Toc171031500)

[2.1.10. Requerimiento Funcional: Recaudación Por Municipio 9](#_Toc171031501)

[2.1.11. Requerimiento Funcional: Visualizar Facturas Impagas 9](#_Toc171031502)

[2.1.12. Requerimiento Funcional: Consultar Automotor 10](#_Toc171031503)

[2.2 Requerimientos No Funcionales 11](#_Toc171031504)

[2.2.1 Requerimientos de Performance 11](#_Toc171031505)

[2.2.2 Requerimientos lógicos de la base de datos 11](#_Toc171031506)

[2.2.3 Restricciones de Diseño 13](#_Toc171031507)

[3. Modelo de Casos de Uso 15](#_Toc171031508)

[3.1 Modelo de Casos de Uso 15](#_Toc171031509)

[3.2 Actores 16](#_Toc171031510)

[3.3 Descripción Casos de uso 16](#_Toc171031511)

[4. Experiencia de Usuario 18](#_Toc171031512)

[4.1 Prototipo de navegación 18](#_Toc171031513)

[4.2 Wireframes Mockups 26](#_Toc171031514)

[5. Arquitectura de Software 30](#_Toc171031515)

[5.1 Diagrama de Arquitectura 31](#_Toc171031516)

[5.2 Frameworks / Componentes utilizados 31](#_Toc171031517)

[5.3 Infraestructura tecnológica 31](#_Toc171031518)

[5.4 Otros diagramas según corresponda 32](#_Toc171031519)

[5.5 Estándar de codificación 32](#_Toc171031520)

[6. Plan de pruebas 33](#_Toc171031521)

[6.1 Diseño y ejecución de casos de prueba 33](#_Toc171031522)

[6.2 Seguimiento de fallas 37](#_Toc171031523)

[Anexos I. Informe de avance 38](#_Toc171031524)

# 1.Visión Preliminar

**Historia de Revisión**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

## 1.1 Alcance

1. Registrar el dominio de los automóviles en distintos municipios.
2. Modificar el titular y el municipio a través del dominio del cliente si este tiene tal vehículo registrado en un municipio.
3. Solo existirá un municipio entre otros.
4. Los vehículos contaran con un dominio.
5. Los pagos se abonarán en una cuota fija y no se podrá pagar en partes.
6. El dominio permite que los automóviles tengan un solo titular.
7. El pago se realizará una vez por mes.
8. El sistema almacenara el dominio, fecha, mes a pagar y monto ingresado.
9. Los autos solo tienen un único dominio.
10. El sistema permite comunicarse con una base de datos para obtener los municipios, las patentes, los pagos y titulares.
11. El sistema no brindará el pago efectuado por el titular, solo se filtrará por dominio
12. El sistema permitirá eliminar un dominio registrado y al titular ligado a esta.

## 1.2 Límites / Restricciones

A1- Un automóvil con un dominio podrá estar inscripto en un solo municipio.

A2- No pueden existir automóviles con dominios repetidos.

E1- El pago se realizará a través del dominio.

E2- No se admite un pago que supere el monto de la boleta (tiene que ser exactamente igual al monto que figura esta)

J1- No se permitirá comprar un dominio por otro medio.

# 2. Especificación de Requerimientos

**Historia de Revisión**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 4/07/2024 | 1.0.8 | Terminaciones sobre las funcionalidades y un resumen breve sobre las introducciones | Jorge Luis Lopez. |
|  |  |  |  |

## 2.1 Requerimientos Funcionales

### 2.1.1. Requerimiento Funcional: Agregar Automotor

|  |  |
| --- | --- |
| Introducción | Se agregará un automotor al sistema. |
| Entradas | * Datos del Titular (DNI, Nombre, Apellido) * Datos del Vehículo (Dominio, Marca, Modelo, Año Fab) * Municipio (Id) |
| Procesos | * Acciones para realizar:   + Se ingresa los datos del titular al sistema   + Validamos que el sistema pueda corroborar de forma satisfactoria los datos   + Se ingresan los datos para ingresar el automotor * Casos Alternativos:   + Si el cliente envía un dato erróneo (DNI Largo) el sistema mandara un mensaje de error diciéndole que el dato ingresado es invalido |
| Salidas | * El sistema obtendrá un nuevo automotor en la base de datos que también agregara el titular a la tabla de titulares, si es que ya existe en la tabla se le agregara el nuevo automotor a la lista. * El municipio ingresara este nuevo automotor a su tabla. |

### 2.1.2. Requerimiento Funcional: Modificar Automotor

|  |  |
| --- | --- |
| Introducción | Se modificará el automotor ingresado en el sistema |
| Entradas | * Datos del anterior titular (DNI) * Datos del nuevo titular (DNI, Nombre, Apellido) * Datos del Vehículo (Dominio) * Municipio (Id) |
| Procesos | * Acciones para realizar:   + Se ingresa los datos del anterior titular al sistema   + Ingresa los datos del vehículo y la patente a cambiar   + Validamos que el sistema tenga la patente registrada   + Se ingresan los datos del nuevo titular * Casos Alternativos:   + El sistema mandara un mensaje de error al usuario porque no existe, no encuentra el nombre o hay un DNI erróneo   + El vehículo o la patente puede no existir, no tener pagos registrados en los últimos meses, o que su estado sea "DADO DE BAJA"   + El nuevo titular puede tener otra patente a su nombre y por lo tanto se le denegara el nuevo registro, también puede tener un DNI erróneo. |
| Salidas | * + El nuevo registro del titular se intercambiará por el anterior en la base de datos, así teniendo un nuevo titular para la tabla patente. * El municipio validará que esta patente exista con un nuevo titular, por lo tanto, aparecerá también en su registro como que la patente sigue activa |

### 2.1.3. Requerimiento Funcional: Eliminar Automotor

|  |  |
| --- | --- |
| Introducción | Se eliminará un automotor del sistema. |
| Entradas | * Datos del Vehículo (Dominio) * Datos del usuario (DNI) |
| Procesos | * Acciones para realizar:   + El usuario ingresara el dominio del vehículo   + Se valida que exista en nuestro sistema   + El sistema le preguntara al usuario si quiere confirmar esta acción. * Casos Alternativos:   + Si la patente no existe en el sistema, se le informará al usuario que esa no existe. |
| Salidas | * La patente ingresada por el usuario se eliminará del sistema. * El titular ligado al dominio se eliminará del sistema siempre en cuando solo tenga un dominio, si tiene más de uno no se eliminará el titular. * El municipio eliminara el automotor de su registro. |

### 2.1.4. Requerimiento Funcional: Realizar Pago

|  |  |
| --- | --- |
| Introducción | Se realizará el pago de una factura elegida previamente |
| Entradas | * Lista de facturas impagas. |
| Procesos | * Acciones para realizar:   + El cliente selecciona la factura a pagar   + El sistema muestra un mensaje si quiere realizar esa acción   + El cliente elegirá realizar el pago * Casos alternativos:   + El cliente decide no realizar el pago |
| Salidas | * El sistema guardara el pago realizado. |

### 2.1.5. Requerimiento Funcional: Visualizar Facturas Pagadas

|  |  |
| --- | --- |
| Introducción | El sistema mostrara un detalle de la factura a visualizar. |
| Entradas | * Rango de Fechas * Datos del Vehículo (Dominio) * Lista de facturas |
| Procesos | * Acciones para realizar:   + El usuario ingresa el dominio de su automotor y la fecha con las que desea filtrar la lista de facturas. * Casos alternativos:   + En caso de fecha o el dominio sea invalida se le pedirá que corrija la información proporcionada. |
| Salidas | * Se mostrará por pantalla la lista de facturas que cumplan con los filtros. |

### 2.1.6. Requerimiento Funcional: Exportar Facturas Pagadas

|  |  |
| --- | --- |
| Introducción | Se creará un archivo en Excel con las facturas pagadas. |
| Entradas | * Lista de facturas pagadas |
| Procesos | * Acciones para realizar:   + El Usuario hará clic en el botón “Exportar” |
| Salidas | * Se mostrará por pantalla un mensaje indicando la ubicación del archivo generado. * Archivo xls con los datos pedidos. |

### 2.1.7. Requerimiento Funcional: Visualizar Comprobante

|  |  |
| --- | --- |
| Introducción | El sistema mostrar de una factura seleccionada el detalle |
| Entradas | * Lista de facturas pagadas |
| Procesos | * Acciones para realizar:   + El cliente selecciona la factura a visualizar |
| Salidas | * El sistema mostrara el detalle de la factura |

### 2.1.8. Requerimiento Funcional: Generar Factura

|  |  |
| --- | --- |
| Introducción | El sistema generara de forma automática las facturas que adeudan los automotores, también si se ingresa por primera vez al sistema |
| Entradas | * Tipo de automotor ingresado |
| Procesos | * Acciones para realizar:   + Se genera una factura para cada mes desde que se ingresa un nuevo automotor. |
| Salidas | * El sistema le mostrar los datos del titular y del automotor en la primera factura a pagar. * Se generar las facturas desde el mes en vigencia hasta fin de año. |

### 2.1.9. Requerimiento Funcional: Automotores Con Deuda

|  |  |
| --- | --- |
| Introducción | El sistema le filtrara respecto a un dominio en específico una tabla con las cuotas que debe el automotor. |
| Entradas | * Lista de automotores * Lista de titulares |
| Procesos | * Acciones para realizar:   + El administrador seleccionara el botón “listar automotores” |
| Salidas | * El sistema le mostrar los datos del titular y del automotor relacionado que por lo menos tengan una deuda. |

### 2.1.10. Requerimiento Funcional: Recaudación Por Municipio

|  |  |
| --- | --- |
| Introducción | El sistema muestra una tabla con la recaudación total por cada municipio. |
| Entradas | Lista de Municipios  Lista de Facturas |
| Procesos | * Acciones para realizar:   + Se filtrará la lista de facturas por municipio y estado   + Se sumarán los datos |
| Salidas | * El sistema mostrará una lista de recaudaciones por municipio. |

### 2.1.11. Requerimiento Funcional: Visualizar Facturas Impagas

|  |  |
| --- | --- |
| Introducción | El sistema muestra una tabla con las facturas impagas de un dominio en especifico |
| Entradas | * Rango de Fechas * Datos del Vehículo (Dominio) * Lista de facturas |
| Procesos | * Acciones para realizar:   + El usuario ingresa el dominio de su automotor y la fecha con las que desea filtrar la lista de facturas. * Casos alternativos:   + En caso de fecha o el dominio sea invalida se le pedirá que corrija la información proporcionada. |
| Salidas | * Se mostrará por pantalla la lista de facturas que cumplan con los filtros. |

### 2.1.12. Requerimiento Funcional: Consultar Automotor

|  |  |
| --- | --- |
| Introducción | El usuario podrá consultar los datos del automotor o también los datos de un titular. |
| Entradas | * Dominio * DNI |
| Procesos | * Acciones para realizar: (Funcional visualizar automotor)   + El usuario ingresa el dominio para luego consultar en el sistema cual es el automotor que está intentando buscar * Casos alternativos:   + Si el dominio es incorrecto o no pone dominio, este abortara la funcionalidad con un mensaje. * Acciones para realizar: (Funcional visualizar titular)   + El usuario ingresa el DNI para luego consultar en el sistema cual es el titular que está intentando buscar * Casos alternativos:   + Si el DNI es incorrecto o no pone ningún DNI, este abortara la funcionalidad con un mensaje. |
| Salidas | * (Funcional visualizar automotor):   + Se le brindara al usuario la información del dominio, DNI del titular, marca, modelo, año y municipio sobre el automotor buscado * (Funcional visualizar titular):   + Se le brindara al usuario la información del DNI, nombre y apellido sobre el titular buscado |

## 2.2 Requerimientos No Funcionales

### 2.2.1 Requerimientos de Performance

Los requerimientos de performance en nuestro sistema estarán arraigados a los datos que obtendremos de la data base, por ejemplo, si utilizamos el caso de uso visualizar facturas impagas tendremos un costo de memoria para almacenar todas las facturas que no están pagadas (además de utilizar un índice implementado en memoria) y solo podría ser usado por un usuario que consulte sobre las facturas impagas que tenga el dominio de su automotor.

### 2.2.2 Requerimientos lógicos de la base de datos

***Tipos de información usados por varias funciones:*** Los datos que nos proveen los clientes se almacenaran en distintos formatos para ser utilizados por otras funciones, por ejemplo, en la función de agregar automotor el cliente ingresara su auto para almacenar los datos en una tabla compuesta de distintos tipos de información: Automotor(dominio,dni\_titular,id\_municipio,marca,modelo,año\_fabr) No obstante el agregar un automotor requiere también hacerlo para la tabla de municipios: Municipios(id,localidad,descripcion).

Estos datos se utilizarán para diversas funciones del sistema, como el proceso de pago de una factura de un automotor, búsqueda y listados de automotores, además de consultas frecuentes sobre estas tablas

***Frecuencia de uso:*** Siguiendo con nuestro sistema, los datos de los Automotores y Boletas serian ingresados y consultados con demasiada frecuencia, ya que los nuevos clientes estarán agregando nuevos, modificando o eliminando sus autos en nuestro sistema. Por otro lado, los datos de Municipio serán accedidos con menos frecuencia, ya que solo se consultará el municipio para cargar los mismo en sistema a la hora de seleccionarlos. En este caso utilizamos el id y la localidad, ordenada por localidad.

***Capacidades de acceso:*** En el sistema de gestión de automotores nosotros podemos especificar que rol se define para cada actor, por ejemplo, el administrador podrá consultar, listar, agregar, modificar y eliminar automotores, mientras que el cliente/usuario solo podrá interactuar con los pagos o visualizar la facturas.

***Entidades de datos y sus relaciones:***

**Titular.**

**Atributos**: DNI (PK), Nombre, Apellido

**Automotor**

**Atributos**: Dominio (PK), DNI Titular (FK), Id Municipio (FK), Marca, Modelo, Año de Fabricación

**Municipio**

**Atributos**: ID (PK), Localidad, Descripción

**Patente**

**Atributos**: ID (PK), Dominio (FK), Estado

**Boleta**

**Atributos**: ID Boleta (PK), ID Patente (FK), Monto, Fecha de Vencimiento, Fecha de Pago, Estado

**Relaciones:**

**Titular - Automotor**

**Tipo**: Uno a muchos (1: N)

**Detalle**: Un titular puede tener múltiples automotores. Clave foránea DNI Titular en Automotor referencia DNI en Titular.

**Automotor - Municipio**

**Tipo**: Muchos a uno (N:1)

**Detalle**: Un automotor está registrado en un único municipio. Clave foránea Id Municipio en Automotor referencia ID en Municipio.

**Automotor - Patente**

**Tipo**: Uno a muchos (1: N)

**Detalle**: Un automotor puede tener varias patentes asociadas. Clave foránea Dominio en Patente referencia Dominio en Automotor.

**Patente - Boleta**

**Tipo**: Uno a muchos (1: N)

**Detalle**: Una patente puede tener múltiples boletas asociadas. Clave foránea ID Patente en Boleta referencia ID en Patente.

**Claves:**

**Primarias**:

DNI en Titular

Dominio en Automotor

ID en Municipio

ID en Patente

ID Boleta en Boleta

**Foráneas**:

DNI Titular en Automotor referencia DNI en Titular

Id Municipio en Automotor referencia ID en Municipio

Dominio en Patente referencia Dominio en Automotor

ID Patente en Boleta referencia ID en Patente

***Restricciones de Integridad:*** Una restricción de integridad en este caso podría ser que el campo de dominio no admite duplicados ni tampoco puede estar vacío, ya que esto define que no existe dos patentes iguales para un mismo auto. Otra restricción podría ser que el titular si o si tenga un automóvil, no pueden existir titulares sin automóviles en nuestro sistema. Y por último una patente debe tener boletas, ya sea pagas o impagas, no puede existir una patente que no tenga boletas ya que esto definiría que no está en nuestro sistema.

***Requerimientos de retención de datos:*** Nuestro sistema deberá omitir algunos datos si no se cumplen ciertas condiciones, el caso podría ser que el usuario intente pagar un automotor que no exista, esto podrían ser las condiciones de la restricción de integridad, ya que un automotor que no tenga pagos no será admitido en nuestro sistema. El pago de una boleta deberá ser total, no se admitirán pagos parciales.

### 2.2.3 Restricciones de Diseño

***Formato de informes:***

* Los informes de registro de vehículos deben incluir el nombre del titular, número de dominio, marca y modelo del vehículo
* Los informes de deudas deben listar la cuota de la boleta, ID de patente, monto, fecha de vencimiento, y estado del pago.

**Convenciones de Nombres**:

* Tablas: Titular, Automotor, Municipio, Patente, Boletas.
* Columnas: Titulares DNI, Automotores Dominio, Municipios ID.

**Procedimientos Contables**:

* Las boletas de pago deben seguir las normas contables locales y registrar correctamente los ingresos y estados de pago.

**Pistas de Auditoría**:

* Registrar todas las modificaciones a los datos de vehículos y titulares, incluyendo los datos actuales del nuevo titular y generarle una boleta de pago por su ingreso inicial.

***Limitaciones de hardware:***

Detallaremos los requerimientos de hardware necesarios para que el software opere de manera eficiente.

**Especificaciones Mínimas**:

* CPU: Intel Pentium
* RAM: 8 GB.
* Espacio en Disco: 500 GB HDD.

**Especificaciones Recomendadas**:

* CPU: Intel Core i7 o equivalente.
* RAM: 16 GB.
* Espacio en Disco: 1 TB SSD.

**Compatibilidad**:

* El software debe ser compatible con MySQL Server y distribuciones como Windows
* Debe soportar bases de datos MySQL.

**Consideraciones de Rendimiento**:

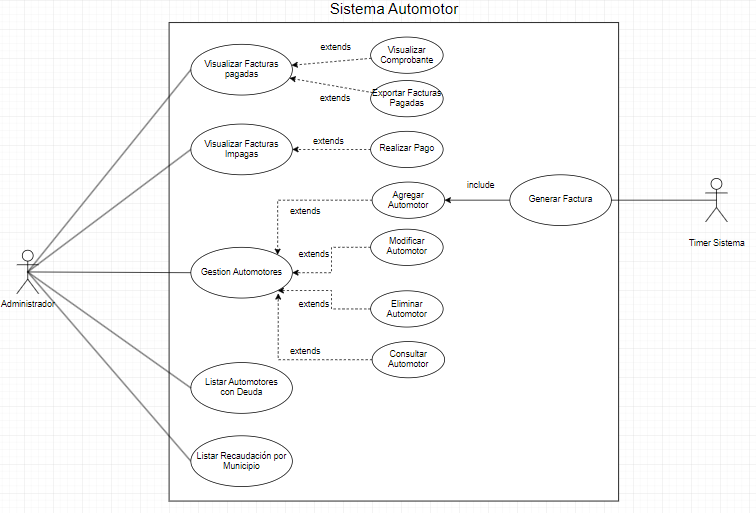
* Capacidad para manejar hasta 5,000 usuarios simultáneos y registrar hasta 100,000 vehículos.

# 3. Modelo de Casos de Uso

**Historia de Revisión**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 04/7/2024 | 1.0.7 | Cambios en los nombres de los actores | Federico Rossendy |
|  |  |  |  |

## 3.1 Modelo de Casos de Uso



## 3.2 Actores

Actor Administrador

El administrador podrá gestionar los automotores, que implica poder ingresar, eliminar y modificar los mismos, listar la recaudación por municipio y listar todos los automotores que no pagaron sus facturas.

Sistema Automotor

El sistema Automotor se encargará de crear las boletas sin pagar desde el mes en curso hasta fin de año.

## 3.3 Descripción Casos de uso

3.3.1 Visualizar Facturas pagadas //

El administrador pedirá visualizar el pago, con el dominio del vehículo y un filtro de fechas. El sistema verificará si el dominio existe y esta pagado, luego podrá extender al caso de uso Exportar Facturas Pagadas o Visualizar el Comprobante, o no hará nada. Si no existe el dominio, se mostrará un mensaje de error.

3.3.2 Visualizar Facturas impagas //

El administrador pedirá visualizar las facturas impagas, con el dominio del vehículo y un filtro de fechas. El sistema verificará si el dominio existe y si esta pagado, luego podrá extenderse al caso de uso Realizar Pago si es que lo desea. Si no existe el dominio, se mostrar un mensaje de error.

3.3.3 Eliminar automotor

El Administrador realiza el caso de uso eliminar motor incluyendo el dominio del automotor y el DNI del titular, el sistema verificará que este automotor exista, luego podrá eliminar el automotor. En caso de que no exista el automotor se mostrara un mensaje de error.

3.3.4 Agregar Automotor

El Administrador realiza el caso de uso ingresar automotor incluyendo los datos del cliente, el sistema verificará que la patente este a nombre del titular ingresado y que esta patente también exista, luego podrá ingresar el automotor. Si no existe la patente del automotor, se le mostrará un mensaje. Si los datos del motor son correctos se llamará al caso de uso incluido "Generar Factura".

3.3.5 Modificar automotor

El Administrador realiza el caso de uso modificar automotor incluyendo los datos del cliente y el nuevo usuario del automotor, el sistema verificará que este automotor y los datos del cliente sean correctos, luego podrá modificar el automotor deseado. Si la patente no existe, se le mostrará un mensaje.

3.3.6 Exportar Facturas Pagadas

El Cliente realiza el caso de uso Exportar Facturas Pagadas, el sistema tomara las patentes y los pagos ligados esta misma y los exportara a un archivo xls.

3.3.7 Listar Recaudación por Municipio

El Administrador realiza el caso de uso listar recaudación, el sistema mostrara por pantalla el nombre del municipio y la recaudación en pesos argentinos.

3.3.8 Listar Automotores con Deuda

El Administrador realiza el caso de uso listar automotores con deuda, el sistema mostrara por pantalla la cantidad de automotores con deuda, el porcentaje total de los automotores con deuda, la cantidad de pesos argentinos adeudados por todos los automotores en el sistema. Luego podrá consultar la desde de un automotor en particular ingresando su dominio.

3.3.9 Gestión Automotores

El Administrador realiza el caso de uso Gestionar automotores, el sistema le mostrará un menú donde podrá el usuario elegir si extenderse a los casos de uso "Agregar automotor", "Modificar automotor", "Eliminar automotor" y “Consultar Automotor”, también podrá elegir no extenderse a ninguno de los anteriores mencionados.

3.3.10 Generar Factura

El sistema realiza el caso de uso Generar factura, solo si se ejecuta el caso de uso "Ingresar automotor". Se genera facturas desde el mes que ocurre hasta el fin de año.

3.3.11 Visualizar comprobante

El Cliente seleccionara la factura que querrá visualizar y el sistema le mostrara todos los detalles de la factura seleccionada.

3.3.12 Realizar pago

El cliente inicia el caso de uso Realizar pago para alguna factura impaga. El sistema muestra un mensaje para confirmar el pago. El cliente confirma el pago.

3.3.13 Consultar.

El cliente inicia el caso de uso Consultar ingresando el dominio para visualizar los datos del automotor (dominio, DNI Titular, Marca, Modelo, Año y Municipio) o podrá ingresar el DNI para visualizar los datos de un Titular (DNI, nombre, apellido), si cualquiera de estos casos se intenta utilizar con un dominio/DNI vacío, el sistema abortará la funcionalidad con un mensaje de error.

# 4. Experiencia de Usuario

<versión>

<integrantes del grupo>

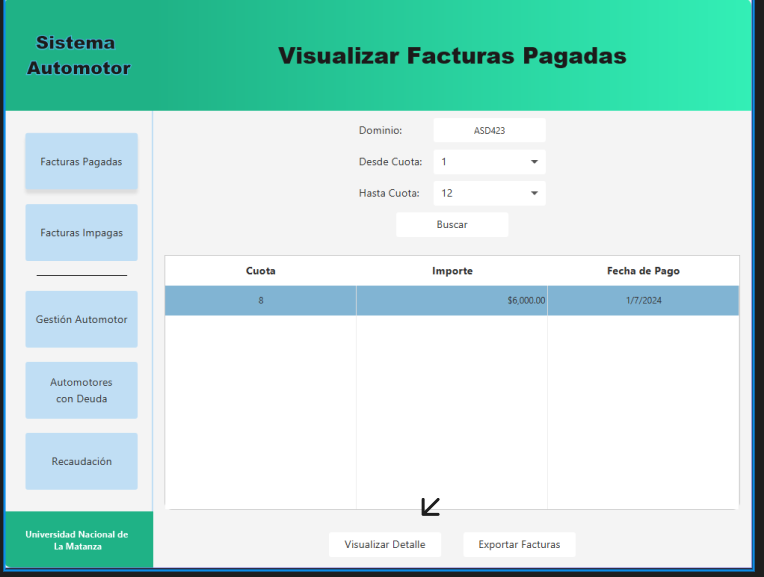
**Historia de Revisión**

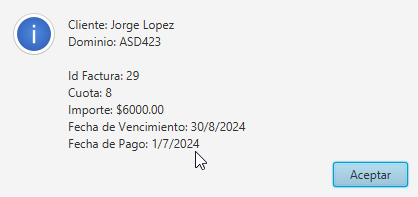
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 28/6/2024 | 1.0.6 | Cambios en el prototipo de navegación y mockups actualizados | Jorge Luis Lopez. |
|  |  |  |  |

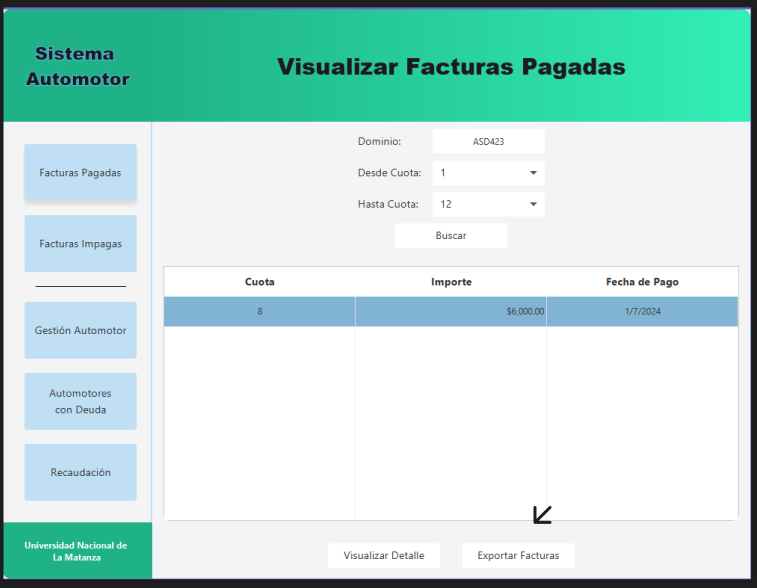
## 4.1 Prototipo de navegación

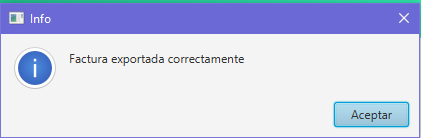
Propondremos los mockups de cada búsqueda por pantalla especifica.

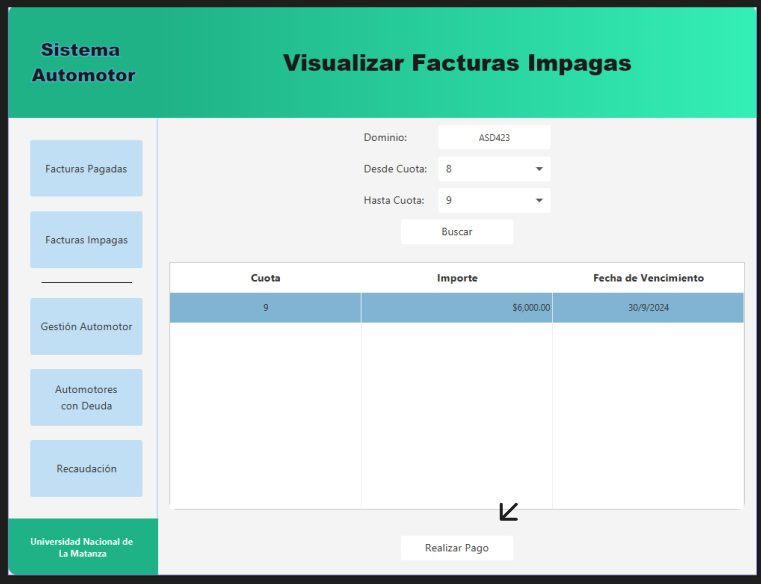
"Con el dominio ASD423"

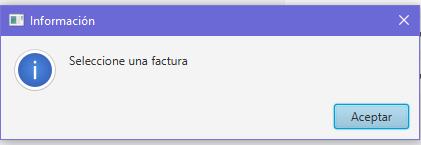


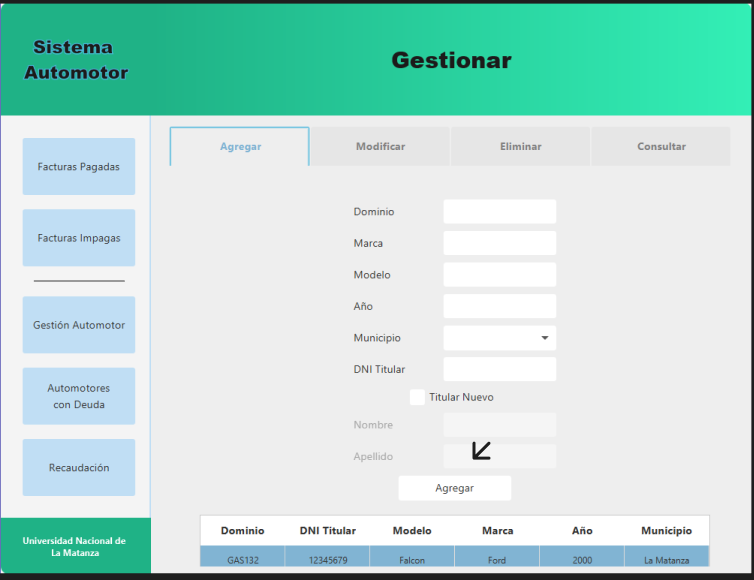


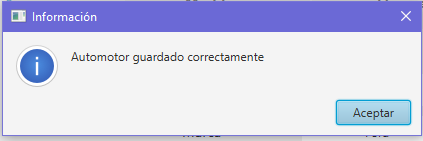


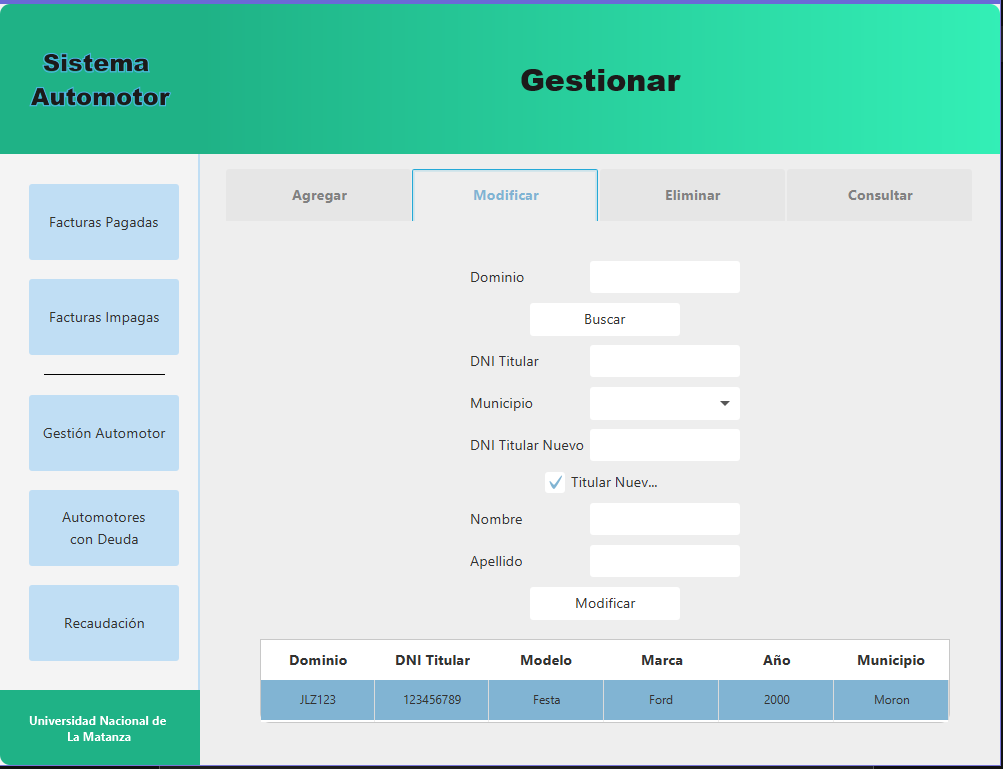


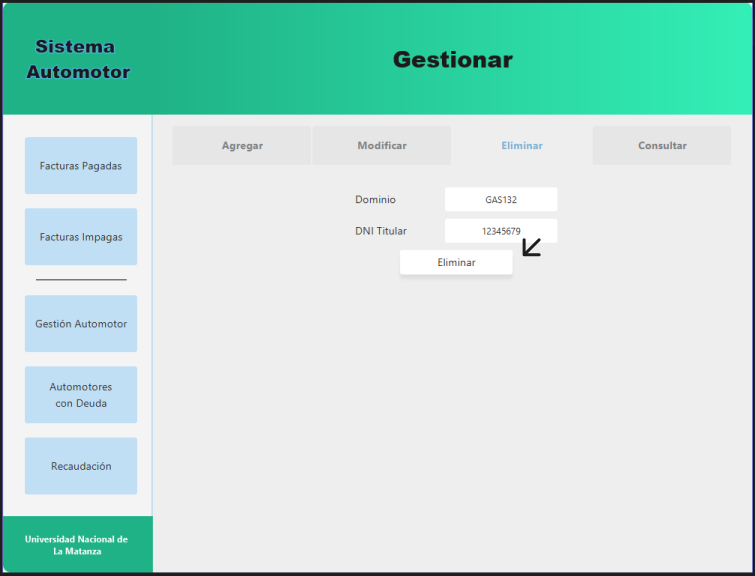


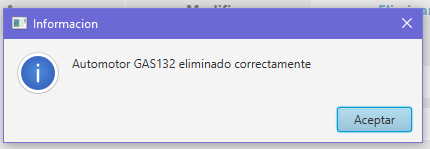


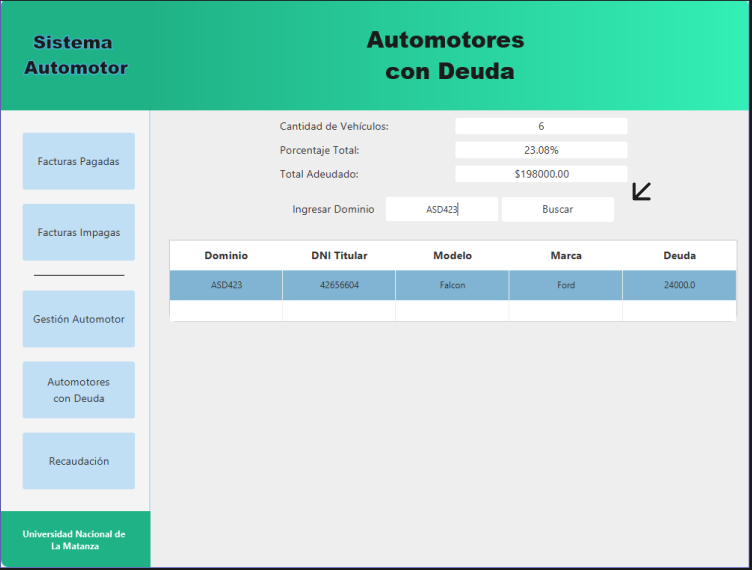








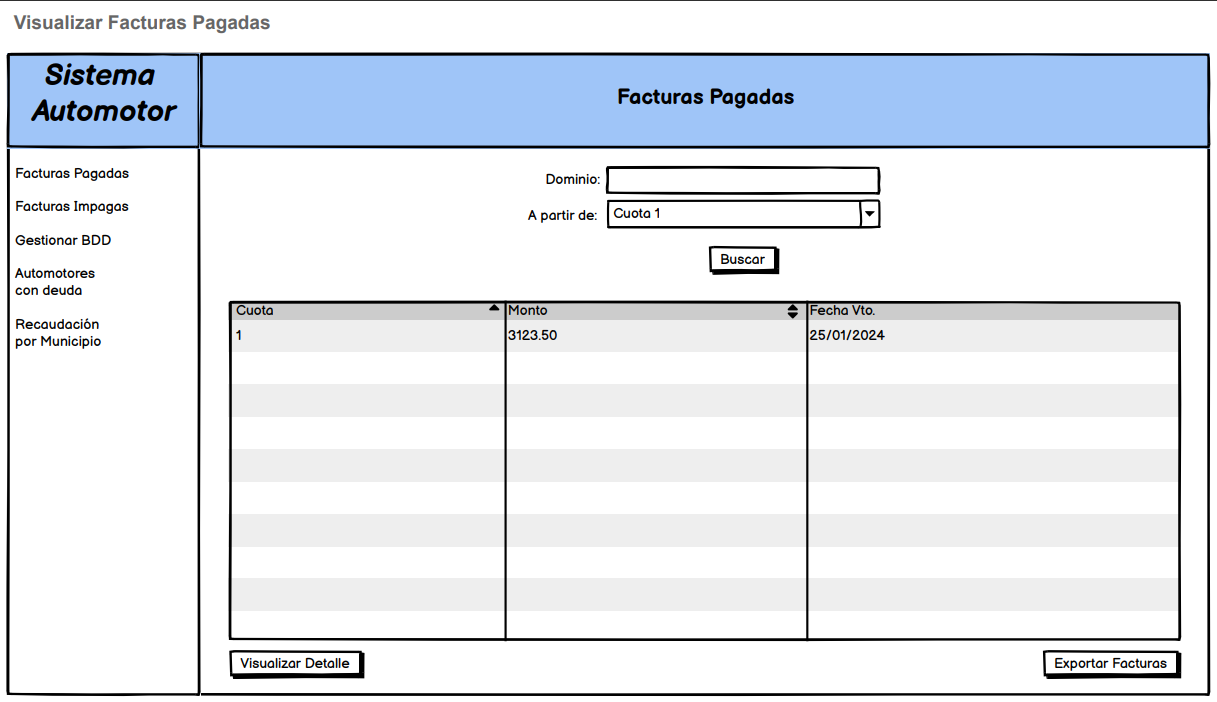


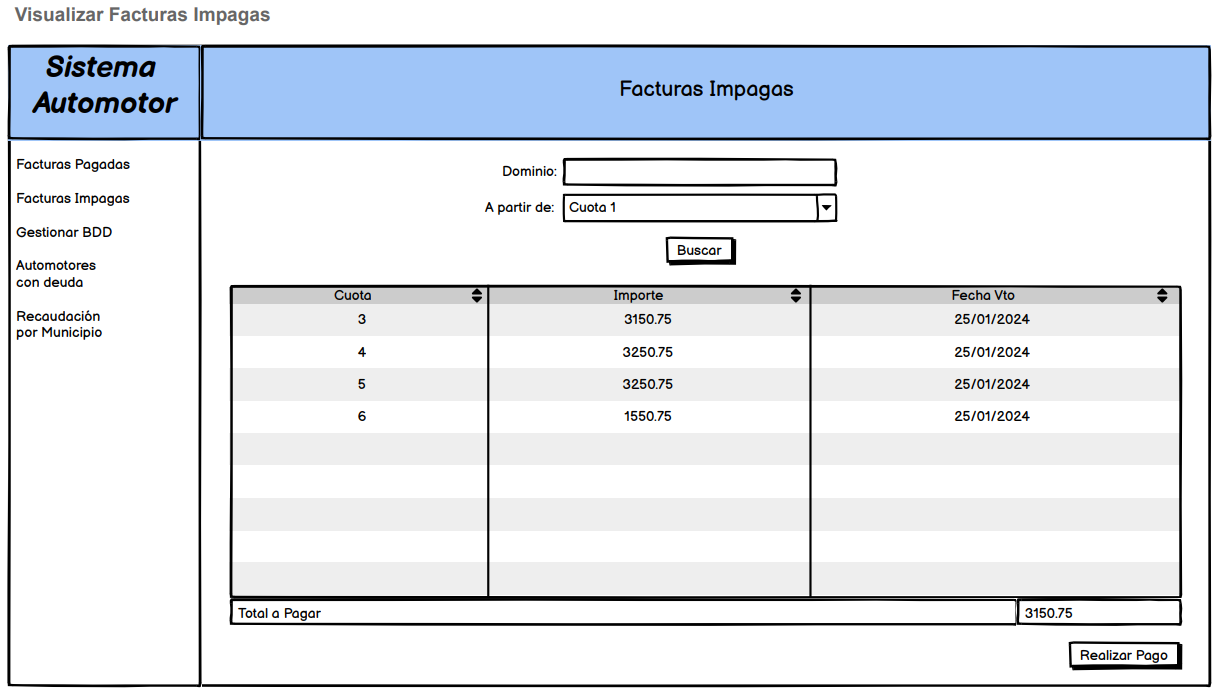


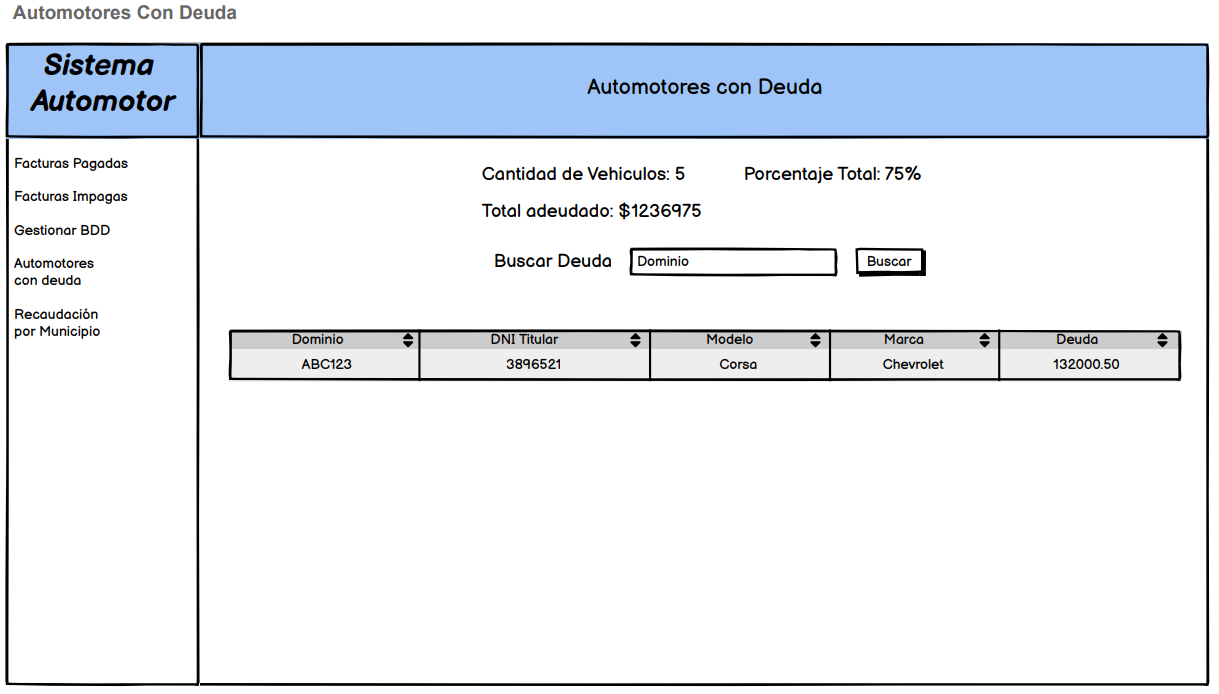


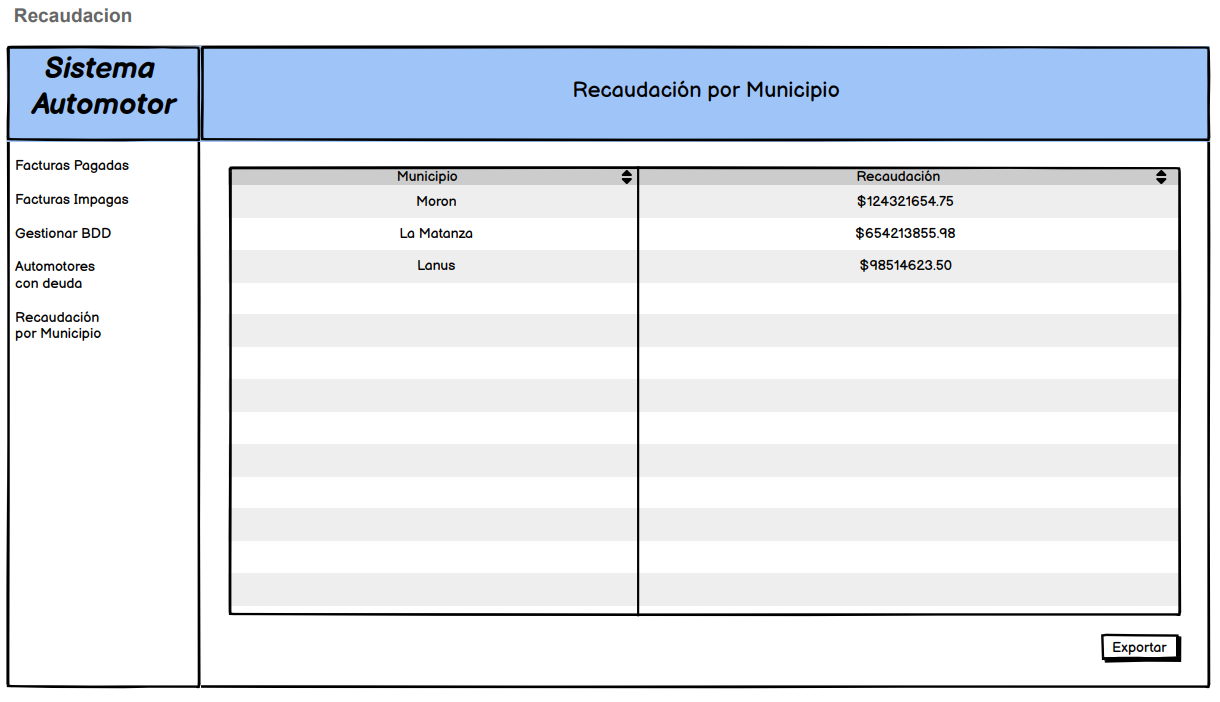
## 4.2 Wireframes Mockups

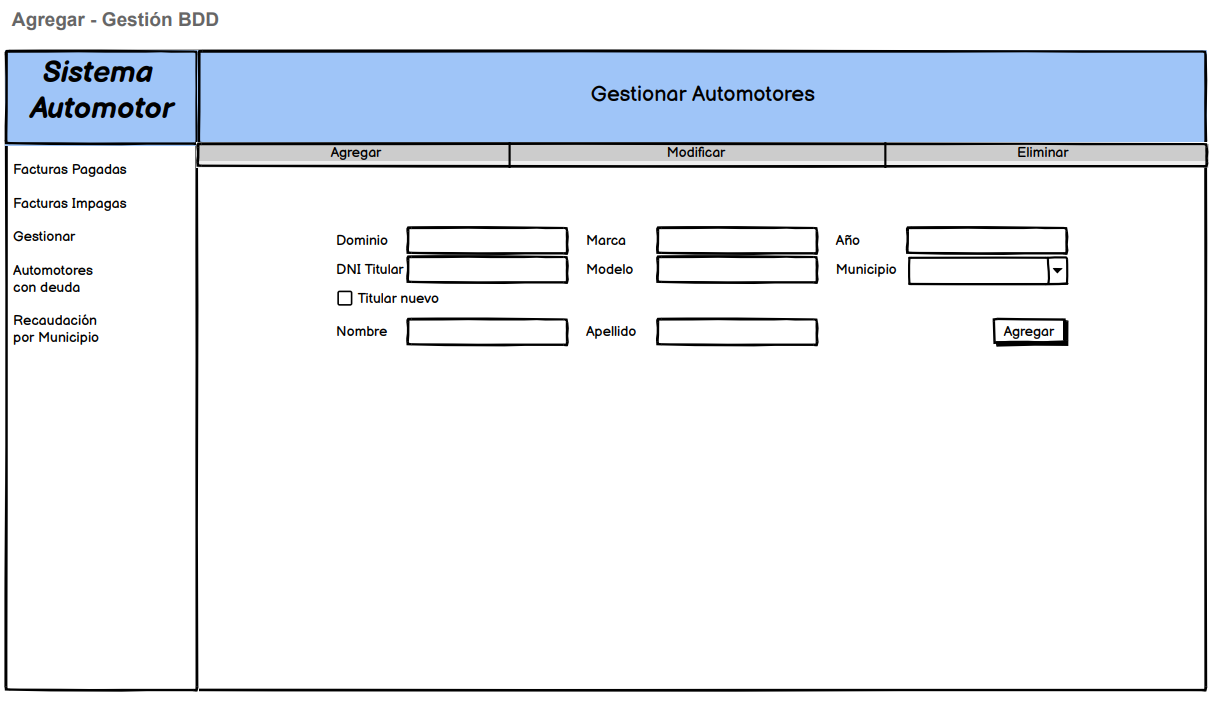
Presentaremos los Mockups trabajados en la aplicación

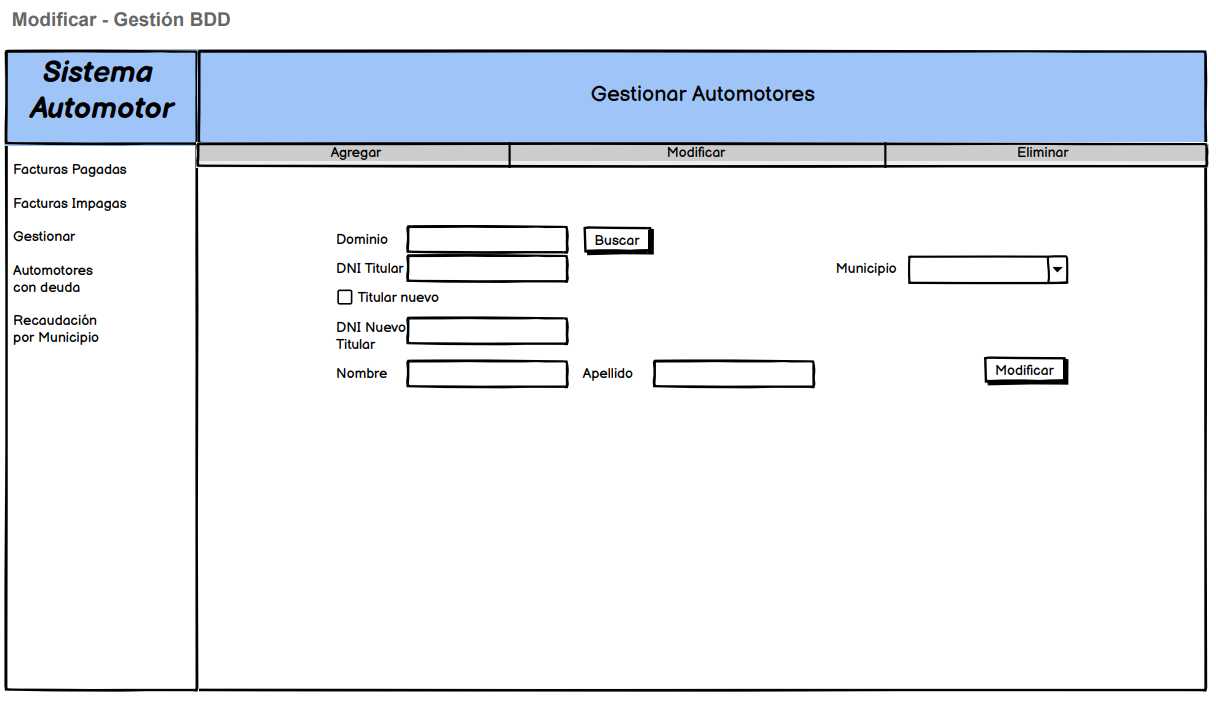


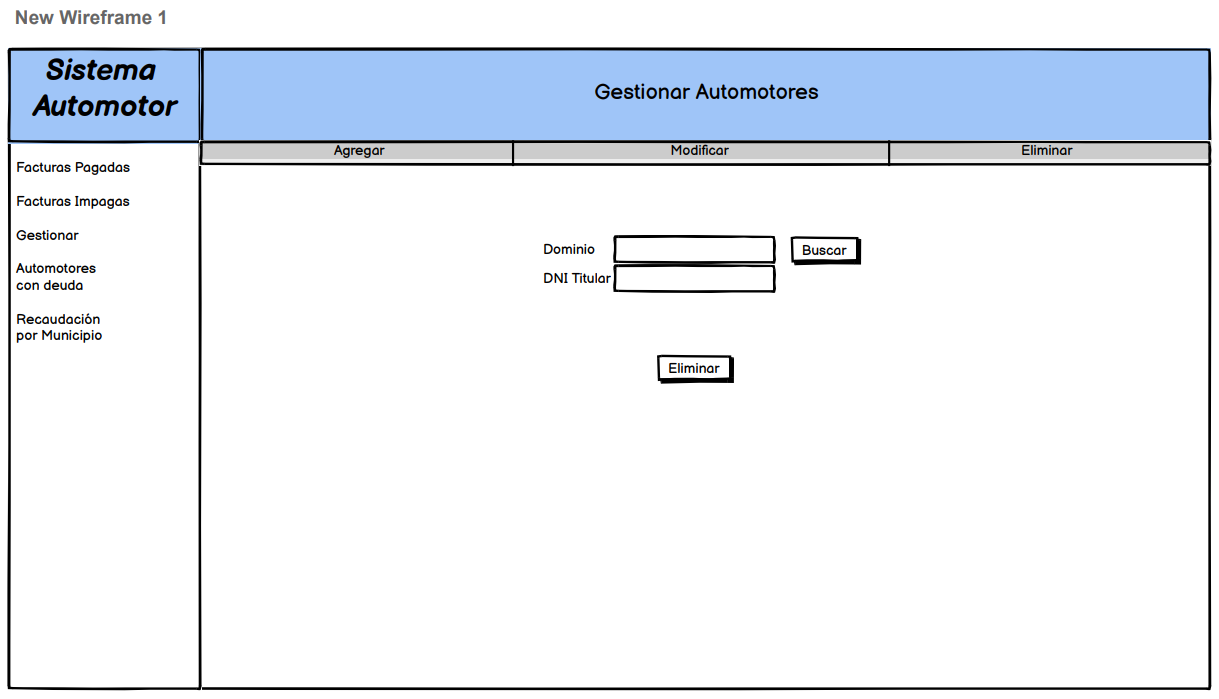












# 5. Arquitectura de Software

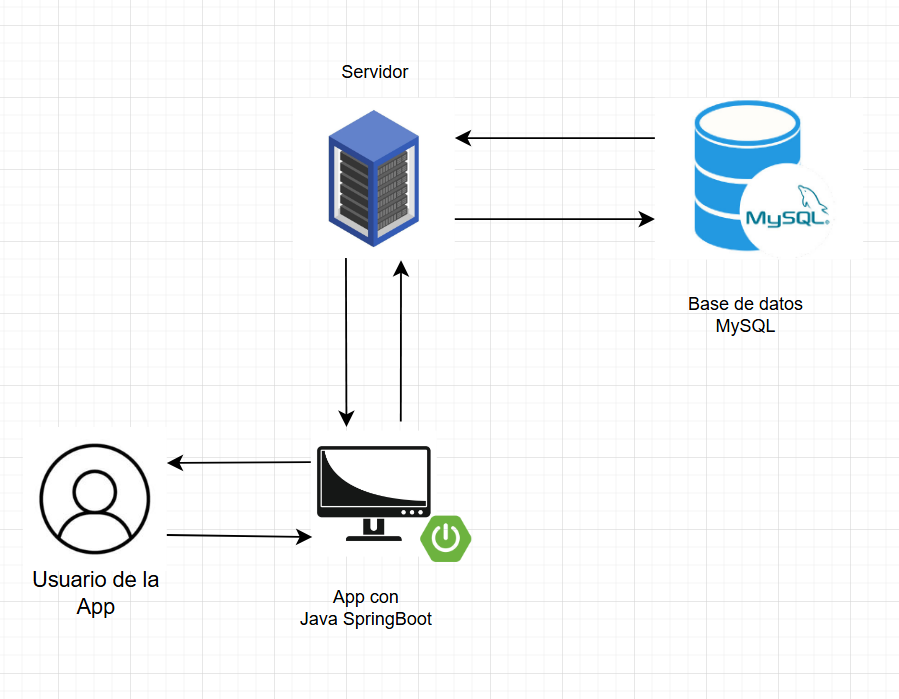
<versión>

<integrantes del grupo>

**Historia de Revisión**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 28/06/2024 | 1.0.6 | Diagramado sobre la comunicación entre componentes de la arquitectura de software. | Jorge Luis Lopez |
|  |  |  |  |

## 5.1 Diagrama de Arquitectura



## 5.2 Frameworks / Componentes utilizados

***Intellij -> para hacer la interfaz***

***MySqlServer***

***Java Spring Boot***

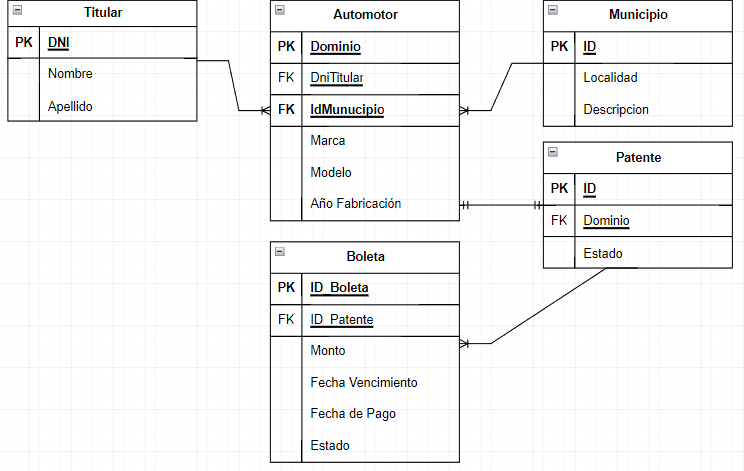
***Balsamiq wireframes***

## 5.3 Infraestructura tecnológica

| **Descripción** | *Implementamos un sistema de Automotores*  *basándonos en las condiciones y*  *restricciones que nos impuso el cliente* | | |
| --- | --- | --- | --- |
| **Sistema Operativo** | *Windows 10* | | |
| **Bases de Datos** | *MySQL* |  |  |
| **App Server / Web Server** | *Spring Boot y MySQL workbench CE* | | |
| **Lenguajes utilizados** | *Java* | *MySQL* |  |

## 5.4 Otros diagramas según corresponda

*DIAGRAMA DE ENTIDAD RELACION:*



## 5.5 Estándar de codificación

# 6. Plan de pruebas

**Historia de Revisión**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 28/06/2024 | 1.0.6 | Testeo de cada funcionalidad. | Federico Rossendy |
|  |  |  |  |

## 6.1 Diseño y ejecución de casos de prueba

Diseñamos el caso de

| Requerimiento | Escenario de prueba | Resultado |
| --- | --- | --- |
| Requerimiento 1 | Verificar que ante x situación, suceda | Aprobado/Fallo |
| ... | ... | ... |

Diseñamos el caso de Ingresar un Automotor.

|  |  |  |
| --- | --- | --- |
| Requerimiento | Escenario de prueba | Resultado |
| Ingresar un automotor por primera vez | La aplicación recibirá todos los datos del usuario, tales que no sean repetidos. | Aprobado -> se ingresó correctamente |
| Caso 1:  Se ingresa un dominio repetido | La aplicación rebotara el ingreso de los datos ya que ese automotor fue ingresado | Fallo-> el automotor ya fue ingresado |
| Caso 2:  Dominio que es mayor a 8 números/caracteres o menor a 6 | La aplicación rebotara el ingreso de los datos ya que este automotor no acepta un rango que no esté comprendido entre 6 a 8 | Fallo-> No lo ace |
| Caso 3:  Se ingresa un DNI titular que no existe | En este caso veremos si el sistema presentara  Si el nuevo dominio se le agregara al titular | Fallo-> perfecto, ya que si no encuentra el titular no se le agrega el auto |

Diseñamos casos de Modificar automotor

|  |  |  |
| --- | --- | --- |
| Requerimiento | Escenario de prueba | Resultado |
| Se ingresa el dominio, DNI y municipio correctamente | Se verifica los datos, buscamos el dominio y si este se encuentra, se traerá los datos del titular. | Aprobado-> se modifica correctamente |
| Caso 1:  Se ingresa un dominio que no se encuentra en el sistema | Se verifica si este tiene titular, si no lo tiene dará un error y dirá que el dominio no tiene ningún titular afiliado | Fallo-> se verifica que no existe titular con ese dominio |
| Caso 2:  No se ingresa ningún dato | Te pedirá que, por favor, ingreses un DNI. | Aprobado-> Caso borde. |

Diseñamos el caso de Eliminar Automotor

|  |  |  |
| --- | --- | --- |
| Requerimiento | Escenario de prueba | Resultado |
| Se requieren el dominio y DNI del titular. | Verifica que estos datos existan. | Aprobado -> se elimina el titular |
| Caso 1:  No se ingresa ningún dato | El sistema verificara que se ingresen datos, si no el sistema aborta la funcionalidad de eliminar y enviara un cartel de error | Fallo-> se debe ingresar un dominio. |
| Caso 2:  Se ingresa dominio, pero no DNI del titular | El sistema verificara que se ingresen datos, si no el sistema aborta la funcionalidad de eliminar y enviara un cartel de error | Fallo-> se debe ingresar el DNI del titular. |

Diseñamos el caso de Automotores con deuda

|  |  |  |
| --- | --- | --- |
| Requerimiento | Escenario de prueba | Resultado |
| Se ingresa un domino | El sistema verificara que este dominio exista y tenga deudas existentes | Aprobado-> nos devuelve una deuda. |
| Caso 1:  No se ingresa ningún dato | El sistema verificara que los datos este correctos, si no abortara la consulta con un cartel de error | Falla-> se debe ingresar un dominio |
| Caso 2:  Ingresa un dominio sin deudas | El sistema verificara que este dominio tengas deudas pendientes, si no le abortara la consulta con un cartel de error | Falla-> Ingrese un automotor con deudas pendientes. |
| Caso 3:  Ingresa un dominio erróneo | El sistema verificara que este dominio tengas deudas pendientes o exista, si no le abortara la consulta con un cartel de error | Fallo->No se encontró el dominio ingresado. |

Diseñamos el caso de Facturas Impagas

|  |  |  |
| --- | --- | --- |
| Requerimiento | Escenario de prueba | Resultado |
| Se ingresa un domino con facturas a pagar. | El sistema le traerá una lista si es que el dominio tiene alguna factura impaga | Aprobado-> La lista le trae las facturas impagas. |
| Caso 1:  Se ingresa un domino con facturas impagas y desea pagar. | El sistema le traerá una lista si es que el dominio tiene alguna factura impaga y el cliente decidirá cuál de estas pagará. | Aprobado-> La lista le trae las facturas impagas y le deja pagar correctamente una de estas. |
| Caso 2:  No se ingresa ningún dominio | El sistema verificara que el domino exista, si no abortara la funcionalidad con un cartel de error. | Fallo-> Ingrese el dominio. |
| Caso 3:  Ingresa un domino erróneo | El sistema verificara que el domino exista, si no abortara la funcionalidad con un cartel de error. | Fallo-> No se encontraron facturas para el dominio ingresado. |

Diseñamos el caso de Facturas Pagadas

|  |  |  |
| --- | --- | --- |
| Requerimiento | Escenario de prueba | Resultado |
| Se ingresa un domino con facturas pagadas | El sistema le traerá una lista si es que el dominio tiene alguna factura pagada. | Aprobado-> La lista le trae las facturas pagadas. |
| Caso 1:  Se ingresa un domino con facturas pagadas | El sistema le traerá una lista si es que el dominio tiene alguna factura pagada. | Aprobado-> La lista le trae las facturas pagadas. |
| Caso 2:  No se ingresa ningún dominio | El sistema verificara que el domino exista, si no abortara la funcionalidad con un cartel de error. | Fallo-> Ingrese el dominio. |
| Caso 3:  Ingresa un domino erróneo | El sistema verificara que el domino exista, si no abortara la funcionalidad con un cartel de error. | Fallo-> No se encontraron facturas para el dominio ingresado. |

## 6.2 Seguimiento de fallas

No usamos seguimiento de fallas, ninguna herramienta

# Anexos I. Informe de avance

**Historia de Revisión**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 04/07/2024 | 1.0.7 | Emprolijamiento de los anteriores historiales de revisión y finalización de informes de avances | Federico Loiero |
|  |  |  |  |

*INFORME DE AVANCE N.º 1:*

| **PROYECTO** | **GRUPO N.º** | **FECHA** |
| --- | --- | --- |
|  | 2 | 11/5/2024 |

***TAREAS***

| FINALIZADAS | ESTA ENTREGA | | OBSERVACIONES |
| --- | --- | --- | --- |
| * Vista Preliminar * Alcance * Limite y Restricciones * Especificación de Requisitos * Requerimientos Funcionales * Modelo de Casos de Uso * Modelo de Casos de Uso * Actores Casos de Uso * Descripción Casos de Uso * Arquitectura de Software * Framework y componentes Utilizados * Arquitectura de Tecnología * Otros Modelos | * Creación de base de datos * Conexión del proyecto Java a MySQL * Creación de Tablas * Creación del actor Sistema Automotor * Creación del caso de uso generar boleta * Detalle de la infraestructura tecnológica | |  |
| PRÓXIMAS TAREAS | | OBSERVACIONES | |
|  | |  | |

***PROBLEMAS PRESENTADOS – SOLUCIONES APLICADAS***

| PROBLEMA | DETALLES |
| --- | --- |
|  |  |

*INFORME DE AVANCE N.º 2:*

| **PROYECTO** | **GRUPO N.º** | **FECHA** |
| --- | --- | --- |
|  | 2 | 12/05/2024 |

***TAREAS***

| FINALIZADAS | ESTA ENTREGA | | OBSERVACIONES |  |
| --- | --- | --- | --- | --- |
| * Mockup de casos de usos + nuevo framework utilizado * Bocetos en Balsamiq wireframes * Interfaz de usuario, administrador y gerente en la aplicación | * Interfaz tentativa de caso de uso | |  |  |
| PRÓXIMAS TAREAS | | OBSERVACIONES | |  |
|  | |  | |  |

***PROBLEMAS PRESENTADOS – SOLUCIONES APLICADAS***

| PROBLEMA | DETALLES |
| --- | --- |
| * Se presenta un problema "AUTOMOTORES NO PAGADOS" | Dudas sobre el filtro que se realizara con respecto a los ítems presentados por el enunciado (puntualmente el ítem 6) |

*INFORME DE AVANCE N.º 3:*

|  |  |  |
| --- | --- | --- |
| **PROYECTO** | **GRUPO Nº2** | **FECHA** |
| Sistema Automotor | 2 | 16/5/2024 |

***TAREAS***

|  |  |  |  |
| --- | --- | --- | --- |
| FINALIZADAS | ESTA ENTREGA | | OBSERVACIONES |
| * Ventana de Automotores para administrador * Visualización de la interfaz y pruebas unitarias * Ventana de Pagos para el usuario * Visualización de la interfaz y pruebas unitarias | * Creación de la ventana, funcional y vinculado con la data base. | |  |
| PRÓXIMAS TAREAS | | OBSERVACIONES | |
| * Tareas del gerente, realizarle una interfaz propia a la ventana y seguir con los demás casos de uso | |  | |

***PROBLEMAS PRESENTADOS – SOLUCIONES APLICADAS***

|  |  |
| --- | --- |
| PROBLEMA | DETALLES |
| * Problemas con la data base, puntualmente foreing keys * Aparición de nuevos casos de uso * Relación con los demás casos de uso existentes * Botón de regresar para la interfaz de Pagos * Generar una pregunta antes de realizar la eliminación en caso de que este no haya querido que sea así |  |

***TAREAS***

|  |  |  |  |
| --- | --- | --- | --- |
| FINALIZADAS | ESTA ENTREGA | | OBSERVACIONES |
| * Especificación de Requisitos * Revisión Requerimientos Funcionales * Modelo de Casos de Uso * Revisión Modelo de Casos de Uso * Revisión Actores Casos de Uso * Revisión Descripción Casos de Uso | Revisión del modelo de caso de uso, requerimientos funcionales | |  |
| PRÓXIMAS TAREAS | | OBSERVACIONES | |
| Revisión de Interfaz | |  | |

***PROBLEMAS PRESENTADOS – SOLUCIONES APLICADAS***

|  |  |
| --- | --- |
| PROBLEMA | DETALLES |
|  |  |

*INFORME DE AVANCE N.º 5:*

|  |  |  |
| --- | --- | --- |
| **PROYECTO** | **GRUPO N.º** | **FECHA** |
| Sistema Automotor | 2 | 22/5/2024 |

 Taller de Integración

|  |  |
| --- | --- |
|  | 12 |

***TAREAS***

|  |  |  |  |
| --- | --- | --- | --- |
| FINALIZADAS | ESTA ENTREGA | | OBSERVACIONES |
| * Arreglo de facturas impagas y pagadas * Reestructuración de automotores con deudas | Interfaz + funcionalidad para ver cada auto y porcentaje | |  |
| PRÓXIMAS TAREAS | | OBSERVACIONES | |
| Revisión de Gestionar automotores | |  | |

***PROBLEMAS PRESENTADOS – SOLUCIONES APLICADAS***

|  |  |
| --- | --- |
| PROBLEMA | DETALLES |
|  |  |

*INFORME DE AVANCE N.º 6:*

|  |  |  |
| --- | --- | --- |
| **PROYECTO** | **GRUPO N.º** | **FECHA** |
| Sistema automotor | 2 | 29/06/2024 |

***TAREAS***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FINALIZADAS | ESTA ENTREGA | | OBSERVACIONES |  |
| * Finalización de requerimientos lógicos de la base de datos * Diagrama de arquitectura tentativo * Mockups y diseños principales * Aplicaciones de diseños y estilos de la GUI * Determinamos estilos sobre temáticas UNLAM * Gestión (Automotor) terminada * ¡TERMINAMOS REALIZAR PAGO! | * Propuestas sobre la GUI * Finalización de todas las funcionalidades del sistema automotor | |  |  |
| PRÓXIMAS TAREAS | | OBSERVACIONES | |  |
| * Finalizar los estilos de la aplicación y terminar prototipo de navegación * Test unitarios | |  | |  |

***PROBLEMAS PRESENTADOS – SOLUCIONES APLICADAS***

|  |  |
| --- | --- |
| PROBLEMA | DETALLES |
| * No se presenta ninguno. |  |