 **Proyecto Desarrollo Web**

**Presentado por:**

Iván Alejandro Martínez Gracia

**Presentado a:**

Jaime Andrés Pavlich Mariscal

**Pontificia Universidad Javeriana**

**Desarrollo Web**

**Bogotá, septiembre 7 de 2023**

**Sistema de Información Parqueadero**

Sistema de Información Parqueadero

**Introducción:**

Es un sistema de información para un parqueadero. El parqueadero está compuesto por varios pisos, en cada uno de los cuales se guarda un tipo de vehículo (automóviles, buses, motos). Puede haber más de un piso para un mismo tipo de vehículo. Dicho ordenamiento lo revisa periódicamente el administrador según el porcentaje de ocupación que haya tenido cada piso en el mes anterior y se puede cambiar. Todos los pisos son de igual tamaño (rectangulares), y el número de vehículos que puede parquearse en cada uno de ellos depende del tipo de vehículo

**Casos de uso y actores:**

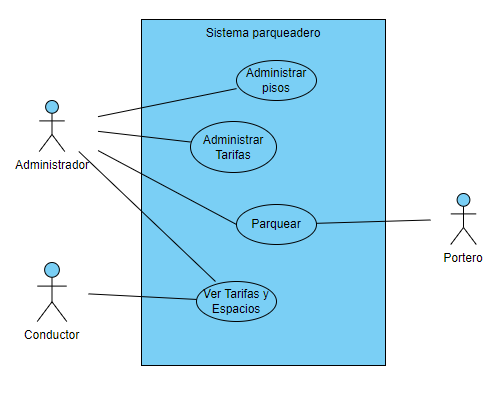
**Administrador:**

* Administrar Pisos: Permite al administrador realizar las operaciones CRUD (Crear, Leer, Actualizar y Eliminar) en los pisos del parqueadero, incluyendo la especificación del tipo de vehículo para cada piso y el cálculo de espacios disponibles.
* Administrar Tarifas: Permite al administrador especificar las tarifas por minuto para cada tipo de vehículo.

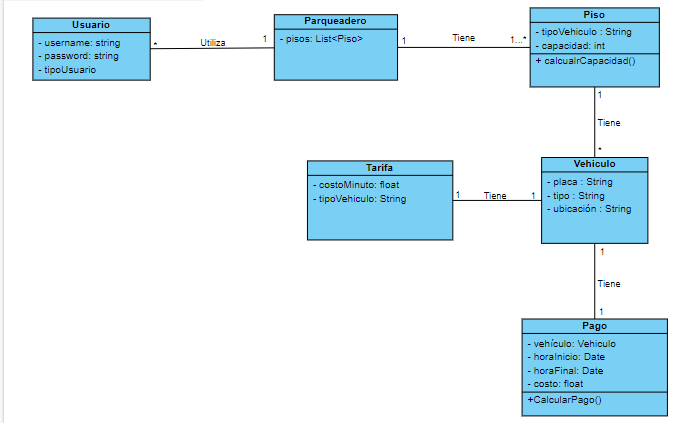
**Portero:**

* Parquear: Permite al portero admitir un vehículo en el parqueadero, asignándole un piso según la disponibilidad de espacios. También registra el piso utilizado por el vehículo y calcula los espacios disponibles. Si el parqueadero está lleno, no se permite el acceso.

**Conductor:**

* Ver Tarifas y Espacios: Permite al conductor ver las tarifas por tipo de vehículo y los espacios disponibles por tipo de vehículo

**Diagrama de clases:**

****

**Modelo lógico base de datos:**

**Tablas principales:**

**Usuario**

ID (Clave primaria)

Username (Nombre de usuario)

Password (Contraseña encriptada)

TipoUsuario (Administrador, Portero, Conductor)

**Piso**

ID (Clave primaria)

Numero (Numero o identificador del piso)

TipoVehiculo (Automóvil, Bus, Moto)

Capacidad (Número máximo de vehículos)

EspaciosDisponibles (Número de espacios disponibles)

TarifaID (Clave foránea que hace referencia a la tabla de Tarifa)

**Vehiculo**

ID (Clave primaria)

Placa (Número de placa del vehículo)

TipoVehiculo (Automóvil, Bus, Moto)

FechaEntrada (Fecha y hora de entrada al parqueadero)

FechaSalida (Fecha y hora de salida del parqueadero)

PisoID (Clave foránea que hace referencia a la tabla de Piso)

**Precio**

ID (Clave primaria)

TipoVehiculo (Automóvil, Bus, Moto)

CostoPorMinuto (Costo por minuto de estacionamiento)

**Pago**

ID (Clave primaria)

VehiculoID (Clave foránea que hace referencia a la tabla de Vehiculo)

TiempoInicio (Hora de inicio de estacionamiento)

TiempoFin (Hora de fin de estacionamiento)

Costo (Monto total a pagar)

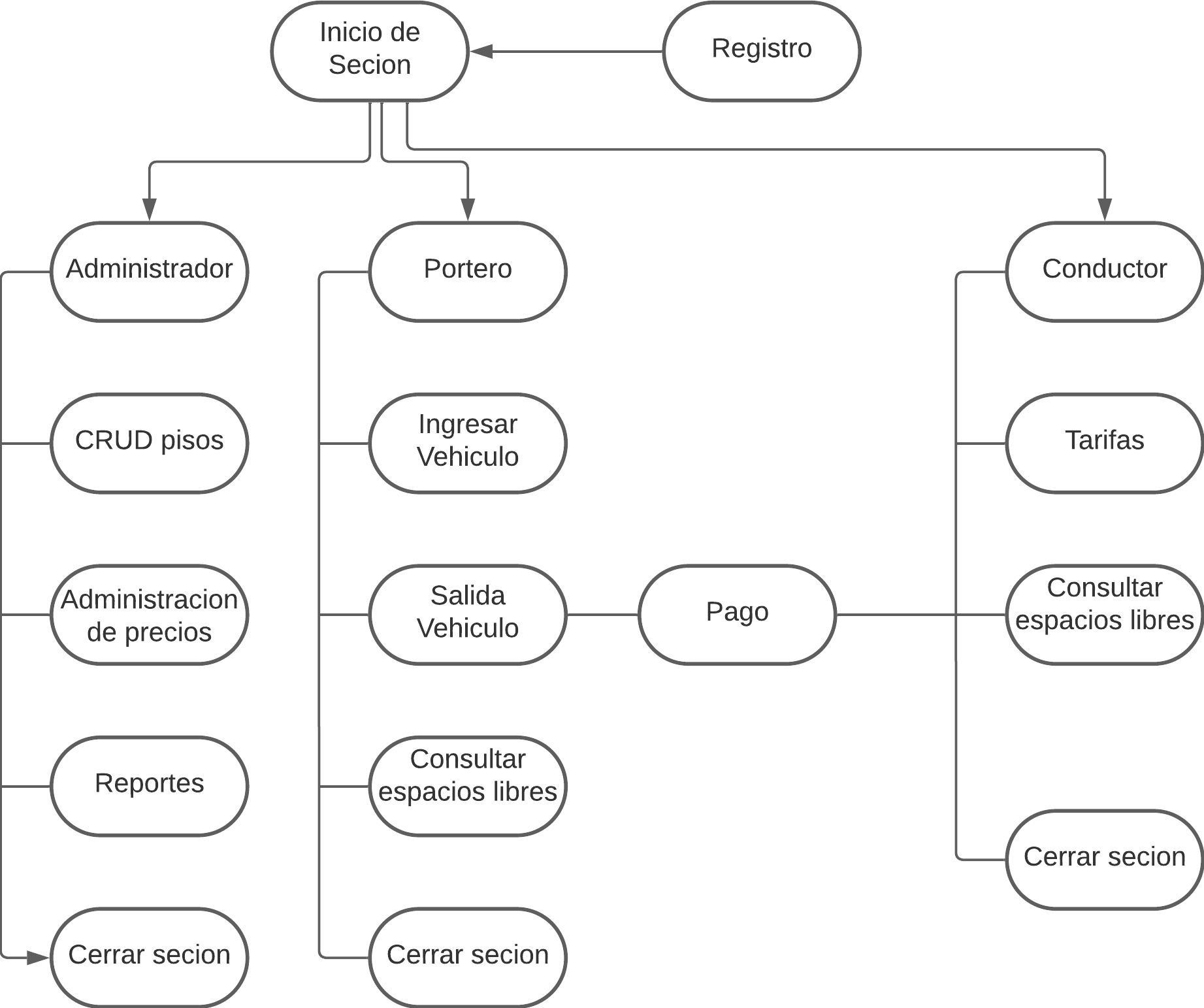
**Relaciones:**

La tabla Piso tiene una relación uno a uno con la tabla Tarifa a través de la columna TarifaID.

La tabla Vehiculo tiene una relación uno a uno con la tabla Piso a través de la columna PisoID.

La tabla Pago tiene una relación uno a uno con la tabla Vehiculo a través de la columna VehiculoID.

**Modelo de navegación:**



**MOCKUPS**

[**https://www.figma.com/file/MtSrO2v9d2n3Nj1MMySSnK/Untitled?type=design&node-id=0%3A1&mode=design&t=8Qe1wcUOqfQ13VTs-1**](https://www.figma.com/file/MtSrO2v9d2n3Nj1MMySSnK/Untitled?type=design&node-id=0%3A1&mode=design&t=8Qe1wcUOqfQ13VTs-1)

**Desarrollo e implementación:**

Para el desarrollo del proyecto se usó el framework SpringBoot como principal herramienta para el desarrollo de la aplicación web, como backend se utilizó java en su versión 20, y para el frontend HTML, CSS y JavaScript. Para la persistencia de datos se utilizo MySQL como gestor de bases de datos, realizando la conexión con los complementos de SpringBoot.

Para este proyecto se procuró manejar la arquitectura MVC, intentando realizar un proyecto lo mas organizado y con la mejor arquitectura posible.

Las pestañas están entre lazadas por el puerto 8080 en el localhost, de igual manera por defecto la base de datos está en el localhost, puerto 3306, bajo el nombre de bd\_crud, con las credenciales **usuario: root, contraseña root.**

Para el manejo de datos y pestañas se utilizaron los métodos Getmapping y Postmapping, los cuales envían o solicitan los valores correspondientes en cada caso, hacia la base de datos o de la base de datos. De igual manera al terminar la solicitud redirecciona o devuelve la ruta hacia la siguiente pestaña.

**VIDEO**

**https://youtu.be/mf\_7lc5F1BU**