

DICIEMBRE 2025



PROYECTO FINAL

FASE 1, 2 Y 3

Alumnos: Flores Albarrán Alexander & Merino López Juan Daniel.

Materia: Análisis y Diseño de Software.

Objetivo General

El objetivo de un estacionamiento es brindar un espacio seguro, ordenado y accesible para el resguardo de vehículos, facilitando la movilidad de los usuarios en sus actividades diarias y contribuyendo al control del orden vial de la zona. Además, busca optimizar la experiencia mediante la implementación de cobro automático con vía celular que evitan tiempos de espera, asignan un lugar de estacionamiento y permiten que el vehículo sea dirigido automáticamente a su espacio correspondiente.

Objetivos detallados

1. Gestión de Clientes y Precios

Desarrollar un sistema de gestión de clientes que asigne un código único a cada abonado mensual

- Almacene su historial de ingresos/salidas
- Aplicar descuentos en el rango del 10% al 30% en la tarifa mensual, en función de su antigüedad y tipo de vehículo (moto, coche estándar, coche grande).
- Implementar una funcionalidad para establecer y administrar tarifas dinámicas (por hora, por día, nocturna),
- Permitir que la configuración de una tarifa sea en base por minuto y un límite de costo máximo por día.

2. Control de Inventario y Ocupación

Implementar una base de datos

- Que registre cada espacio de estacionamiento por tipo (general, discapacitado, moto, carga eléctrica)
- Mantenga un historial de ocupación por zona y por categoría para facilitar el control y el análisis de la demanda.
- Incorporar un sistema de registro de entrada y salida
- Capturar la matrícula del vehículo mediante tecnología de reconocimiento óptico y que asigne automáticamente un ticket digital o código QR

3. Transacciones y Cumplimiento

Establecer un sistema de registro para cada transacción

- Almacenar la fecha y hora de entrada
- Almacenar la fecha y hora de salida.
- Calcule automáticamente el monto total a pagar, incluyendo impuestos y descuentos aplicables.
- Calcular una penalización que aplique un recargo del 50% sobre la tarifa diaria máxima para vehículos no abonados que permanezcan en el estacionamiento más de 72 horas sin movimiento o aviso.

4. Reportes y Accesibilidad

Crear reportes detallados para el administrador

- Incluir ingresos financieros (por tarifa y por abono).
- Estado de ocupación en tiempo real (disponibilidad por zona).
- Índice de rotación de vehículos por franja horaria.

5. Permitir que cada cliente abonado acceda a su reporte

- Historial de ingresos/egresos
- Facturación mediante una interfaz web accesible desde cualquier dispositivo móvil.
- Desarrollar una funcionalidad que envíe automáticamente una factura electrónica, al completar el pago de rotación o al inicio de cada ciclo de abono mensual.
- Asegurar que el sistema de gestión y la plataforma de acceso al cliente sean accesibles vía web desde un dominio propio.

6. Desarrollar un módulo de gestión de usuarios

- implementar un esquema de Control de Acceso Basado en Roles
- Permitir el registro, modificación y asignación de permisos a tres tipos de usuarios
- Cliente: Acceso restringido a su historial de ingresos/salidas, facturación y gestión de su abono personal.
- Administrador: Acceso total a los reportes operativos y financieros
- Gestión de tarifas y control del inventario de espacios.
- Super Usuario: Control absoluto sobre la configuración general del sistema
- Creación y modificación de cuentas de Administrador
- Acceso a funciones críticas de mantenimiento de la base de datos.
- Supervisor: Acceso a todos los reportes consolidados y la capacidad de revisar y auditar las acciones y los registros de los Administradores, pero sin capacidad para modificar configuraciones críticas del sistema.

Definición de objetivos claros

- El sistema debe registrar y alertar al super usuario, cuando exista alguna falla con el mecanismo
- El sistema deberá de registrar la cantidad de vehículos que ingresaron al día
- El sistema debe gestionar al menos 100 espacios de estacionamiento y 50 usuarios registrados en un plazo máximo de tres semanas."

Control de acceso automatizado

- El sistema debe permitir la entrada y salida de vehículos mediante un control automático con código QR o tarjeta en un plazo máximo de un mes.

Interfaz de usuario

- El sistema debe ofrecer una interfaz web y móvil intuitiva que permita reservar y pagar un espacio de estacionamiento en menos de 2 minutos.

Reportes y control administrativo

- "El sistema debe generar reportes diarios de ocupación y cobros en formato PDF o Excel de forma automática, dentro de los primeros 15 días de uso."

Seguridad y confiabilidad

- "El sistema debe garantizar que el 95% de los accesos de vehículos y usuarios se registren sin errores durante las primeras seis semanas de funcionamiento."

Módulos que formarán parte del sistema

En esta tarea se hace un desglose funcional del sistema en diferentes módulos o componentes. Esto permite dividir el trabajo y asignar responsabilidades claras a cada miembro del equipo.

1. Módulo de gestión de usuarios

- Funciones: Registro, edición, eliminación de usuarios. Permitir la búsqueda de usuarios por ID o nombre.
- Actores: Cliente, Administrador, Super usuario, Supervisor

Módulo de gestión de Espacios

- Funcionalidad: Asignar espacios, editar espacios (por casos de que estén ocupados o en reparación), eliminar espacios
- Actores: Administrador

Módulo de Renta

- Funcionalidad: Registrar la renta del espacio, verificar la disponibilidad de los lugares, registrar el pago de la renta del lugar
- Actores: Administrador y Cliente

Módulo de reportes

- Funcionalidad: Generar reportes sobre la renta de espacios, espacios ocupados (por semana o mes), lugares en mantenimiento o reparación
- Actores: Supervisor

Asignar roles a los miembros de equipo

En esta fase se hará la asignación de tareas dependiendo el rol dentro de la operación

Analista de sistemas

- Recopilar los requisitos del cliente (administradores del estacionamiento)
- Comprender sus necesidades y traducirlas en especificaciones técnicas claras que el equipo de desarrollo pueda seguir.
- Coordinar con los demás miembros del equipo para asegurarse de que los objetivos del sistema (gestión de espacios, cobros automáticos, reportes, accesos, etc.) sean realistas y alcanzables

Diseñador de Software

- Crea los planos técnicos del sistema, como los diagramas UML, la arquitectura del sistema, la estructura de la base de datos (usuarios, espacios, tarifas, reportes)
- modelos de interacción entre los módulos del estacionamiento.
- Definir los patrones de diseño a utilizar y cómo se implementarán las funcionalidades: asignación de lugares, control de accesos con QR, cálculo de tarifas, reportes automáticos, etc

Desarrolladores

- Implementar el código para las funcionalidades del sistema del estacionamiento.
- desarrolladores encargados de diferentes módulos (gestión de usuarios, control de espacios, renta y cobros, generación de reportes).
- coordinación con el analista y el diseñador para asegurarse de que el código cumpla los requisitos como registro de vehículos, control de barreras automáticas, cobro de tarifas y reportes de ocupación

Probador de Software

- Realizar las pruebas del sistema, tanto a nivel unitario como a nivel de integración.
- Crear un plan de pruebas que cubra todos los casos de uso del estacionamiento como (entrada/salida de vehículos, registro de usuarios, cobro automático, generación de facturas)
- Reportar errores o problemas en la funcionalidad para que los desarrolladores los corrijan antes de la entrega final.

Líder de proyecto o coordinador

- Coordina y supervisa el trabajo del equipo garantizando que se cumplan los plazos y los objetivos definidos en el proyecto del estacionamiento.
- Actúa como puente entre el cliente (administración del estacionamiento) y el equipo técnico.
- gestionar los recursos (tiempo, personal, herramientas) y resolver problemas que puedan surgir a lo largo del desarrollo

CRONOGRAMA

| A | B | C | D | E |
|---------------------------------|---|--------------------------------|----------------|--|
| CRONOGRAMA DE ACTIVIDADES | | | | |
| FASE | ACTIVIDADES CLAVE | ROL PRINCIPAL | PLAZO DEFINIDO | META |
| 1.- PLANIFICACION Y DISEÑO | RECOPIACION DE REQUISITOS Y DEFINICION DE ESPECIFICACIONES TECNICAS | ANALISTA DE SISTEMAS | SEMANA 1 | ESPECIFICACIONES TECNICAS FINALES |
| | CREACION DE PLANOS TECNICOS (DIAGRAMAS UML Y BASE DE DATOS) | DISEÑADOR DE SOFTWARE | SEMANA 2-3 | DISEÑO DE ARQUITECTURA Y BASE DE DATOS |
| 2.- DESARROLLO Y IMPLEMENTACION | IMPLEMENTACION DE MODULOS DE GESTION DE ESPACIOS Y BASE DE DATOS DE INVENTARIO | DESARROLLADORES | SEMANA 3-4 | MODULO DE INVENTARIOS DE ESPACIOS FUNCIONAL |
| | IMPLEMENTACION DE MODULO DE GESTION DE USUARIOS Y ROLES | SUPER-USUARIO | SEMANA 4-5 | MODULO DE USUARIOS FUNCIONAL CON CONTROL DE ACCESO POR ROLES |
| | IMPLEMENTACION DEL MODULO DE CONTROL DE ACCESO (RECONOCIMIENTO QR) | DESARROLLADORES | SEMANA 5-7 | CONTROL DE ENTRADA Y SALIDA AUTOMATIZADO |
| | IMPLEMENTACION DE LOGICA DE COBROS Y TRANSACCIONES (TARIFAS, DESCUENTOS Y PENALISACIONES | DESARROLLADOR | SEMANA 7-9 | SISTEMA DE CALCULO DE MONTOS Y TRANSACCIONES |
| | DESARROLLO DE INTERFAZ WEB Y MOVIL Y ENVIO DE FACTURA ELECTRONICA | DESARROLLADORES | SEMANA 9-11 | INTERFAS DE USUARIO Y FUNCIONALIDAD DE FACTURACION |
| 3.- PRUEBAS Y REPORTES | CREACION DE PLAN DE PRUEBAS Y PRUEBAS UNITARIAS Y DE INTEGRACION | TESTER | SEMANA 11-12 | PLAN DE PRUEBAS Y REPORTE INICIAL DE ERRORES |
| | DESARROLLO DEL MODULO DE REPORTES (FINANCIEROS, OCUPACION DE TIEMPO Y ROTAION) | DESARROLLADORES | SEMANA 12-13 | GENRACIO AUTOMATICA DE REPORTES PDF/ EXCEL |
| | FUNCIONALIDAD DE ACCESO AL CLIENTE (HISTORIAL DE INGRESOS, EGRESOS Y FACTURACION) | DESARROLLADORES | SEMANA 13-14 | INTERFA WEB DEL CLIENTE |
| 4.- DESPLIEGUE Y VALIDACION | DESPLIEGUE DEL SISTEMA EN DOMINIO PROPIO Y CONTROL DE CALIDAD INICIAL | DESARROLLADORES / SUPE USUARIO | SEMANA 14-15 | SISTEMA OPERATIVO Y ACCESIBLE MEDIANTE LA WEB |
| | PRUEBAS DE CONFIANZA Y ESTABILIDAD OPERACIONAL | TESTER | SEMANA 15-20 | VALIDACION DE FUNCIONALIDAD (ARRIVA DEL 90 %) |

| RECURSOS DEL SOFTWARE | | |
|--------------------------------|--------------------|--|
| Categoría | Recurso Específico | Propósito en el Proyecto |
| Lenguaje y Framework Principal | PYTHON | Lenguaje de programación principal para el backend. |
| | DJANGO | Framework web de alto nivel para el desarrollo rápido y Seguro del sistema de gestión (Módulos de Gestión de Usuarios, Espacios, Renta, y Reportes). |
| BASE DE DATOS | MY SQL | Sistema de gestión de bases de datos para almacenar el inventario de espacios, historial de ingresos/salidas, transacciones, y datos de clientes/usuarios. |
| Control de Versiones | Git y GitHub | Para el control de versiones del código y la colaboración del equipo. |
| Reportes y facturacion | Reporte Labs | Para la generación automática de reportes diarios de ocupación y cobros en formato PDF o Excel y el envío de la factura electrónica. |

| RECURSOS DE HARDWARE | |
|------------------------|---|
| Recurso | Funcion |
| Servidor de Producción | Servidor con suficiente RAM (mino 16 GB) y CPU para manejar transacciones y el almacenamiento de la base de datos. Debe asegurar que el sistema esté accesible vía web desde un dominio propio. |
| Dispositivos de Acceso | Cámaras con capacidad para el reconocimiento óptico de matrícula. Lectores de códigos QR o lectores de tarjetas para el control automático de entrada y salida. |
| Equipos de Desarrollo | Estaciones de trabajo para el equipo de desarrollo, diseñador de software, y probador de software. |

| Rol Definido Responsabilidades Clave | |
|--------------------------------------|--|
| Analista de Sistemas | Recopilar requisitos del cliente y traducirlos a especificaciones técnicas claras. Coordinar para asegurar que los objetivos sean realistas y alcanzables. |
| Diseñador de Software | Crear planos técnicos, diagramas UML, la arquitectura del sistema, y la estructura de la base de datos. Definir patrones de diseño y cómo se implementarán las funcionalidades. |
| Desarrolladores | Implementar el código para las funcionalidades del sistema (gestión de usuarios, control de espacios, renta y cobros, reportes). Asegurar que el código cumpla requisitos como registro de vehículos, control de barreras, cobro, y reportes de ocupación. |
| Probador de Software | Realizar las pruebas a nivel unitario y de integración. Crear un plan de pruebas que cubra todos los casos de uso (entrada/salida, cobro automático, facturación). Reportar errores para corrección. |
| Líder de Proyecto o Coordinador | Coordinar y supervisar el trabajo para garantizar que se cumplan los plazos y objetivos. Actuar como puente entre el cliente y el equipo técnico, y gestionar recursos y resolver problemas. |

FASE 2

Análisis de Requisitos del Sistema de Estacionamiento

A continuación, se clasifican los objetivos y características descritos en el documento del proyecto de estacionamiento en **Requisitos Funcionales** (qué hace el sistema) y **Requisitos No Funcionales** (cómo funciona el sistema).

1. Requisitos Funcionales

Estos describen las acciones y funcionalidades específicas que el sistema debe ser capaz de realizar.

Gestión de Clientes y Tarifas:

- Asignar un código único a cada cliente abonado.
- Almacenar el historial de ingresos y salidas de los clientes.

- Aplicar descuentos de entre 10% y 30% a clientes abonados.
- Permitir al administrador establecer y gestionar tarifas dinámicas (por hora, día, etc.).
- Configurar tarifas por minuto con un costo máximo por día.

Control de Acceso y Ocupación:

- Registrar cada espacio de estacionamiento por tipo (general, discapacitado, moto, etc.) en una base de datos.
- Mantener un historial de ocupación por zona y categoría.
- Capturar la matrícula del vehículo usando tecnología de reconocimiento óptico.
- Asignar automáticamente un ticket digital o código QR al ingresar un vehículo.
- Permitir la entrada y salida de vehículos mediante un control automático con código QR o tarjeta.

Transacciones y Facturación:

- Registrar cada transacción, almacenando fecha y hora de entrada y salida.
- Calcular automáticamente el monto total a pagar, aplicando impuestos y descuentos.
- Calcular y aplicar una penalización del 50% sobre la tarifa diaria a vehículos que permanezcan más de 72 horas sin movimiento.
- Enviar automáticamente una factura electrónica al cliente tras el pago.

Gestión de Usuarios y Roles:

- Permitir el registro, modificación y asignación de permisos a diferentes tipos de usuarios (Cliente, Administrador, Super Usuario, Supervisor).
- Restringir el acceso del "Cliente" solo a su historial y facturación.
- Permitir al "Administrador" el acceso total a reportes operativos y gestión de tarifas.

- Otorgar al "Super Usuario" control absoluto sobre la configuración del sistema y las cuentas de administrador.
- Permitir al "Supervisor" auditar las acciones de los administradores y revisar reportes consolidados.

Reportes y Alertas:

- Generar reportes detallados para el administrador sobre ingresos financieros, estado de ocupación y rotación de vehículos.
 - Registrar y alertar al super usuario cuando exista alguna falla con el mecanismo del sistema.
 - Registrar la cantidad de vehículos que ingresaron al día.
-

2. Requisitos No Funcionales

Estos describen las cualidades, características y restricciones del sistema, definiendo cómo debe operar.

Accesibilidad:

- El sistema de gestión y la plataforma del cliente deben ser accesibles vía web desde un dominio propio.
- La interfaz web para clientes debe ser accesible desde cualquier dispositivo móvil.

Usabilidad:

- El sistema debe ofrecer una interfaz web y móvil intuitiva.
- La interfaz debe permitir a un usuario reservar y pagar por un espacio de estacionamiento en menos de 2 minutos.

Rendimiento y Escalabilidad:

- El sistema debe ser capaz de gestionar un mínimo de 100 espacios de estacionamiento y 50 usuarios registrados en un plazo de tres semanas.

Confiabilidad y Seguridad:

- El sistema debe garantizar que el 95% de los accesos (de vehículos y usuarios) se registren sin errores durante las primeras seis semanas de operación.

Entrega y Formato:

- El sistema debe generar reportes diarios de ocupación y cobros de forma automática.
- Los reportes automáticos deben poder generarse en formato PDF o Excel

A CONTINUACION SE MUESTRAN LAS DEPENDENCIAS Y PRIORIDADES

| ID | Requisito | Categoría | Justificación de la Categoría | Dependencias (ID) |
|----|--|-----------------|--|-------------------|
| 1 | Implementar una base de datos para el sistema | Esencial | Es la base fundamental para almacenar toda la información del sistema (espacios, usuarios, transacciones). Sin ella, nada puede funcionar. | - |
| 2 | Funcionalidad para registrar y gestionar los espacios de estacionamiento | Esencial | El sistema debe saber qué "inventario" tiene. Es imposible asignar un vehículo si los espacios no existen en el sistema. | 1 |
| 3 | Registrar la fecha y hora de entrada y salida de cada vehículo | Esencial | Es el núcleo de la operación. Sin este registro, no se puede calcular el tiempo de estancia ni el costo. | 1, 2 |
| 4 | Calcular automáticamente el monto a pagar según el tiempo de estancia | Esencial | Es el propósito principal del sistema desde una perspectiva de negocio: monetizar el servicio. | 3 |

| | | | | |
|----|---|-----------------|--|----|
| 5 | Módulo básico de gestión de usuarios (Rol Administrador) | Esencial | Se necesita al menos un tipo de usuario para operar, configurar y supervisar el sistema. | 1 |
| 6 | Accesibilidad del sistema vía web para la administración | Esencial | El administrador necesita una interfaz para poder interactuar con el sistema y realizar sus funciones. | 5 |
| 7 | Automatización del acceso con reconocimiento de matrícula (OCR) y QR/tarjeta | Deseable | Aumenta la eficiencia y moderniza la experiencia, pero el registro de entrada/salida podría hacerse manualmente en una versión mínima. | 3 |
| 8 | Gestión completa de clientes abonados (código único, historial) | Deseable | Agrega un modelo de negocio recurrente y fideliza clientes, pero el sistema puede operar solo con clientes ocasionales. | 5 |
| 9 | Aplicación de descuentos para clientes abonados | Deseable | Es una característica de valor añadido para la gestión de abonados, pero no es vital para el funcionamiento básico. | 8 |
| 10 | Configuración de tarifas dinámicas (por hora, día, noche) | Deseable | Ofrece flexibilidad de negocio, pero una tarifa única y fija sería suficiente para una versión esencial. | 4 |
| 11 | Generación de reportes detallados para el administrador (ingresos, ocupación) | Deseable | Proporciona información clave para la toma de decisiones, pero el sistema puede operar transaccionalmente sin ellos. | 3 |
| 12 | Portal web para que los clientes consulten su historial y facturación | Deseable | Mejora la experiencia del cliente abonado y reduce la carga administrativa, pero no es indispensable. | 8 |
| 13 | Envío automático de facturas electrónicas | Deseable | Automatiza un proceso administrativo importante, pero la facturación podría | 12 |

| | | | | |
|----|---|-----------------|--|-------|
| | | | gestionarse por otros medios inicialmente. | |
| 14 | Implementación de roles avanzados (Super Usuario, Supervisor) | Deseable | Proporciona una capa de seguridad y organización más robusta, pero un único rol de Administrador es suficiente para empezar. | 5 |
| 15 | Aplicación de penalizaciones por estancia prolongada (+72h) | Deseable | Es una regla de negocio específica para optimizar el uso del espacio, pero no es parte de la transacción estándar. | 4 |
| 16 | Garantizar 95% de accesos sin errores en las primeras 6 semanas | Deseable | Es un requisito de calidad (no funcional). El sistema esencial debe funcionar, pero este define un alto estándar de confiabilidad. | Todos |
| 17 | Interfaz intuitiva que permita reservar y pagar en menos de 2 minutos | Deseable | Es un requisito de usabilidad. Una interfaz funcional es esencial, pero una altamente optimizada es deseable. | 12 |
| 18 | Direccionamiento automático del vehículo a su espacio correspondiente | Opcional | Es una funcionalidad de muy alta complejidad y costo. Aporta un gran valor, pero puede ser implementada en una fase muy posterior. | 7 |
| 19 | Aplicar descuentos basados en el tipo de vehículo (moto, coche) | Opcional | Añade una capa de complejidad a la lógica de precios que puede ser simplificada en las primeras versiones del sistema. | 9 |
| 20 | Mantener historial de ocupación por zona para análisis de demanda | Opcional | Es una función de inteligencia de negocio avanzada. Un reporte de ocupación en tiempo real es deseable; el análisis histórico es opcional. | 11 |

Documento de Requisitos: Sistema de Gestión de Estacionamiento

1. Descripción General del Sistema

El propósito del sistema es gestionar de manera eficiente y automatizada un estacionamiento para brindar un espacio seguro, ordenado y accesible para el resguardo de vehículos. El sistema busca optimizar la experiencia del usuario final mediante la implementación de tecnologías como el reconocimiento de matrículas, el acceso con códigos QR y la gestión de pagos y facturación vía web. Para la administración, el sistema proveerá herramientas para la gestión de clientes, tarifas, ocupación y la generación de reportes financieros y operativos.

2. Requisitos Funcionales

A continuación, se detallan las funcionalidades que el sistema debe implementar.

Módulo: Control de Acceso y Transacciones

- **ID:** RF-01
- **Descripción:** El sistema debe registrar la entrada de un vehículo capturando su matrícula y asignando un ticket digital o código QR.
- **Criterios de Aceptación:**
 - Se puede registrar una matrícula de forma automática (OCR) o manual.
 - El sistema registra la fecha y hora exacta de entrada y la asocia a la matrícula.
 - El espacio de estacionamiento asignado cambia su estado a "Ocupado".
 - Se genera un ticket digital único para la transacción.
- **Prioridad:** Alta
- **ID:** RF-02
- **Descripción:** El sistema debe registrar la salida de un vehículo, calcular el monto total a pagar y procesar el pago.

- **Criterios de Aceptación:**
 - El sistema recupera los datos de entrada usando el ticket digital o la matrícula.
 - El cálculo del costo total basado en el tiempo de estancia y la tarifa aplicable es correcto.
 - Cualquier descuento de cliente abonado se aplica correctamente.
 - Una vez realizado el pago, el sistema registra la transacción como "completada" y libera el espacio de estacionamiento.
- **Prioridad:** Alta

Módulo: Gestión de Clientes y Abonos

- **ID:** RF-03
- **Descripción:** El sistema debe permitir la gestión de clientes abonados, asignando un código único y almacenando su historial.
- **Criterios de Aceptación:**
 - Un administrador puede crear, editar y eliminar perfiles de clientes abonados.
 - Cada cliente abonado tiene un código único en el sistema.
 - Todos los ingresos y salidas del abonado se registran en su historial.
- **Prioridad:** Media
- **ID:** RF-04
- **Descripción:** El sistema debe aplicar descuentos en la tarifa mensual a los clientes abonados en función de su antigüedad y tipo de vehículo.
- **Criterios de Aceptación:**
 - El administrador puede configurar las reglas para los descuentos (ej. >1 año = 15% de descuento).
 - El sistema aplica el descuento correcto automáticamente al generar el cobro mensual del abono.
- **Prioridad:** Media

Módulo: Administración y Reportes

- **ID:** RF-05
- **Descripción:** El sistema debe permitir al administrador configurar tarifas dinámicas (por hora, día, nocturna) con un costo máximo por día.
- **Criterios de Aceptación:**
 - Existe una interfaz donde el administrador puede crear, modificar y activar diferentes esquemas de tarifas.
 - El sistema utiliza la tarifa activa correcta al momento de calcular el costo de una estancia.
 - El cobro a un vehículo no excede el límite de costo máximo por día establecido.
- **Prioridad:** Media
- **ID:** RF-06
- **Descripción:** El sistema debe generar reportes detallados para el administrador, incluyendo ingresos financieros, estado de ocupación y rotación de vehículos.
- **Criterios de Aceptación:**
 - El administrador puede seleccionar un rango de fechas para generar los reportes.
 - El reporte de ingresos muestra un desglose correcto de los pagos por abono y por rotación.
 - El reporte de ocupación muestra la disponibilidad por zona en tiempo real.
 - Los reportes se pueden exportar en formato PDF o Excel.
- **Prioridad:** Media
- **ID:** RF-07
- **Descripción:** El sistema debe permitir a los clientes abonados acceder a su historial de ingresos/egresos y facturación a través de una interfaz web.

- **Criterios de Aceptación:**
 - El cliente puede iniciar sesión de forma segura en un portal web.
 - Dentro del portal, el cliente puede ver una lista de todas sus visitas pasadas, con fechas y costos.
 - El cliente puede ver y descargar sus facturas mensuales.
- **Prioridad:** Media

3. Requisitos No Funcionales

- **Accesibilidad:** El sistema de gestión y la plataforma del cliente deben ser accesibles vía web desde un dominio propio y desde cualquier dispositivo móvil.
- **Usabilidad:** La interfaz web y móvil debe ser intuitiva, permitiendo a un cliente reservar y pagar por un espacio en menos de 2 minutos.
- **Rendimiento:** El sistema debe ser capaz de gestionar al menos 100 espacios de estacionamiento y 50 usuarios registrados de manera eficiente.
- **Confiability:** El sistema debe garantizar que el 95% de los accesos de vehículos y usuarios se registren sin errores durante las primeras seis semanas de funcionamiento.
- **Seguridad:** El sistema debe implementar un esquema de Control de Acceso Basado en Roles para asegurar que los usuarios solo puedan acceder a las funciones permitidas para su perfil (Cliente, Administrador, Supervisor, Super Usuario)

Plan de Revisión y Aprobación de Requisitos

1. Proceso de Revisión de Requisitos con el Cliente

Para asegurar que no existan malentendidos y que todas las expectativas estén cubiertas, se llevará a cabo una sesión de revisión formal con el cliente.

- **Participantes:**
 - **Por parte del equipo de desarrollo:** El Líder de Proyecto y el Analista de Sistemas.
 - **Por parte del cliente:** Representantes de la administración del estacionamiento.
- **Agenda de la Reunión:**
 1. **Presentación del Documento de Requisitos:** Se expondrá el documento formalizado (creado en el paso anterior) que contiene la descripción del sistema, los requisitos funcionales y no funcionales.
 2. **Revisión Detallada:** Se analizará cada requisito uno por uno, explicando su descripción, criterios de aceptación y prioridad.
 3. **Sesión de Aclaraciones:** Se abrirá un espacio para que el cliente pueda hacer preguntas, expresar dudas y asegurar que los requisitos reflejan fielmente sus necesidades.
 4. **Recopilación de Feedback:** Se documentará toda la retroalimentación, sugerencias o posibles ajustes solicitados por el cliente.
 5. **Acuerdo de Aprobación:** Al finalizar la revisión, se buscará la aprobación formal ("sign-off") del documento por parte del cliente para poder congelar el alcance y proceder con la fase de diseño.

2. Propuesta de Prototipos y Mockups

Para facilitar la visualización del sistema y obtener una retroalimentación más precisa, se crearán y presentarán los siguientes mockups de la interfaz de usuario, correspondiendo a los roles de usuario clave del sistema:

- **Mockup 1: Panel de Control del Administrador**
 - **Propósito:** Mostrar al cliente cómo el "Administrador" gestionará las operaciones diarias.

- **Contenido:**
 - Una vista principal con el estado de ocupación en tiempo real (disponibilidad por zona).
 - Accesos directos para la gestión de tarifas dinámicas.
 - Un módulo para generar los reportes financieros y de rotación de vehículos.
- **Mockup 2: Portal Web del Cliente Abonado**
 - **Propósito:** Ilustrar la experiencia del "Cliente" que tiene un abono mensual.
 - **Contenido:**
 - Pantalla de inicio de sesión segura.
 - Una vista donde el cliente pueda consultar su historial de ingresos/egresos.
 - Un apartado para visualizar y descargar su facturación mensual.
- **Mockup 3: Interfaz de Entrada y Salida (Quiosco)**
 - **Propósito:** Simular la interacción de un usuario (abonado o no) en los puntos de acceso y salida del estacionamiento.
 - **Contenido:**
 - Una pantalla de bienvenida que muestra la matrícula capturada por la cámara de reconocimiento óptico.
 - Una simulación del proceso de escaneo del código QR para la entrada y salida automática.
 - La interfaz de pago en la salida, mostrando el cálculo del total a pagar.

3. Proceso de Gestión de Cambios en los Requisitos

Cualquier cambio o ajuste a los requisitos que surja después de la aprobación inicial debe seguir un proceso formal para evitar desviaciones en el alcance, tiempo y costo del proyecto.

1. **Solicitud Formal de Cambio (RFC):** El cliente o un miembro del equipo debe presentar una solicitud por escrito que describa el cambio deseado, la justificación y los beneficios esperados.
2. **Análisis de Impacto:** El Analista de Sistemas y el Líder de Proyecto evaluarán la solicitud para determinar su impacto en:
 - El cronograma del proyecto (las 20 semanas definidas).
 - Los recursos y costos.
 - Otros requisitos existentes (posibles conflictos o nuevas dependencias).
3. **Aprobación del Cambio:** El análisis de impacto se presentará al cliente y a la dirección del proyecto. El cambio solo se implementará si ambas partes están de acuerdo y aprueban formalmente el ajuste en el alcance, tiempo y costo.
4. **Actualización de la Documentación:** Una vez aprobado un cambio, el Líder de Proyecto se asegurará de que el Documento de Requisitos y toda la documentación relevante sean actualizados para reflejar el nuevo ajuste.

Plan de Validación y Herramientas de Análisis de Requisitos

5. Plan de Validación de Requisitos

La validación de los requisitos es un paso crítico para asegurar que el producto final cumpla con las expectativas del cliente. Este proceso se basará en los **Criterios de Aceptación** definidos para cada requisito en el "Documento de Requisitos". Estos criterios servirán como la base para el plan de pruebas de aceptación del sistema (UAT) que realizará el rol de "Probador de Software" junto con el cliente.

A continuación, se muestra un ejemplo de cómo se validará un requisito funcional clave:

- **Requisito (ID: RF-02):** El sistema debe registrar la salida de un vehículo, calcular el monto total a pagar y procesar el pago.
- **Plan de Validación y Pruebas de Aceptación:**
 - **Para el Criterio "El sistema recupera los datos de entrada usando el ticket digital o la matrícula":**
 - **Prueba:** El probador escaneará un código QR de un vehículo activo en el sistema. Se validará que la interfaz muestre la matrícula y la hora de entrada correctas asociadas a ese código.
 - **Para el Criterio "El cálculo del costo total basado en el tiempo de estancia y la tarifa aplicable es correcto":**
 - **Prueba 1:** Se simulará la salida de un vehículo con una estancia de 2 horas y 30 minutos. Se calculará manualmente el costo esperado según la tarifa por minuto/hora configurada y se comparará con el total arrojado por el sistema.
 - **Prueba 2:** Se simulará la salida de un vehículo con una estancia de 26 horas. Se validará que el sistema aplique correctamente el "límite de costo máximo por día".
 - **Para el Criterio "Cualquier descuento de cliente abonado se aplica correctamente":**
 - **Prueba:** Se procesará la salida de un vehículo perteneciente a un "cliente abonado" que tenga un descuento del 20%. Se validará que el sistema aplique dicho descuento sobre el total calculado antes de mostrar el monto final a pagar.
 - **Para el Criterio "Una vez realizado el pago, el sistema registra la transacción como 'completada' y libera el espacio de estacionamiento":**
 - **Prueba:** Tras simular el pago, el probador revisará el panel de control del "Administrador" para confirmar que el espacio de estacionamiento ahora figura como "Disponible" en el estado de ocupación en tiempo real.

6. Herramientas y Técnicas para el Análisis de Requisitos

Para asegurar una comprensión completa y precisa de las necesidades del cliente, el "Analista de Sistemas" empleará las siguientes herramientas y técnicas.

- **Diagrama de Casos de Uso:** Se diseñará un diagrama de casos de uso para representar visualmente las interacciones entre los diferentes actores y el sistema.
 - **Actores Identificados:** Cliente , Administrador , Supervisor , Super Usuario.
 - **Casos de Uso Principales a Diagramar:**
 - **Actor Administrador:** "Gestionar Clientes y Abonos" , "Establecer Tarifas Dinámicas" , "Gestionar Inventario de Espacios" , "Generar Reportes Financieros".
 - **Actor Cliente:** "Acceder a Historial de Ingresos/Egresos" , "Consultar y Descargar Facturación".
 - **Actor Supervisor:** "Generar Reportes Consolidados" , "Auditar Registros de Administradores".
 - **Actor Super Usuario:** "Gestionar Cuentas de Administrador" , "Realizar Mantenimiento de Base de Datos".
- **Entrevistas con el Cliente:** El "Analista de Sistemas" realizará entrevistas detalladas con la "administración del estacionamiento" para profundizar en los requisitos. Se utilizarán preguntas como:
 - Para el requisito de Penalizaciones: "¿El recargo del 50% para vehículos que exceden las 72 horas es fijo, o debería ser configurable? ¿Se requiere enviar una notificación al administrador antes de que se aplique?"
 - Para el requisito de Tarifas: "¿Existen escenarios de tarifas especiales no contemplados, como para eventos o días festivos? ¿Cuál es el procedimiento deseado si un cliente pierde su ticket digital?"

- Para el requisito de Reportes: "¿El reporte de rotación de vehículos debe poder filtrarse por días de la semana (ej. lunes vs. sábado) para analizar patrones de uso? ".
- **Análisis de Sistemas Similares:** El equipo de diseño y análisis investigará otros sistemas de gestión de estacionamientos comerciales para identificar funcionalidades estándar de la industria, comparar interfaces de usuario y evaluar soluciones tecnológicas. Esto ayudará a validar que todos los aspectos importantes han sido cubiertos y a proponer mejoras basadas en las mejores prácticas del mercado.

Matriz de Trazabilidad de Requisitos

Esta matriz conecta los requisitos funcionales que documentamos previamente con los módulos del sistema, los casos de uso que los implementan y los casos de prueba que los validarán, asegurando una cobertura completa desde la definición hasta la validación.

Matriz de Trazabilidad de Requisitos: Sistema de Estacionamiento

| ID de Requisito | Descripción del Requisito | Módulo del Sistema | Casos de Uso Relacionados | Casos de Prueba Relacionados | Estado |
|-----------------|--|--|--------------------------------------|------------------------------|-------------------|
| RF-01 | Registrar la entrada de un vehículo capturando su matrícula y asignando un ticket digital. | Módulo de Renta / Gestión de Espacios ¹¹¹ | CU-01: Registrar Entrada de Vehículo | CT-01.1, CT-01.2, CT-01.3 | Aprobado |
| RF-02 | Registrar la salida de un vehículo, calcular el monto total a pagar y procesar el pago. | Módulo de Renta ² | CU-02: Procesar Salida y Cobro | CT-02.1, CT-02.2, CT-02.3 | En Proceso |

| ID de Requisito | Descripción del Requisito | Módulo del Sistema | Casos de Uso Relacionados | Casos de Prueba Relacionados | Estado |
|------------------------|---|---|---------------------------------------|-------------------------------------|-------------------|
| RF-03 | Permitir la gestión de clientes abonados, asignando un código único y almacenando su historial. | Módulo de Gestión de Usuarios ³ | CU-03: Gestionar Clientes Abonados | CT-03.1, CT-03.2 | Aprobado |
| RF-04 | Aplicar descuentos en la tarifa mensual a los clientes abonados. | Módulo de Renta / Gestión de Usuarios ⁴⁴ | CU-04: Aplicar Descuentos a Abonados | CT-04.1, CT-04.2 | En Proceso |
| RF-05 | Permitir al administrador configurar tarifas dinámicas con un costo máximo por día. | Módulo de Renta ⁵ | CU-05: Configurar Tarifas del Sistema | CT-05.1, CT-05.2 | Aprobado |
| RF-06 | Generar reportes detallados para el administrador (ingresos, ocupación, etc.). | Módulo de Reportes ⁶ | CU-06: Generar Reporte Administrativo | CT-06.1, CT-06.2, CT-06.3 | En Proceso |
| RF-07 | Permitir a clientes abonados acceder a su historial y | Portal Cliente / Gestión de Usuarios ⁷ | CU-07: Consultar Historial y Facturas | CT-07.1, CT-07.2, CT-07.3 | Pendiente |

| ID de Requisito | Descripción del Requisito | Módulo del Sistema | Casos de Uso Relacionados | Casos de Prueba Relacionados | Estado |
|-----------------|---------------------------|--------------------|---------------------------|------------------------------|--------|
| | facturación vía web. | | | | |

Definición de un Caso de Prueba

A continuación, se detalla como se llevaron acabo los detalles de la matriz anterior con un ejemplo:

- **Identificador del Caso de Prueba:** CT-02.2
- **Requisito Relacionado:** RF-02
- **Descripción del Objetivo:** "Verificar que el sistema calcula correctamente el costo total de la estancia basándose en la tarifa por tiempo y aplicando el límite máximo diario cuando corresponda."
- **Precondiciones:**
 - Debe existir un vehículo registrado en el sistema con una hora de entrada conocida.
 - Debe existir al menos una tarifa activa (ej. \$1.50 por minuto, con un máximo de \$200 por día).
 - El usuario que realiza la prueba debe estar autenticado como "Administrador" o un rol con permisos de cobro.
- **Pasos de Ejecución:**
 1. Seleccionar el vehículo de prueba cuya estancia ha superado las 24 horas.
 2. Iniciar el proceso de "Registrar Salida".
 3. Observar el monto total calculado por el sistema.

- **Resultado Esperado:**

- El sistema debe mostrar un monto a pagar igual al "límite de costo máximo por día" configurado en la tarifa (ej. \$200), y no el cálculo total de minutos, que sería mayor.

Matriz de Priorización de Requisitos

Esta matriz asigna valores numéricos a cada requisito según criterios clave para determinar de forma objetiva su prioridad final en el desarrollo.

| ID del Requisito | Descripción | Importancia (1-5) | Urgencia (1-5) | Impacto (1-5) | Esfuerzo (1-5) | Costo (1-5) | Prioridad Final |
|------------------|---|-------------------|----------------|---------------|----------------|-------------|-----------------|
| RF-01 | Registrar la entrada de un vehículo. | 5 | 5 | 5 | 3 | 3 | Muy Alta |
| RF-02 | Registrar la salida y procesar el pago. | 5 | 5 | 5 | 4 | 4 | Muy Alta |
| RF-03 | Gestionar clientes abonados. | 4 | 4 | 4 | 3 | 3 | Alta |
| RF-04 | Aplicar descuentos a los abonados. | 3 | 3 | 3 | 3 | 2 | Media |
| RF-05 | Configurar tarifas dinámicas. | 4 | 5 | 4 | 2 | 2 | Muy Alta |
| RF-06 | Generar reportes para el administrador. | 4 | 3 | 4 | 4 | 3 | Alta |

| | | | | | | | |
|--------------|---|---|---|---|---|---|--------------|
| RF-07 | Permitir acceso a portal web de clientes. | 3 | 2 | 4 | 4 | 4 | Media |
|--------------|---|---|---|---|---|---|--------------|

Explicación de los Criterios

- **Importancia:** Grado de relevancia para los objetivos del proyecto (1 = Muy baja, 5 = Muy alta).
- **Urgencia:** Necesidad de implementar el requisito pronto (1 = No urgente, 5 = Muy urgente).
- **Impacto:** Influencia en el funcionamiento del sistema o en la experiencia del usuario (1 = Bajo impacto, 5 = Alto impacto).
- **Esfuerzo:** Dificultad para implementar el requisito (1 = Poco esfuerzo, 5 = Mucho esfuerzo).
- **Costo:** Recursos necesarios para el desarrollo (1 = Bajo costo, 5 = Alto costo).
- **Prioridad:** Clasificación general de la prioridad, que se determina combinando los criterios anteriores.

En una matriz de priorización, generalmente se asigna una prioridad final calculando una fórmula o aplicando una escala personalizada en función de la suma o ponderación de cada criterio.

FASE 3

1. OBJETIVO DEL DISEÑO TÉCNICO

Esta etapa del ciclo de desarrollo funge como el pilar central que conecta el análisis de necesidades operativas con la ingeniería del software. Su finalidad esencial es traducir los requerimientos del negocio (control de flujos vehiculares, tarificación dinámica, administración de abonados) en una arquitectura de software robusta, modular y preparada para el crecimiento.

Más allá de una simple descripción, este diseño establece los cimientos técnicos para resolver los desafíos críticos de la implementación:

- **Orquestación de Componentes:** Define la sinergia entre el framework **Django**, que gestiona la lógica del servidor, y la interfaz de usuario basada en tecnologías web estándar, operando bajo el paradigma Modelo-Vista-Template para asegurar una respuesta ágil.
- **Integridad de la Información:** Estructura un modelo de datos relacional en **MySQL** diseñado para mantener la coherencia absoluta de la información, garantizando que las relaciones entre vehículos, tarifas y espacios físicos sean inviolables y que el estado de ocupación se refleje con precisión milimétrica.
- **Experiencia de Operación:** Configura flujos de trabajo optimizados que reducen la fricción en la interacción humano-máquina, permitiendo al personal operativo ejecutar tareas clave (como el registro de ingresos o la liberación de cajones con cálculo automático) de manera fluida y eficiente.
- **Resiliencia del Sistema:** Establece protocolos de manejo de excepciones para mitigar riesgos operativos, tales como la asignación errónea de espacios saturados o la gestión de consultas sobre matrículas no registradas, asegurando la continuidad del servicio.

En definitiva, este esquema actúa como el "plano arquitectónico" que orienta la codificación, garantizando que el sistema "Estacionamiento Urbano" no solo sea funcional, sino también seguro, mantenible y escalable.

2. DISEÑO DE ARQUITECTURA DEL SISTEMA

Para cumplir con los requisitos de velocidad y fiabilidad en el cobro por tiempo, se ha seleccionado una arquitectura basada en el patrón nativo de Django.

2.1 Arquitectura Lógica (Cliente-Servidor)

El ecosistema tecnológico opera bajo un esquema **Cliente-Servidor** robusto, orquestado sobre el protocolo, garantizando una comunicación fluida y segura.

- **Capa de Presentación (Frontend):** Actúa como la interfaz visual y punto de interacción con el usuario final. Es responsable de renderizar dinámicamente el "Tablero de Control", visualizando el estado de ocupación de las zonas (Balbuena, Moctezuma, Aeropuerto) mediante indicadores visuales semánticos. Está construida sobre estándares web modernos e implementa lógica condicional en el cliente para reflejar en tiempo real la disponibilidad de espacios (Semáforo Verde/Rojo), optimizando la experiencia de usuario.
- **Núcleo de Procesamiento (Backend):** Constituye el cerebro lógico del sistema. Reside en el servidor y es responsable de ejecutar la lógica de negocio compleja, incluyendo el cálculo preciso de tarifas basado en tiempos de estancia, la aplicación de reglas de negocio para topes diarios y descuentos a abonados, así como la gestión de sesiones y seguridad. Está implementado en **Python**, aprovechando la potencia y seguridad del framework **Django**.
- **Capa de Persistencia de Datos (Base de Datos):** Funciona como el repositorio centralizado y estructurado de la información. Se utiliza un sistema de gestión de bases de datos relacionada **MySQL**, administrado mediante la herramienta **HeidiSQL**, asegurando la integridad, consistencia y disponibilidad de los datos críticos del negocio.

2.2 Flujo de Interacción entre Componentes

El ciclo de vida de una transacción (ej. Registrar Entrada) sigue el siguiente orden estricto:

1. **Solicitud:** El navegador envía los datos del vehículo (Placa y Tipo) vía POST.
2. **URL:** Django recibe la ruta (ej. / o /cobrar/) y la dirige a la Vista correspondiente (inicio o cobrar).

3. **Lógica:** La vista ejecuta el algoritmo: busca el primer espacio disponible (ocupado=False) correspondiente al tipo de vehículo seleccionado.
4. **Consulta:** El ORM de Django traduce la petición a SQL para insertar un nuevo registro en la tabla app_registro y actualizar el estado en app_espacio.
5. **Respuesta:** Se retorna la plantilla HTML actualizada, mostrando el mensaje de éxito y cambiando el color de la tarjeta de la zona a Rojo si se llenó el cupo.

3. ARQUITECTURA DE DATOS Y PERSISTENCIA

El esquema de base de datos ha sido diseñado bajo principios de normalización para asegurar la integridad referencial y optimizar las consultas transaccionales.

3.1 Diccionario de Datos y Entidades

A continuación, se detalla la estructura lógica de las tablas principales del sistema:

| Entidad (Tabla) | Descripción Funcional | Atributos Clave (Campos) |
|---|---|--|
| CATÁLOGO DE TARIFAS (TipoEspacio) | Maestro de configuración financiera. Define las reglas de cobro aplicables según la categoría del vehículo. | <ul style="list-style-type: none">• ID (PK): Identificador único.• Nombre: Categoría (Automóvil, Moto, Camión).• Tarifa Hora: Costo base unitario.• Tope Diario: Límite financiero máximo por día (Regla de negocio). |
| INVENTARIO FÍSICO (Espacio) | Representación digital de la infraestructura del estacionamiento. | <ul style="list-style-type: none">• Identificador: Código operativo (Ej: A-01).• Zona: Segmentación lógica (Aeropuerto, Balbuena, Moctezuma).• Estado: Bandera booleana (ocupado) crítica para la asignación de lugares. |

| | | |
|---|---|--|
| | | <ul style="list-style-type: none"> • Tipo (FK): Vinculación con la tarifa correspondiente. |
| BITÁCORA TRANSACCIONAL (Registro) | Historial inmutable de operaciones. Registra el ciclo de vida de cada estancia vehicular. | <ul style="list-style-type: none"> • Matrícula: Identificador del vehículo cliente. • Timestamp Entrada: Fecha/Hora de inicio del servicio. • Timestamp Salida: Fecha/Hora de fin (Nullable hasta el cierre). • Monto: Valor financiero calculado. • Status Pago: Confirmación de liquidación. |
| PERFIL DE ABONADO (Cliente) | Extensión del modelo de autenticación para gestión de usuarios recurrentes. | <ul style="list-style-type: none"> • User (FK): Relación 1:1 con el sistema de autenticación nativo. • Datos Contacto: Teléfono. • Datos Vehículo: Placas predeterminadas y Tipo (Para automatización de entradas). |

3.2 Diagrama de Relaciones y Cardinalidad

El modelo relacional garantiza la consistencia de los datos mediante las siguientes restricciones de integridad:

- **Relación Tarifaria (Espacio -- Tipo_Espacio):**
 - *Cardinalidad:* Muchos a Uno (N:1).

- *Lógica:* Múltiples cajones de estacionamiento pueden compartir la misma categoría de tarifa, facilitando la actualización masiva de precios.
- **Relación Operativa (Registro -- Espacio):**
 - *Cardinalidad:* Muchos a Uno (N:1).
 - *Lógica:* Un espacio físico puede tener múltiples registros históricos a lo largo del tiempo, pero un registro activo solo puede estar asociado a un espacio en un momento dado.
- **Relación de Identidad (Cliente -- Usuario):**
 - *Cardinalidad:* Uno a Uno (1:1).
 - *Lógica:* Cada cuenta de usuario del sistema posee un único perfil de abonado asociado, lo que permite separar las credenciales de acceso de los datos operativos del negocio (placas, teléfono), facilitando la escalabilidad y seguridad.

4. DISEÑO DE INTERFACES Y EXPERIENCIA DE USUARIO

La interfaz prioriza la velocidad de operación para el cajero y la claridad para el administrador.

4.1 Interfaces de Operación

- **Tarjetas de Estado en Tiempo Real:** Diseño visual que representa las zonas del estacionamiento. Utiliza lógica condicional en el template:
 - **Verde:** Si espacio esta disponible
 - **Rojo:** Si no hay espacios disponibles.
- **Formulario de Entrada Rápida:** Lista desplegable dinámica que se alimenta de la tabla Tipo_Espacio, evitando errores de captura manual en las tarifas.

4.2 Interfaz de Salida y Cobro (Caja)

- **Buscador Inteligente:** Campo único para ingresar matrícula. El backend busca registros abiertos
- **Ticket Digital:** Al encontrar el vehículo, se genera una vista previa con:

- Cálculo de horas
- Desglose de costos (Subtotal, Descuentos por ser Cliente, Penalizaciones >72h).
- Total a Pagar.

4.3 Interfaces de Administración

- **Django Admin:** Panel nativo para el alta, baja y modificación de Precios y Creación de nuevos Espacios físicos sin tocar código.
- **Reportes:** Botón de exportación directa a **Excel (CSV)** que descarga el historial de ingresos financieros filtrando solo las transacciones pagadas (pagado=True).

DIAGRAMA DE SECUENCIA DEL SISTEMA

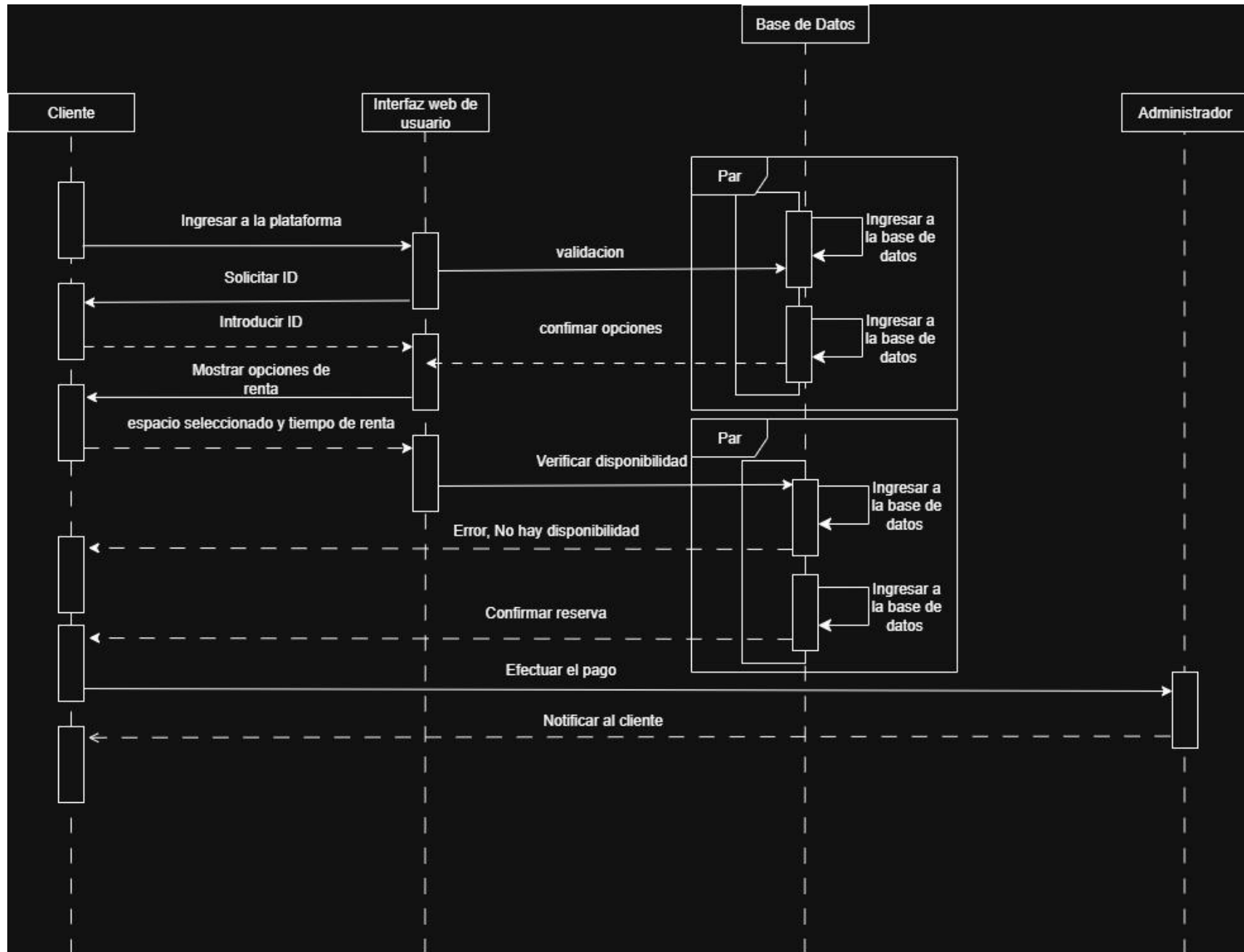


DIAGRAMA DE CASOS DE USO

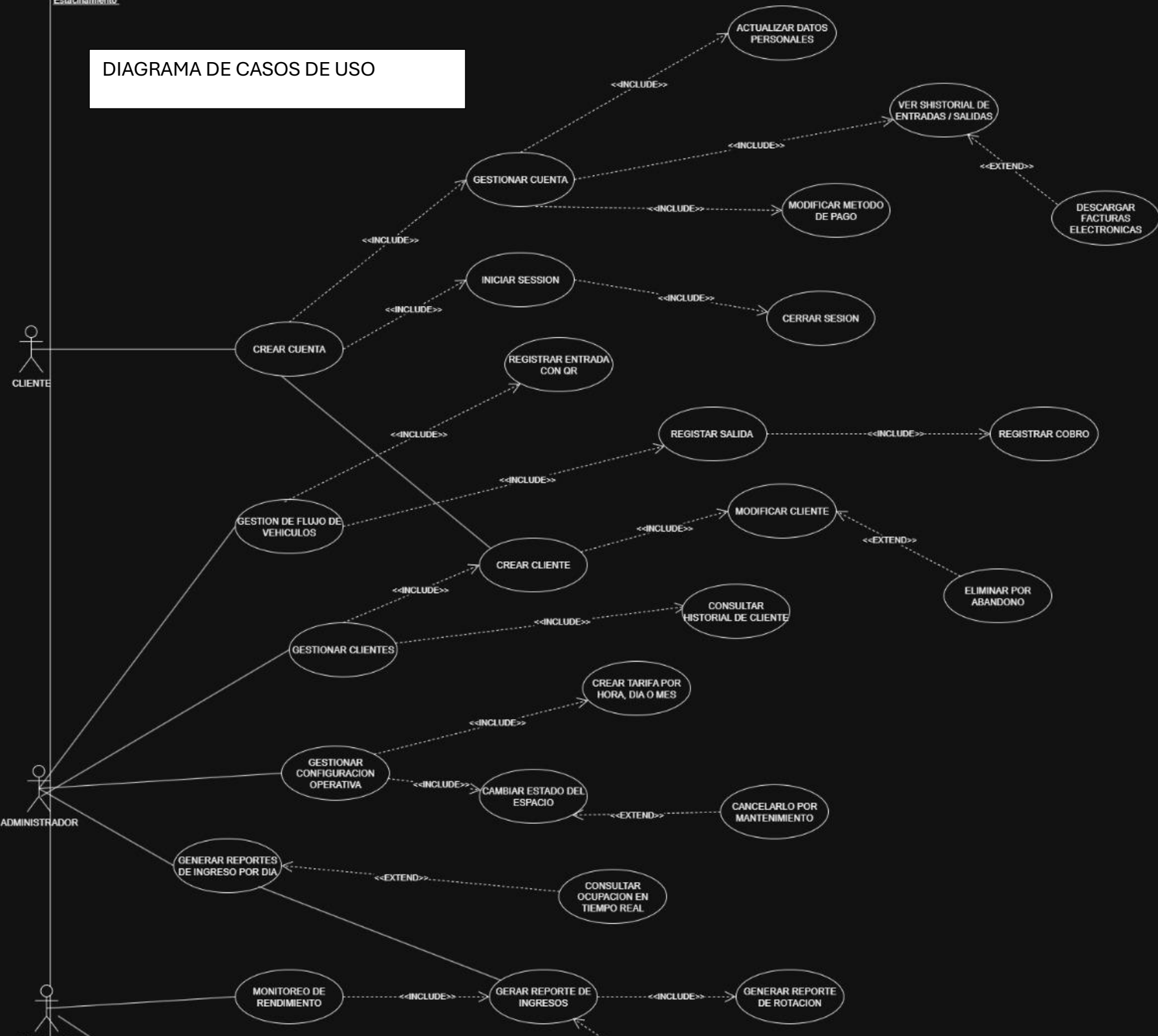


DIAGRAMA DE CLASES

