

I.E.S TORRE DEL REY

TRABAJO FIN DE GRADO

SISTEMA DE GESTIÓN DE PARKING DE UN HOTEL



NOMBRE: Juan Francisco

APELLIDOS: Márquez Arias

CURSO: 2º Dam (Desarrollo de aplicaciones multiplataforma).

2024-2025

Contenido

[1. Resumen / Abstract 3](#_Toc198288606)

[1.1. Objetivo general y logros obtenidos. 3](#_Toc198288608)

[2. Introducción 4](#_Toc198288609)

[3. Análisis del sistema 4](#_Toc198288611)

[3.1. Análisis de requisitos 4](#_Toc198288612)

[ Funcionales: 5](#_Toc198288613)

[ No funcionales 5](#_Toc198288614)

[3.2. Casos de uso 6](#_Toc198288615)

[3.3 Usuarios del sistema 6](#_Toc198288616)

[3.4. Historias de usuario (opcional pero recomendable) 6](#_Toc198288617)

[6. Diseño 6](#_Toc198288618)

[7. Desarrollo 6](#_Toc198288619)

[8. Pruebas 7](#_Toc198288620)

[9. Manual de usuario 7](#_Toc198288621)

[10. Conclusiones 7](#_Toc198288622)

[11. Bibliografía y referencias 7](#_Toc198288623)

[12. Anexos (opcional) 7](#_Toc198288624)

# Resumen / Abstract

Este proyecto consiste en el desarrollo de una aplicación de escritorio, creada con Java y JavaFX, para gestionar el aparcamiento de coches en un hotel. El sistema permite al personal del hotel llevar un control eficiente de las plazas de aparcamiento disponibles, registrar entradas y salidas de vehículos, realizar reservas, y visualizar el estado del parking en tiempo real. La interfaz ha sido diseñada para ser intuitiva y funcional, adaptada al uso diario por parte de los recepcionistas y personal administrativo. Además, se ha implementado una base de datos para almacenar la información de forma segura y persistente. Este proyecto demuestra la aplicación de buenas prácticas en programación orientada a objetos, diseño de interfaces gráficas, y gestión de datos, ofreciendo una solución tecnológica realista y escalable para mejorar el servicio al cliente en el sector hotelero.

# Objetivo general y logros obtenidos.

Desarrollar una aplicación de gestión de aparcamiento para un hotel, utilizando Java y JavaFX, que permita controlar de forma eficiente y visual la ocupación de las plazas, registrar entradas y salidas de vehículos, y facilitar al personal del hotel una herramienta intuitiva y funcional para la administración del parking.

Se ha utilizado para este proyecto JAVAFX, SCENE BUILDER, HEIDISQL para hacer una interfaz amigable al usuario y hacer uso de los datos necesarios.

En cuanto a los logros obtenidos, he conseguido un diseño e implementación de una interfaz gráfica amigable con JavaFX para aquellas personas del hotel, y así hacerle más fácil el trabajo. Además, he seguido en el proyecto una estructuración,siguiendo el patrón Modelo-Vista-Controlador (MVC). Incluso la creación de una base de datos funcional para almacenar información de plazas, vehículos y registros.

Se han implementado funcionalidades clave como:

* Registro de entrada y salida de vehículos.
* Consulta en tiempo real del estado del aparcamiento.
* Gestión de reservas.

# Introducción

Consideré hacer un aparcamiento de hotel porque a raíz de buscar ideas me pareció interesante llevar el control de entrada y salida de coches y hablando con un amigo que tiene un hotel sin parking vi esa oportunidad de negocio para ofrecerle y que diera el paso a construir un parking para su hotel.

También la digitalización en el sector hotelero se ha convertido en un factor clave para mejorar la eficiencia operativa, la calidad del servicio y la experiencia del cliente. La incorporación de tecnologías en los procesos internos permite automatizar tareas rutinarias, reducir errores humanos y optimizar la gestión de recursos como el aparcamiento, el check-in, las reservas o el mantenimiento. En un entorno cada vez más competitivo, contar con herramientas digitales no solo agiliza el trabajo del personal, sino que también transmite una imagen moderna y profesional del hotel. La gestión digital del aparcamiento, en particular, ayuda a maximizar el uso de las plazas disponibles, reducir los tiempos de espera y ofrecer un servicio más cómodo y personalizado al cliente desde su llegada.

# 3. Análisis del sistema

## 3.1. Análisis de requisitos

### Funcionales:

* El sistema permite realizar el registro de un cliente en el hotel, almacenando sus datos personales en la base de datos.
* El sistema permite realizar una búsqueda de los vehículos asociados a un cliente, facilitando la gestión individualizada de sus coches.
* El sistema permite registrar un vehículo vinculado a un cliente específico, garantizando la trazabilidad entre personas y coches.
* El sistema permite registrar un nuevo usuario con un rol específico (por ejemplo, recepcionista o administrador). Esta funcionalidad estará restringida únicamente al administrador.
* El sistema permite cambiar el rol del usuario de forma ágil y eficaz, actualizando automáticamente los permisos asignados.
* El sistema permite emitir y registrar facturas asociadas a los servicios utilizados por un cliente.
* El sistema permite registrar una reserva de aparcamiento y marcar automáticamente la plaza como ocupada en tiempo real.
* El sistema permite registrar un pago realizado por el cliente y, tras confirmarlo, enviar un correo electrónico de notificación.
* El sistema permite al usuario cerrar sesión o salir del programa de manera segura desde la interfaz.

### No funcionales

* + - El sistema tiene una interfaz gráfica amigable y fácil de usar.
    - La aplicación está desarrollada en Java con JavaFX.
    - El sistema responde en menos de 5 segundos ante acciones del usuario.
    - La base de datos garantiza la integridad y seguridad de la información.
    - El sistema debe poder instalarse en cualquier equipo con Java instalado.

## 3.2. Casos de uso

Diagrama UML + explicación

## 3.3 Usuarios del sistema

### 1. ****Administrador****

* Tiene control total sobre el sistema.
* Puede registrar nuevos usuarios y asignar roles.
* Supervisa y modifica reservas, facturas y pagos.

### 2. ****Recepcionista****

* Registra la entrada y salida de vehículos.
* Realiza reservas y gestiona las existentes.
* Consulta el estado de las plazas en tiempo real.
* Registra pagos y puede emitir facturas.
* Accede a la información de clientes y sus vehículos.

### 3. ****Cliente****

* Puede hacer reservas de plazas.
* Recibe notificaciones sobre pagos con su factura incluida.

# 6. Diseño

* **Arquitectura del sistema**
  + MVC (Modelo-Vista-Controlador) con JavaFX
  + Diagrama de clases UML
* **Modelo de datos**
  + Diagrama E-R o esquema de la base de datos
* **Flujo de navegación**
  + Cómo se interactúa con la interfaz
* **Diseño de la interfaz (UI)**
  + Pantallas principales con capturas
  + Justificación del diseño (usabilidad, colores, etc.)

# 7. Desarrollo

* **Tecnologías utilizadas**
  + Java, JavaFX, SceneBuilder, SQLite/MySQL, Git, etc.
* **Herramientas de desarrollo**
  + IDE (IntelliJ, Eclipse), control de versiones (Git)
* **Explicación del código clave**
  + Gestión de plazas
  + Sistema de reserva
  + Control de acceso
  + Carga de datos en tablas
* **Problemas encontrados y soluciones**

# 8. Pruebas

* **Pruebas funcionales**
  + Qué funcionalidades fueron testeadas y cómo
* **Pruebas de interfaz**
  + Test de usabilidad con usuarios reales (si puedes hacerlo)
* **Errores corregidos**
  + Breve registro de bugs detectados y solucionados

# 9. Manual de usuario

* Cómo instalar y ejecutar el programa
* Manual de uso por roles (ej. recepcionista)
* Capturas de pantallas con explicación paso a paso

# 10. Conclusiones

* Qué has aprendido en el proyecto
* Retos superados
* Posibles mejoras o evolución futura (versión web, integración con otros sistemas…)

# 11. Bibliografía y referencias

* Documentación oficial de JavaFX, SQLite, libros, tutoriales, StackOverflow (bien citados)

# 12. Anexos (opcional)

* Código fuente adicional
* Casos de prueba
* Wireframes o bocetos previos