# Documento de Despliegue - Control de Parking Privado

**Versión 1.0**  
**Fecha:** 01/02/2025  
**Autor:** [Víctor Hong Gragera]

## Historial de Revisiones

| **Fecha** | **Descripción** | **Autor** |
| --- | --- | --- |
| 01/02/2025 | Inicialización | [Víctor Hong Gragera] |

## Índice de Contenidos

1. Introducción
2. Detalle de las mejoras
   1. Mejoras técnicas
   2. Mejoras funcionales
3. Servidor de aplicaciones
   1. Identificador del servidor
   2. Requisitos mínimos
   3. Configuración recomendada
4. Servidor de base de datos
   1. Identificación del servidor
   2. Requisitos mínimos
   3. Scripts
5. Ficheros o binarios a desplegar
   1. Ubicación
6. Operativa
   1. Operaciones de base de datos
   2. Operaciones de despliegue
   3. Condiciones y verificación de éxito
7. Control de versiones
8. Observaciones

## 1. Introducción

Este documento describe el despliegue inicial de la aplicación Control de Parking Privado, una herramienta diseñada para gestionar el acceso y control de vehículos en un parking privado de camiones. La versión de lanzamiento incluye autenticación segura, control de roles y persistencia de datos con SQLite. Además, permite exportar datos del historial a PDF y realizar un seguimiento detallado de entradas y salidas de vehículos.

## 2. Detalle de las mejoras

### 2.1. Mejoras técnicas

* Uso de Spring Boot para un backend escalable y seguro.
* Autenticación y autorización con Spring Security.
* Encriptación de contraseñas con BCrypt.
* Persistencia de datos mediante SQLite.
* Generación de PDF para reportes.

### 2.2. Mejoras funcionales

* Control automatizado de fecha y hora en registros de vehículos.
* Exportación de datos del historial para auditoría y análisis.
* Sistema intuitivo de gestión basado en roles (Administrador y Empleado).
* Implementación de un instalador para facilitar el despliegue.

## 3. Servidor de aplicaciones

### 3.1. Identificador del servidor

* **Tipo:** Local
* **URL:** [http://localhost:8080](http://localhost:8080" \t "_new)

### 3.2. Requisitos mínimos

* **JDK:** Versión 17 o superior.
* **Maven:** Para la gestión de dependencias y compilación.
* **Navegador web:** Compatible con Chrome, Firefox o Edge.

### 3.3. Configuración recomendada

* **Memoria mínima:** 512 MB.
* **Tiempo de espera (timeout):** 30 segundos.

## 4. Servidor de base de datos

### 4.1. Identificación del servidor

* **Tipo:** Base de datos embebida SQLite.
* **Ubicación:** Incluida en el directorio principal del proyecto.

### 4.2. Requisitos mínimos

* **SQLite:** 3.0 o superior.

### 4.3. Scripts

* La base de datos se inicializa automáticamente al arrancar la aplicación.

## 5. Ficheros o binarios a desplegar

### 5.1. Ubicación

* **Archivo ejecutable:** control-parking.exe para sistemas Windows.
* **Archivo JAR:** parking-control-app.jar en el directorio de distribución.

## 6. Operativa

### 6.1. Operaciones de base de datos

* La base de datos SQLite se configura automáticamente al ejecutar la aplicación.
* No se requiere configuración adicional por parte del administrador.

### 6.2. Operaciones de despliegue

1. Descargue el instalador desde el repositorio oficial.
2. Ejecute el archivo control-parking.exe para instalar la aplicación.
3. Acceda al sistema a través de [http://localhost:8080](http://localhost:8080" \t "_new).

### 6.3. Condiciones y verificación de éxito

* Acceso exitoso a la aplicación mediante un navegador web.
* Validación de login con usuarios predefinidos.
* Verificación de operaciones CRUD en usuarios, propietarios y vehículos.
* Generación de PDF del historial.

## 7. Control de versiones

* **Repositorio GitHub:** [control\_vehiculos](https://github.com/Honcito/control_vehiculos.git" \t "_new)

## 8. Observaciones

* Para realizar pruebas adicionales, puede encontrar una colección de endpoints en Postman en el siguiente enlace: Postman Collection.
* Si encuentra errores o tiene sugerencias, por favor, cree un ticket en el repositorio oficial.