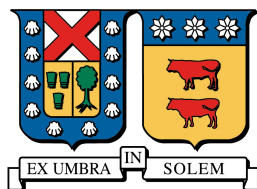


UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE ELECTRÓNICA
CHILE



**SISTEMA INTELIGENTE PARA EL CONTROL DE ACCESOS
VEHICULARES Y LA GESTIÓN DE ESTACIONAMIENTOS EN
COMUNIDADES RESIDENCIALES DE ACCESO CERRADO:
ENFOQUE EN VISIÓN POR COMPUTADOR**

BENJAMÍN CRISTIAN MENA ARDURA

**MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL TELEMÁTICO**

**PROFESOR GUÍA
NICOLÁS SEBASTIÁN GÁLVEZ RAMÍREZ**

**PROFESORES CORREFERENTES
NICOLÁS TORRES RUDLOFF
NICOLÁS VALENZUELA LEVI**

Octubre 2026



CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☒ Memoria o trabajo de título ☐ Tesis de Postgrado
Sistema Inteligente para el Control de Accesos Vehiculares y la Gestión de Estacionamientos en
Título del trabajo: Comunidades Residenciales de Acceso Cerrado: Enfoque en Visión por Computador

Nombre del candidato(a): Benjamín Cristian Mena Ardura

Carrera / Grado: Ingeniería Civil Telemática

Campus: San Joaquín

Departamento: Departamento de Electrónica

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Nicolás Gálvez Ramírez, en mi calidad de profesor(a) guía/director(a)
del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución.

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL (marcar una opción)

☒ El trabajo **NO contiene** información que amerite confidencialidad y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (**embargo**) por (**marcar una opción**):

☐ 6 meses ☐ 12 meses ☐ 2 años ☐ 3 años ☐ 5 años ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 19/10/2026

Firma:

Estudiante o Candidato(a):

Fecha: 07/10/2026

Firma:

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

Agradecimientos

El cierre de este proyecto no habría sido posible sin el apoyo, el cariño y la compañía de las personas que han estado presentes a lo largo de este camino. A todos quienes, de una u otra forma, contribuyeron a que pudiera llegar hasta aquí, muchas gracias.

En primer lugar, quiero agradecer a mi mamá, por el enorme sacrificio y esfuerzo que ha realizado para sacarnos adelante a mi hermana Antonia y a mí. Gracias a todo lo que nos entregaste, tuvimos la oportunidad de estudiar, cumplir nuestras metas y convertirnos en profesionales. A la Marta, quien me cuidó con un cariño inmenso desde que tenía quince días de nacido hasta mis doce años y que, hasta el día de hoy, continúa pendiente y preocupada por mí, gracias por ser una presencia constante y amorosa en mi vida.

A mi pareja, Catalina, gracias por acompañarme durante una parte fundamental de mi carrera. Gracias por tu paciencia, por tu apoyo emocional incondicional y por estar presente en los momentos más difíciles. También agradezco profundamente tu ayuda para comprender y aterrizar los aspectos sociales y arquitectónicos relacionados con las comunidades residenciales, aporte que fue fundamental para darle forma a este proyecto.

A mis amigos de la selección de vóley, gracias por creer en mí y por seguir presentes a pesar de mis reiteradas ausencias en entrenamientos, partidos y salidas. A mis amigos de carrera, gracias por hacer este camino mucho más ameno y por convertirse en una red de apoyo fundamental frente a los desafíos académicos y personales que surgieron durante estos años.

En mi formación escolar, quiero hacer una mención muy especial a Sandra Marín, mi profesora de física y profesora jefe. Ella tuvo un papel fundamental en el camino que finalmente me llevó a convertirme en ingeniero. Aunque en un comienzo mi sueño era estudiar astronomía, supo guiarme y despertar en mí una vocación que no había descubierto. Gracias por cultivar mi amor por la física, por confiar siempre en mis capacidades y por tener una fe inquebrantable en que algún día me convertiría en un gran profesional.

Durante mi etapa universitaria, quiero agradecer especialmente al profesor Nicolás Gálvez, por sus enseñanzas y por estar siempre dispuesto a ayudar cuando lo necesité. Al profesor Nicolás Torres, por haber sido una influencia fundamental en mi especialización en Inteligencia Artificial y por orientarme hacia el perfil profesional que quería construir. Finalmente, a la profesora Patricia Franco, por creer en nuestra idea, confiar en nuestro trabajo y brindarnos el apoyo necesario para hacer posible este proyecto. Sin su respaldo, mis compañeros y yo no habríamos podido llevar adelante esta iniciativa. A todos ustedes, infinitas gracias.

A mi tío Ricardo.

*Por su cariño, su recuerdo imborrable y por acompañarme
en espíritu hasta el final de mi carrera.*

Resumen

La inseguridad ciudadana constituye uno de los principales desafíos sociales y urbanos de América Latina, región que presenta una elevada concentración de homicidios y una creciente percepción de inseguridad. Esta problemática afecta la calidad de vida y modifica las dinámicas de interacción con el entorno, manifestándose de manera diferenciada según las características de cada ciudad. En Chile, si bien los niveles de homicidios son inferiores a los observados en otros países de la región, la inseguridad mantiene una alta relevancia, particularmente en la Región Metropolitana, donde delitos como robos violentos, portonazos y encerronas evidencian vulnerabilidades en los accesos a los recintos residenciales.

En las comunidades residenciales de acceso cerrado, el control de accesos vehiculares y la gestión de estacionamientos de visita continúan dependiendo principalmente de procesos manuales, caracterizados por la intervención humana, la limitada trazabilidad y la ausencia de información en tiempo real. Frente a estas problemáticas, se diseña e implementa *ResiPark*, un sistema inteligente que integra Inteligencia Artificial, Visión por Computador e Internet de las Cosas para automatizar el control de accesos y optimizar la gestión de estacionamientos.

La solución se estructura en cuatro módulos: gestión de accesos vehiculares, gestión de estacionamientos, interfaces de usuario y servicios centrales. Para el reconocimiento vehicular, se implementan modelos YOLOv8s y YOLOv11s para la detección de vehículos y patentes, respectivamente, complementados con Florence-2 para la transcripción OCR. El monitoreo de estacionamientos se desarrolla mediante ESP32, sensores ultrasónicos y comunicación LoRa, mientras que la interacción con los usuarios se realiza mediante una aplicación móvil y una plataforma web. Finalmente, el sistema es desplegado y evaluado en un pasaje cerrado de la comuna de Cerrillos, alcanzando un mAP50 de 0,995 en la detección de vehículos y 0,977 en la detección de patentes, validándose el funcionamiento integrado de los principales componentes.

Palabras clave: inseguridad ciudadana, control de acceso vehicular, reconocimiento automático de patentes, visión por computador, Internet de las Cosas, comunidades residenciales de acceso cerrado, estacionamientos inteligentes, Inteligencia Artificial.

Abstract

Citizen insecurity is one of the main social and urban challenges in Latin America, a region characterised by high homicide rates and a growing perception of insecurity. This situation affects quality of life and changes how people interact with their surroundings, with manifestations varying by city. In Chile, although homicide rates are lower than those observed in other countries in the region, insecurity remains a major concern, particularly in the Metropolitan Region, where crimes such as violent vehicle theft, home-gate robberies, and vehicle ambushes expose vulnerabilities at residential access points.

In gated residential communities, vehicular access control and visitor parking management still rely largely on manual processes, resulting in strong dependence on human intervention, limited traceability, and a lack of real-time information. To address these challenges, *ResiPark* is an intelligent system that integrates Artificial Intelligence, Computer Vision, and the Internet of Things to automate vehicular access control and improve visitor parking management.

The proposed solution is organised into four modules: vehicular access management, parking management, user interfaces, and central services. For vehicle identification, YOLOv8s and YOLOv11s models perform vehicle and license plate detection, respectively, and Florence-2 performs optical character recognition. Visitor parking occupancy is monitored using ESP32 microcontrollers, ultrasonic sensors, and LoRa communication, while users interact with the system through a mobile application and a web platform. Finally, the system is deployed and evaluated in a gated residential lane in the commune of Cerrillos, achieving an mAP@50 of 0.995 for vehicle detection and 0.977 for license plate detection, while validating the integrated operation of the main system components.

Keywords: citizen insecurity, vehicular access control, automatic license plate recognition, computer vision, Internet of Things, gated residential communities, smart parking, Artificial Intelligence.

Glosario

Para la correcta comprensión de la presente memoria, se definen los siguientes términos técnicos utilizados en el desarrollo del sistema **ResiPark**:

ALPR (Automatic License Plate Recognition): Tecnología de vigilancia y gestión que utiliza el reconocimiento óptico de caracteres e inteligencia artificial para leer automáticamente las placas de matrícula de los vehículos. Este sistema procesa imágenes capturadas por cámaras en tiempo real para identificar caracteres alfanuméricos, permitiendo la automatización en el control de accesos, peajes y la gestión de tráfico urbano.

API REST (Representational State Transfer): Arquitectura de software que impone condiciones sobre cómo debe funcionar una API. En este proyecto, se utiliza para la comunicación entre el cliente web/móvil y el servidor backend, gestionando recursos como usuarios, vehículos y reservas mediante métodos HTTP (GET, POST, PUT, DELETE).

Backend: Capa del sistema encargada de procesar la lógica de negocio, gestionar la base de datos y autenticar usuarios. En este proyecto, fue desarrollado utilizando el framework FastAPI en Python.

CNN (Convolutional Neural Network): Tipo de red neuronal profunda diseñada específicamente para procesar datos con una estructura de cuadrícula, como las imágenes. Su arquitectura utiliza capas de convolución que permiten extraer automáticamente características jerárquicas, desde bordes simples hasta formas complejas, siendo la arquitectura fundamental en tareas de clasificación y detección de objetos.

Cloudflare: Plataforma global que proporciona servicios de seguridad, rendimiento y confiabilidad para activos digitales. Actúa como una red de entrega de contenido (CDN) y un escudo de seguridad perimetral. En este proyecto, se utiliza para exponer públicamente el servicio de detección alojado en Google Colab mediante un túnel, sin requerir una dirección IP pública fija.

Dataset: Conjunto de datos estructurados utilizados para entrenar modelos de Inteligencia Artificial. Para este trabajo, se construyeron dos datasets propios obtenidos de Roboflow: uno de 9.689 imágenes de vehículos y otro de 10.125 imágenes de patentes, utilizados para el entrenamiento de los modelos YOLOv8s y YOLO11s respectivamente.

Endpoint: Punto final de comunicación en una API. Corresponde a una URL específica donde el servicio puede ser accedido por una aplicación cliente para realizar una operación (por ejemplo, `/auth/login` o `/verificar-patente/{patente}/{tipo}`).

ESP32: Microcontrolador de bajo costo con conectividad WiFi y soporte para módulos de radio de largo alcance. En este proyecto, se utiliza tanto en el nodo emisor –conectado a un sensor ultrasónico para detectar la ocupación de estacionamientos– como en el nodo receptor, encargado de reenviar dicha información al backend mediante WiFi.

Florence-2: Modelo de visión por computador desarrollado por Microsoft, capaz de resolver múltiples tareas mediante *prompts* de tarea específicos. En este sistema, se utiliza mediante la tarea <OCR> para la transcripción óptica de caracteres de las patentes detectadas, en reemplazo de PaddleOCR (véase entrada correspondiente).

Google Colab: Entorno de desarrollo basado en la nube que permite la redacción y ejecución de código en lenguaje Python a través del navegador. Facilita el acceso gratuito a recursos de cómputo de alto rendimiento, como unidades de procesamiento gráfico (GPU), sin requerir configuración local. En este proyecto, aloja el servicio de detección y reconocimiento (YOLO y Florence-2).

IoT (Internet of Things): Interconexión digital de objetos cotidianos con internet, permitiendo que estos recolecten, envíen y reciban datos de manera autónoma. Esta infraestructura facilita la monitorización y el control remoto de dispositivos físicos, integrándolos en sistemas inteligentes para optimizar procesos y toma de decisiones.

JWT (JSON Web Token): Estándar abierto (RFC 7519) que define un método compacto y autónomo para transmitir información de forma segura entre partes como un objeto JSON. Se utiliza en el sistema para gestionar la autenticación y mantener la sesión de los usuarios de forma segura y sin estado (*stateless*).

LoRa (Long Range): Tecnología de modulación de radio para redes de baja potencia y largo alcance (LPWAN). Se seleccionó para la comunicación de los sensores de estacionamiento debido a su capacidad para atravesar estructuras de concreto en subterráneos donde la señal Wi-Fi es deficiente.

mAP (mean Average Precision): Métrica estándar utilizada para evaluar el rendimiento de los modelos de detección de objetos. Se calcula promediando la precisión media de todas las categorías de objetos que el modelo es capaz de identificar, proporcionando una medida única que equilibra la capacidad de encontrar todos los objetos relevantes (exhaustividad) con la exactitud de las predicciones realizadas.

OCR (Optical Character Recognition): Tecnología que permite convertir diferentes tipos de documentos, como imágenes capturadas por una cámara digital, en datos editables y buscables. En el sistema, se utiliza el modelo **Florence-2** para extraer el texto alfanumérico de las patentes

vehiculares.

PaddleOCR: Biblioteca de código abierto para reconocimiento óptico de caracteres, desarrollada sobre el framework PaddlePaddle. Fue la tecnología contemplada inicialmente para la transcripción de patentes en este proyecto; sin embargo, su implementación presentó severos conflictos de dependencias y problemas de compatibilidad de versiones que bloquearon el progreso del desarrollo, motivando su reemplazo por el modelo Florence-2.

ORM (Object-Relational Mapping): Técnica de programación para convertir datos entre sistemas de tipos incompatibles en lenguajes de programación orientados a objetos. Se utilizó **SQLAlchemy** para interactuar con la base de datos SQL desde código Python.

Raspberry Pi: Serie de computadoras monoplaca de bajo costo y dimensiones reducidas desarrollada con el fin de promover la informática y la ingeniería. Debido a su versatilidad, puertos de conexión físicos (GPIO) y capacidad para ejecutar sistemas operativos completos, es una herramienta fundamental en el desarrollo de prototipos electrónicos y sistemas embebidos. En este proyecto, ejecuta el orquestador de borde y controla el relé de apertura del portón.

ROI (Region of Interest): Región delimitada dentro de una imagen o fotograma sobre la cual se restringe el procesamiento o el análisis. En este sistema, se utiliza para acotar la zona del fotograma donde habitualmente circulan los vehículos frente a la cámara, descartando detecciones fuera de dicha área.

Shadow IT: Uso de sistemas, dispositivos, software, aplicaciones y servicios de TI sin la aprobación explícita del departamento de TI. Aunque el sistema busca mitigar riesgos físicos, su diseño seguro busca evitar vulnerabilidades digitales asociadas a dispositivos IoT mal gestionados.

Supabase: Plataforma de backend como servicio (*BaaS*) que provee una base de datos PostgreSQL gestionada, entre otras funcionalidades. En este proyecto, se utiliza como base de datos SQL centralizada del sistema, almacenando usuarios, vehículos, reservas y el registro de eventos de acceso.

TLS (Transport Layer Security): Protocolo criptográfico que proporciona comunicaciones seguras sobre una red. En este sistema, se emplea en la conexión WiFi del nodo receptor de sensores y en las solicitudes HTTPS hacia el backend, garantizando la confidencialidad de la información transmitida.

Uvicorn: Implementación de servidor de interfaz de puerta de enlace de servidor asíncrono (ASGI) diseñada para Python. Se caracteriza por su alta eficiencia y rapidez en la ejecución de aplicaciones web modernas, permitiendo el manejo de múltiples conexiones simultáneas mediante programación

asíncrona, lo que optimiza el rendimiento en entornos de producción.

Visión por Computador: Subcampo de la inteligencia artificial que se ocupa de cómo las computadoras pueden obtener una comprensión de alto nivel a partir de imágenes o videos digitales. Su objetivo es automatizar tareas que el sistema visual humano puede realizar, tales como el reconocimiento de patrones, la detección de objetos y la reconstrucción de escenas en tres dimensiones.

YOLO (You Only Look Once): Familia de modelos de detección de objetos en tiempo real basados en redes neuronales convolucionales (CNN). En este sistema, se utilizan dos variantes diferenciadas por subtarea: YOLOv8s para la detección vehicular y YOLOv11s para la localización de patentes, ambas seleccionadas mediante una Matriz de Decisión Ponderada.

Índice general

Índice de figuras	13
Índice de tablas	17
1. Introducción	18
1.1. Objetivos	20
1.1.1. Objetivo General	20
1.1.2. Objetivos Específicos	20
1.2. Propósito	21
1.3. Contribuciones	21
1.4. Estructura del Informe	22
2. Problemática	23
2.1. Inseguridad en Latinoamérica	23
2.2. Problemas derivados de la Inseguridad en Santiago de Chile	29
2.2.1. Espacios de Acceso Público	30
2.2.2. Espacios Residenciales	31
2.2.2.1. Control de Acceso en Comunidades Residenciales	33
2.3. Problemática	33
3. Estado del Arte	35
3.1. Sistemas de Monitoreo y Acceso Tradicionales en Comunidades Residenciales de Acceso Cerrado	36
3.2. Vulnerabilidades de Seguridad del Modelo Tradicional	37
3.3. De Edificios Residenciales a Comunidades Inteligentes	39
3.3.1. Internet de las Cosas (IoT) e Infraestructura Sensorial	39
3.3.2. Analítica de Datos y Visión por Computador	41

3.4. Control de Acceso y Seguridad en Comunidades Inteligentes	47
3.4.1. Reconocimiento Automático de Patentes ALPR	48
3.4.1.1. Enfoques Tradicionales	48
3.4.1.2. Enfoques Basados en Aprendizaje Profundo	49
3.4.2. Gestión de Estacionamientos: Visión por Computador vs. Sensores IoT	51
3.4.2.1. Sistemas Basados en Visión por Computador	51
3.4.2.2. Sistemas Basados en Sensores IoT	52
3.5. Soluciones Comerciales de Gestión Residencial	54
3.5.1. Plataformas de gestión administrativa	55
3.5.2. Soluciones de control de acceso físico	56
3.5.3. Referentes internacionales	58
3.5.4. Síntesis comparativa	58
4. Metodología y Desarrollo del Sistema	60
4.1. Requisitos del Sistema	60
4.1.1. Requisitos Funcionales	61
4.1.2. Requisitos No Funcionales	62
4.1.3. Requisitos de Interfaces	63
4.1.3.1. Interfaces de Usuario	63
4.1.3.2. Interfaces Técnicas y de Comunicación	64
4.1.3.3. Interfaces con Servicios Externos	64
4.1.4. Requisitos de Ambiente	65
4.1.4.1. Hardware	65
4.1.4.2. Software	66
4.1.5. Perfiles de Usuario	67
4.2. Arquitectura del Sistema	68
4.3. Módulos de Desarrollo	70
4.4. Diagrama de Contexto	72
4.4.1. Matriz de Requisitos Funcionales y Módulos	73
4.5. Diseño de Interfaces	75
4.5.1. Modelo de Navegación	75
4.5.1.1. Flujo No Autenticado	75
4.5.1.2. Flujo Autenticado	76
4.6. Mockup	76
4.6.1. Interfaz Usuario – App Web	77

4.6.2. Interfaz Usuario – App Móvil	79
5. Resultados	81
5.1. Caso de Estudio	81
5.1.1. Evaluación de tecnologías	82
5.1.2. Metodología de Cálculo	82
5.1.3. Evaluación de Opciones por Componente	83
5.1.3.1. Hardware para Procesamiento en el Borde	83
5.1.3.2. Modelos de Detección de Objetos	84
5.1.3.3. Tecnologías de Sensores para Detección Vehicular	85
5.1.3.4. Protocolos de Comunicación Inalámbrica para IoT	86
5.1.3.5. Sistema de Gestión de Base de Datos (SGBD)	86
5.2. Módulos del Sistema	87
5.2.1. Módulo 1 - Gestión de Accesos Vehiculares (M1):	87
5.2.1.1. Control de Acceso	87
5.2.1.2. Servicio de Detección y Reconocimiento	89
5.2.2. Módulo 2 - Gestión de Estacionamientos de Visitantes (M2):	90
5.2.2.1. Módulo Monitoreo Estacionamientos	90
5.2.3. Módulo 3 - Interfaces de Usuario (M3):	91
5.2.3.1. Interfaz web	91
5.2.3.2. Aplicación móvil	94
5.2.4. Módulo 4 - Servicios Centrales del Sistema (M4):	95
5.2.4.1. Servicio de Verificación y Registro	95
5.2.4.2. Orquestador de Borde	96
5.3. Revisión del Prototipo	97
5.3.1. Contexto General del Prototipo	97
5.3.2. Esquema General de Componentes del Prototipo	98
5.3.3. Descripción de Componentes	99
5.3.4. Funcionamiento del sistema	100
5.3.4.1. Flujo de Conserjería (Interfaz Web)	101
5.3.4.2. Flujo de Reserva (Aplicación Móvil de Residentes)	102
5.3.4.3. Ciclo de Asignación y Ocupación de Estacionamientos	103
5.4. Verificación y Validación del Sistema	106
5.4.1. Verificación	106
5.4.1.1. Pruebas unitarias de detección	106

5.4.1.2. Integración validada entre API, Colab y base de datos	110
5.4.2. Validación	110
5.4.2.1. YOLO y Patentes	111
5.4.3. Plan de Testing	113
5.5. Gestión de Riesgos	113
5.5.1. Supuestos	113
5.5.2. Dependencias	114
5.5.3. Restricciones	114
5.5.4. Riesgos	115
5.6. Análisis Crítico, Seguridad y Limitaciones	117
5.6.1. Riesgos Técnicos del Prototipo Implementado	117
5.6.2. Riesgos Materializados y Mitigaciones Aplicadas	118
5.6.3. Análisis Extendido de Seguridad y Casos Borde	119
5.6.3.1. Seguridad ante Patentes Falsas	119
5.6.3.2. Seguridad de APIs y Control de Acceso	119
5.6.4. Limitaciones del Sistema	120
6. Conclusiones	122
6.1. Cumplimiento de los Objetivos Planteados	122
6.1.1. OE1 - Análisis de mercado de sistemas inteligentes de seguridad	122
6.1.2. OE2 - Aplicación y entrenamiento de un modelo YOLO para detección de vehículos y patentes	123
6.1.3. OE3 - Sistema de sensores <i>open-source</i> para detección de estacionamientos disponibles	123
6.1.4. OE4 - Aplicación móvil y plataforma web para residentes, conserjes y administra- dores	124
6.1.5. Vínculo con la problemática y el estado del arte	124
6.2. Trabajos futuros	125
6.3. Comentarios	126
Bibliografía	128

Índice de figuras

2.1. Distribución y tasas de homicidios a nivel global por región para el año 2017.	24
2.2. Victimización, percepción de inseguridad y consumo de televisión abierta en Chile, México, Perú y Colombia, según encuestas nacionales oficiales (ENUSC, ENVIPE, ENAPRES, ECSC) y estudios de audiencia (CNTV, Kantar IBOPE Media). El asterisco (*) indica indicadores con definición no equivalente entre países; ver nota metodológica en el texto.	26
2.3. Tasa de homicidios por cada 100 mil habitantes, promedio 2011 - 2020	27
2.4. Tasa de robos violentos por cada 100 mil habitantes, promedio 2011 - 2020	28
2.5. Número de portonazos registrados por año en el sector oriente de la Región Metropolitana, período 2022-2025. [1]	31
2.6. Carencias nacionales de seguridad por año. [2]	32
3.1. Componentes del sistema de control de acceso por radiofrecuencia (RF) en portones residenciales: (a) control de código fijo [3] ; (b) control de código evolutivo [4] ; (c) receptor RF [5]	36
3.2. Flujo tradicional de ingreso de visitas y principales vulnerabilidades asociadas a cada etapa del proceso.	37
3.3. Diagrama de circuito sistema de aparcamiento dinámico, que muestra la integración de Arduino Uno, sensores infrarrojos para la detección de plazas, una pantalla LCD para visualización en tiempo real y un servomotor para el control automatizado de la barrera.	40
3.4. Diagrama de flujo del sistema de detección y monitoreo de vehículos mediante sensores infrarrojos, que ilustra el proceso desde la activación del sensor hasta la actualización dinámica del estado y la visualización.	41
3.5. Función del flujo principal del sistema.	42
3.6. Diagrama de flujo del proceso de detección de placa vehicular: desde la imagen RGB de entrada hasta la localización del contorno rectangular de cuatro vértices que delimita la placa.	43

3.7. Arquitectura de la red neuronal convolucional (CNN) empleada para el reconocimiento de caracteres de patentes vehiculares. La red se compone de tres capas convolucionales y una capa <i>softmax</i> de salida con 36 clases (10 dígitos y 26 letras).	43
3.8. Resultado del proceso de detección de placa vehicular sobre una imagen real: (a) imagen de entrada en formato RGB, (b) conversión a escala de grises, (c) aplicación de suavizado gaussiano, (d) umbralización adaptativa y (e) placa vehicular localizada y delimitada.	44
3.9. Arquitectura general de la red YOLOv11.	45
3.10. Métricas de evaluación del modelo YOLOv11 para detección vehicular en tiempo real.	45
3.11. Resultados visuales de detección obtenidos con la red YOLOv11.	46
3.12. Componentes funcionales de un sistema de control de acceso en comunidades residenciales inteligentes: identificación, autorización y actuación.	47
3.13. Arquitectura integral del sistema VehicleTracker, ilustrando la separación en capas de presentación, aplicación y datos.	50
3.14. Arquitectura general del sistema SPARK, compuesta por seis subsistemas interconectados: WSN, Sink, Gestión de Estacionamiento, Guía Automatizada, Pantalla de Entrada y Cliente.	52
3.15. Capas funcionales de un sistema de estacionamiento inteligente basado en WSN e IoT: Capa de Detección o Sensores, Capa de Red, Capa de Middleware y Capa de Aplicación	
6.	53
4.1. Arquitectura de Sistema ResiPark, modelo 4 capas sistema IoT	70
4.2. Diagrama de Contexto del Sistema.	72
4.3. Diagrama de flujo rediseñado que representa la navegación del usuario en la aplicación ResiPark.	75
4.4. Pantalla de inicio de sesión, interfaz web.	77
4.5. Panel de ocupación de estacionamientos, interfaz web.	77
4.6. Módulo de registro de patentes por departamento, interfaz web.	78
4.7. Módulo de portón y registro de accesos vehiculares, interfaz web.	78
4.8. Pantalla de inicio de sesión, aplicación móvil.	79
4.9. Panel principal de residente, aplicación móvil.	79
4.10. Interfaz de reserva de estacionamientos, aplicación móvil.	80
5.1. Diagrama de flujo de la metodología de cálculo para la matriz de decisión.	83
5.2. Componentes del módulo de control de acceso: (a) integración con relé para accionamiento del portón; (b) integración con cámara para captura de imágenes vehiculares.	88
5.3. Entorno real de los módulos actuando en la recolección de información.	88

5.4. Pipeline de cinco etapas del endpoint /predict: desde la definición de la zona de interés hasta la validación heurística del texto transcrito.	89
5.5. Sensor ultrasónico integrado con módulo de radio LoRa y ESP32 para la detección de estacionamientos.	90
5.6. Flujo de comunicación del sensor de ocupación de estacionamientos: el ESP32 transmisor evalúa el estado a partir del sensor ultrasónico, construye payload correspondiente (libre/ocupado) y lo transmite vía LoRa; el ESP32 receptor recibe el payload, se conecta a la red WiFi mediante TLS y reenvía el estado al backend del sistema mediante una solicitud HTTPS.	91
5.7. Pestaña de ingreso	92
5.8. Pestaña de inicio	92
5.9. Pestaña historial de ingresos	93
5.10. Pestaña Administración de Patentes	93
5.11. Flujo de reserva de estacionamientos en la aplicación móvil implementada: (a) inicio de sesión, (b) panel principal del residente, (c) formulario de reserva.	94
5.12. Arquitectura del sistema de control de acceso: El flujo separa la recolección local de imágenes (Edge) del procesamiento de redes neuronales profundas en la nube, ilustrando el ciclo completo de petición y respuesta.	95
5.13. Diagrama de secuencia del flujo de comunicación en la Capa de Soporte: desde la captura del fotograma hasta la apertura del portón.	97
5.14. Esquema general prototipo	98
5.15. Pestaña de inicio, interfaz web.	101
5.16. Registro de visita, interfaz web.	101
5.17. Flujo de reserva de estacionamientos en la aplicación móvil implementada: (a) inicio de sesión, (b) panel principal del residente, (c) formulario de reserva.	102
5.18. Confirmación, cancelación y estado posterior de una reserva de estacionamiento: (a) ventana emergente de confirmación, (b) menú de cancelación de reservas activas, (c) panel principal del residente tras registrar una reserva.	103
5.19. Asignación de estacionamientos, interfaz web.	103
5.20. Estado y visualización de acceso vehicular, interfaz web.	104
5.21. Estado de estacionamiento ocupado con reserva, interfaz web.	104
5.22. Desasignación de estacionamiento a una reserva, interfaz web.	105
5.23. Estado de estacionamiento ocupado sin reserva, interfaz web.	105
5.24. Dataset de vehículos <i>huanung77</i> (v1), por huang77, Roboflow Universe: 9.689 imágenes (70 % entrenamiento, 20 % validación, 10 % prueba)	107

5.25. Resultados entrenamiento 50 épocas, modelo de vehículos (YOLOv8s)	107
5.26. Matriz de confusión entrenamiento modelo de autos YOLOv8s	108
5.27. Dataset <i>License Plate Recognition</i> (v11), por Roboflow Universe Projects: 10.125 imágenes (70 % entrenamiento, 20 % validación, 10 % prueba)	108
5.28. Resultados entrenamiento 50 épocas, modelo de patentes (YOLOv11s)	109
5.29. Matriz de confusión entrenamiento modelo de patentes YOLOv11s	109
5.30. Salida modelo de detección y OCR	110
5.31. Validación del sistema con video real de acceso vehicular: (a) fotograma de entrada procesado por el modelo de detección; (b) resultado de la verificación mostrando la patente detectada, correctamente identificada como visitante registrado en la aplicación.	111

Índice de tablas

2.1. Resumen comparativo de seguridad ciudadana y consumo audiovisual	27
2.2. Porcentaje de victimización y percepción por quintil socioeconómico. Comparación	
CASEN 2022 y CASEN 2024. [2]	30
3.1. Comparación del rendimiento de modelos de detección de última generación.	46
3.2. Resumen comparativo de enfoques ALPR modernos basados en aprendizaje profundo.	50
3.3. Comparación entre sistemas de gestión de estacionamientos basados en visión por	
computador y en sensores IoT.	54
3.4. Funcionalidades de plataformas de gestión administrativa	56
3.5. Funcionalidades de soluciones de control de acceso físico	57
3.6. Funcionalidades de referentes internacionales de reconocimiento de patentes	58
4.1. Matriz de Trazabilidad de Requisitos y Módulos	73
5.1. Matriz de decisión ponderada para hardware de procesamiento en el borde	84
5.2. Puntuación ponderada total por modelo de detección de objetos	84
5.3. Matriz de decisión ponderada para tecnologías de sensores	85
5.4. Matriz de decisión ponderada para protocolos de comunicación inalámbrica	86
5.5. Puntuación ponderada total por SGBD finalista	86
5.6. Resumen de componentes seleccionados por módulo	87
5.7. Endpoints del servicio de verificación y registro	96
5.8. Descripción de componentes del esquema general del prototipo	99
5.9. Registro de eventos de detección y transcripción de patentes durante pruebas en campo	112
5.10. Riesgos y Medidas de Mitigación	115

Capítulo 1

Introducción

En los últimos años, se ha observado un aumento en el número de ciudades que se declaran inteligentes a nivel global. Estas iniciativas buscan incorporar tecnologías avanzadas para optimizar procesos urbanos y mejorar la calidad de vida de sus habitantes y residentes. Impulsadas por la masificación del acceso a internet, la reducción en el costo de los sensores y el avance sostenido de la inteligencia artificial, estas propuestas han dejado de ser proyectos experimentales para convertirse en soluciones con despliegues reales en ciudades de distintos continentes [7]. En este marco, han comenzado a implementarse soluciones basadas en Inteligencia Artificial (IA), Visión por Computador (CV, del inglés *Computer Vision*) e Internet de las Cosas (IoT, del inglés *Internet of Things*), orientadas a la vigilancia, el control de accesos y la gestión eficiente del entorno urbano. Estas tendencias adquieren especial relevancia en el contexto latinoamericano, región que presenta una de las mayores incidencias de delincuencia, violencia e inseguridad ciudadana a nivel mundial [8]. La inseguridad se ha convertido en uno de los principales problemas que afecta a los países de la región, haciéndose presente en la vida cotidiana de los ciudadanos y deteriorando su calidad de vida tanto en términos económicos como psicológicos. Algunos casos documentados corresponden a: (i) Lima, donde más del 60 % de sus habitantes señala la inseguridad como su principal preocupación [9]; (ii) Bogotá, donde estudios han evidenciado la insuficiencia del gasto público en seguridad para contener la violencia urbana [10]; y (iii) Ciudad de México, donde el control territorial por parte de entidades criminales se ha extendido significativamente, y la actividad policial comunitaria ha resultado insuficiente ante el avance de la delincuencia [11]. Estos casos ilustran la magnitud y diversidad de este fenómeno a lo largo de la región [8]. Esta realidad también se manifiesta en Santiago de Chile, donde la percepción de inseguridad ha aumentado de manera sostenida durante la última década [12]. Entre los factores comunes identificados se encuentran la desigualdad social, el deterioro urbano, la ineficiencia de las políticas públicas y el

crecimiento del crimen organizado [12], que se refleja directamente en la amplia diversidad de tipologías habitacionales existentes en la región: villas, blocks, pasajes, cités, conventillos y condominios; cada una con dinámicas comunitarias y desafíos propios en estas materias.

Entre estas tipologías, las comunidades residenciales de acceso cerrado se han consolidado como una de las modalidades más predominantes en la Región Metropolitana, donde decenas o cientos de familias comparten infraestructura común, incluyendo accesos vehiculares y estacionamientos de visitas [13]. La seguridad y el uso eficiente de estos espacios se han convertido en una preocupación central tanto para los residentes como para los administradores de los espacios comunes. La gestión de estos espacios sigue siendo un proceso esencialmente manual. El control de ingreso vehicular recae habitualmente sobre el personal de conserjería, que debe decidir en tiempo real si autorizar o denegar el acceso a un vehículo, basándose únicamente en criterios subjetivos o, en algunos casos, en la autorización de un residente. Esta dependencia del criterio humano puede introducir ciertas vulnerabilidades, tales como: (i) personas externas pueden persuadir al conserje para ingresar sin autorización formal, (ii) vehículos no registrados ocupan estacionamientos de visita y de residentes sin control de tiempo, y (iii) no existe trazabilidad confiable de los movimientos al interior del recinto. Frente a este escenario, las tecnologías de visión por computador, sensores inteligentes y aplicaciones móviles ofrecen una vía concreta para automatizar y robustecer el control de accesos en comunidades residenciales de acceso cerrado. La integración de estas herramientas permite identificar y registrar el flujo vehicular de forma automática, monitorear en tiempo real el uso de espacios compartidos y mantener una comunicación fluida entre residentes, administradores y personal, reduciendo la dependencia del criterio humano y aumentando la trazabilidad de los accesos. Si bien existen soluciones comerciales [14] en esta línea, muchas son parciales, de alto costo o difíciles de adaptar a contextos locales, sin contemplar particularidades como las normativas chilenas vigentes de protección de datos personales [15] y la Ley de Copropiedad Inmobiliaria [16]. Esta situación abre un espacio para el desarrollo de propuestas integrales, modulares y económicamente accesibles, adaptadas a las necesidades reales de las comunidades residenciales de acceso cerrado. En este trabajo se presenta un sistema inteligente para el control de accesos vehiculares en comunidades residenciales de acceso cerrado, denominado *ResiPark*. El sistema integra visión por computador para el reconocimiento automático de patentes y tipo de vehículo, una red de sensores IoT para el monitoreo de estacionamientos de visita, y aplicaciones web y móvil para la gestión por parte de residentes, conserjes y administradores de estos espacios comunes, cuyo objetivo es reemplazar el control manual y subjetivo por un sistema automatizado, trazable y seguro, que contribuya a mejorar la calidad de vida y la seguridad de las comunidades residenciales.

1.1. Objetivos

Con el propósito de dar respuesta a las problemáticas identificadas, se define un conjunto de objetivos que guían el diseño e implementación del sistema. Estos se estructuran en un objetivo general, que establece el alcance global de la solución, y en objetivos específicos, que desglosan las áreas técnicas concretas a abordar a lo largo del desarrollo.

1.1.1. Objetivo General

Diseñar e implementar un sistema inteligente de seguridad para comunidades residenciales de acceso cerrado, orientado a optimizar la protección de los residentes y reducir vulnerabilidades ante accesos no autorizados, mediante el uso integrado de tecnologías de automatización de procesos, microcontroladores, sensores IoT, aplicaciones móviles y plataformas web de monitoreo en tiempo real.

1.1.2. Objetivos Específicos

- **OE1.** Realizar un análisis de mercado de los sistemas inteligentes de seguridad, identificando al menos cinco tecnologías, metodologías o buenas prácticas aplicables para el montaje y desarrollo del sistema propuesto.
- **OE2.** Aplicar y entrenar un modelo de visión por computador basado en la arquitectura *YOLO* para la detección automática de vehículos, así como de sus respectivas placas patentes.
- **OE3.** Crear y desplegar un sistema de sensores basado en hardware *open-source* que permita detectar espacios de estacionamiento disponibles, transmitiendo dicha información en tiempo real.
- **OE4.** Diseñar e implementar una aplicación móvil para residentes y una plataforma web para conserjes y administradores, que integren funcionalidades críticas de registro, control de acceso, notificaciones, historial y gestión de reservas.

1.2. Propósito

El propósito del presente trabajo es contribuir a la mejora de la seguridad y la calidad de vida en comunidades residenciales de acceso cerrado, mediante el diseño e implementación de un sistema inteligente que automatice y centralice el control de accesos vehiculares y la gestión de estacionamientos. A través de la integración de tecnologías de visión por computador, sensores IoT e interfaces, se busca reemplazar los procesos manuales actuales por una solución trazable, escalable y adaptada al contexto chileno, que pueda ser adoptada por distintos tipos de comunidades residenciales de acceso cerrado de la Región Metropolitana.

1.3. Contribuciones

Se presenta evidencia empírica de las pruebas, entrenamiento y despliegue de dos modelos de *YOLO*: v11 para la detección de patentes y v8 para la detección de vehículos, ambos integrados a una lógica que restringe las detecciones mediante una zona de interés. Además, se implementa el modelo *Florence-2 OCR* para la transcripción de los caracteres en cada patente detectada. Estos modelos coexisten en una única lógica.

Por otra parte, se desarrolla un backend que se ejecuta en una Raspberry Pi 4, cuya finalidad es enviar las imágenes en tiempo real capturadas mediante una cámara IP, a través de un *endpoint*, a los modelos de detección y transcripción, y almacenar la información en una base de datos. Además, se implementan dos endpoints, uno para la verificación de cada patente detectada por los modelos, y otro para su correcta inserción en la base de datos.

Se desarrolla un frontend orientado a los tres perfiles de usuario del sistema: residentes, conserjes y administradores. Por un lado, los residentes pueden gestionar reservas de estacionamientos de visita a través de una aplicación móvil. Por otro lado, conserjes y administradores acceden a una aplicación web desde la cual pueden monitorear en tiempo real los accesos vehiculares mediante cámara, consultar la información obtenida de la lectura de patentes y administrar las reservas activas dentro de la comunidad residencial de acceso cerrado.

1.4. Estructura del Informe

La estructura del presente documento se organiza de la siguiente manera:

- En el Capítulo [2](#) se aborda en detalle la problemática que enmarca este proyecto, incluyendo sus distintas dimensiones: el control manual de accesos vehiculares, la gestión ineficiente de estacionamientos de visita y la falta de trazabilidad en los recintos residenciales. Asimismo, se analizan problemáticas afines que comparten componentes técnicos o contextuales con la solución propuesta, con el fin de delimitar con precisión el alcance del sistema.
- En el Capítulo [3](#) se revisa el estado del arte y las soluciones tecnológicas existentes para resolver este problema y sus subproblemas. Se examinan las herramientas, metodologías y arquitecturas que han sido empleadas en casos similares, incluyendo tanto experiencias exitosas como limitaciones identificadas en implementaciones previas, tanto en el ámbito del control de accesos vehiculares como en la gestión de estacionamientos y el monitoreo mediante sensores IoT.
- En el Capítulo [4](#) se describe la metodología de desarrollo y la arquitectura general del sistema, abarcando el diseño de la comunicación entre componentes, el flujo de autenticación, el entrenamiento y despliegue de los modelos de visión por computador, la lógica de verificación vehicular y la integración de los sensores IoT, todo ello complementado con los *mockups* de las aplicaciones web y móvil.
- En el Capítulo [5](#) se exponen los resultados obtenidos a lo largo del desarrollo e implementación del sistema. Se presentan las métricas de entrenamiento y validación de los modelos de visión por computador, incluyendo su desempeño en condiciones reales de operación. Asimismo, se describe la implementación del sistema en un entorno real, detallando el despliegue de los componentes de hardware y software, la integración entre los módulos de detección vehicular, sensores IoT y las aplicaciones web y móvil. Finalmente, se valida el funcionamiento del flujo completo del sistema, desde la captura de imágenes y la lectura de patentes hasta la gestión de reservas y el monitoreo de estacionamientos.
- Finalmente, en el Capítulo [6](#) se presentan las conclusiones de este trabajo y se proponen trabajos futuros relevantes.

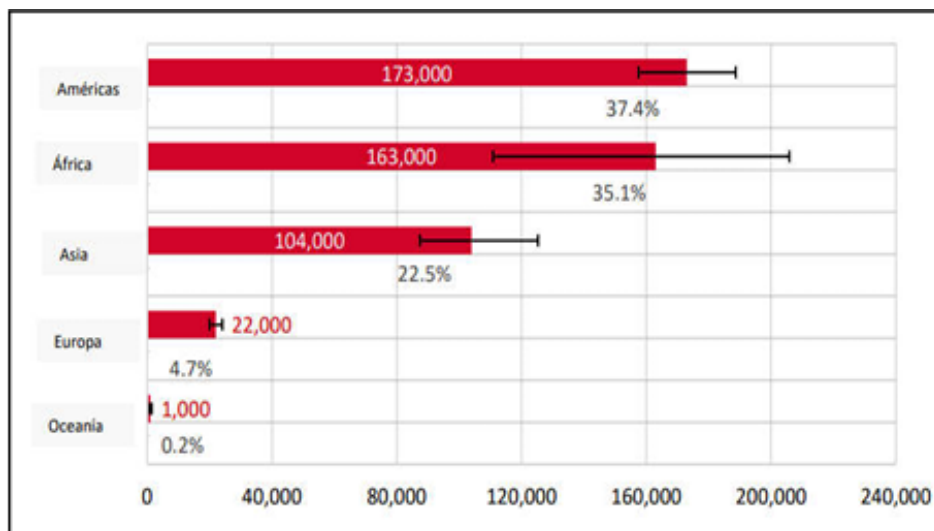
Capítulo 2

Problemática

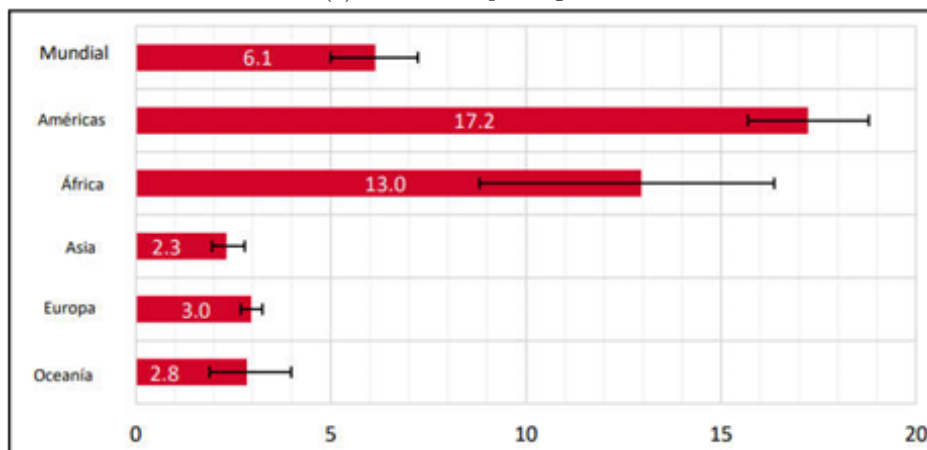
La inseguridad ciudadana constituye uno de los fenómenos sociales más complejos y de mayor impacto en la calidad de vida de las personas [17]. Sus manifestaciones son diversas y atraviesan distintos ámbitos de la vida cotidiana, afectando tanto los espacios públicos como los residenciales, así como la percepción subjetiva de seguridad y las condiciones objetivas en las que los ciudadanos se desenvuelven día a día. En este capítulo se aborda la inseguridad como fenómeno regional, partiendo desde una mirada amplia sobre Latinoamérica y descendiendo progresivamente hacia el contexto específico de Santiago de Chile. A partir de este encuadre, se identifican y analizan las principales problemáticas derivadas de la inseguridad que afectan a los habitantes de la capital, distinguiendo dos ámbitos centrales: los espacios de acceso público y los espacios residenciales. Finalmente, se delimita con precisión la problemática específica que aborda este trabajo, sentando las bases para la propuesta de solución desarrollada en los capítulos siguientes.

2.1. Inseguridad en Latinoamérica

América Latina se ha consolidado históricamente como una de las zonas con mayores índices de violencia criminal a nivel global. A pesar de albergar aproximadamente al 13 % de la población mundial, estudios estadísticos indican que la región concentra el 37 % (véase en Figura 2.1a) de los homicidios a nivel mundial [18], con una tasa de 17,2 homicidios por cada 100.000 habitantes (véase en Figura 2.1b), cifra que supera de manera significativa el promedio global. Este panorama de violencia objetiva no solo afecta la integridad física de las personas, sino que condiciona y restringe el desarrollo social y urbano de sus principales ciudades.



(a) Homicidios por región.



(b) Tasa por cada 100 mil habitantes.

Figura 2.1: Distribución y tasas de homicidios a nivel global por región para el año 2017.

En paralelo a las cifras objetivas de criminalidad, se ha gestado un fenómeno sociológico de igual o mayor impacto: el crecimiento sostenido de la percepción de inseguridad ciudadana. Durante la última década, la delincuencia se ha posicionado como la principal preocupación para los habitantes de los países latinoamericanos, desplazando de la discusión pública a temas estructurales tradicionales como el desempleo, la educación o la desigualdad económica. Esta preocupación colectiva no se explica de manera unidimensional a través de la victimización directa, sino que se nutre de un entorno simbólico y empírico donde las incivildades y el desorden social actúan como predictores del miedo al crimen [19]. En este sentido, la constante exposición a un contexto urbano percibido como criminógeno o falta de control formal, eleva drásticamente la sensación de vulnerabilidad situacional de los ciudadanos. Por otra parte, es imperativo señalar que la dimensión subjetiva de la inseguridad no responde de manera

lineal a la evolución de la criminalidad objetiva.

De acuerdo con la Encuesta Nacional Urbana de Seguridad Ciudadana (ENUSC) 2025 del Instituto Nacional de Estadísticas de Chile, el 37,7 % de los hogares a nivel nacional fue víctima de al menos un delito durante los últimos doce meses, cifra que no presentó variación estadísticamente significativa respecto a 2024 [20]. En cuanto a la percepción de inseguridad, el 82,9 % de las personas percibió un aumento de la delincuencia en el país durante los últimos doce meses, lo que representa una disminución estadísticamente significativa de 4,8 puntos porcentuales respecto a 2024 [20]. Respecto al consumo de medios tradicionales, un estudio del Consejo Nacional de Televisión (CNTV), difundido en 2023, señaló que el 64 % de la población optaba por la televisión abierta como medio de consumo audiovisual [21].

En México, la Encuesta Nacional de Victimización y Percepción sobre Seguridad Pública (ENVIPE) 2025 del Instituto Nacional de Estadística y Geografía estimó que el 29,0 % de los hogares tuvo al menos un integrante víctima de algún delito durante 2024 [22]. En cuanto a la percepción de inseguridad, la Encuesta Nacional de Seguridad Pública Urbana (ENSU) reportó que, en junio de 2026, el 59,8 % de la población de 18 años y más residente en 91 áreas urbanas de interés consideró que era inseguro vivir en su ciudad, cifra que representó un cambio estadísticamente significativo respecto a marzo de 2026 (61,5 %) y a junio de 2025 (63,2 %) [23]. Respecto al consumo de medios tradicionales, la Encuesta Nacional sobre Disponibilidad y Uso de Tecnologías de la Información en los Hogares (ENDUTIH) 2025 del INEGI reportó que el 49,1 % de la población de 6 años y más vio televisión abierta durante ese año, cifra en descenso sostenido respecto al 61,1 % registrado en 2020 [24].

En el Perú, el Instituto Nacional de Estadística e Informática (INEI), a través de la Encuesta Nacional de Programas Presupuestales (ENAPRES), reportó que, en el semestre móvil setiembre 2025-febrero 2026, el 24,4 % de la población urbana de 15 años y más fue víctima de algún hecho delictivo, cifra que mantiene una tendencia decreciente respecto a semestres anteriores [25]. En cuanto a la percepción de inseguridad, el 82,4 % de la misma población manifestó percibir que, en los próximos doce meses, podría ser víctima de algún hecho delictivo en cualquier lugar del país [25]. Respecto al consumo de medios tradicionales, el estudio Cross Platform View de Kantar IBOPE Media (setiembre de 2025) indicó que la televisión lineal – que agrupa señal abierta y de pago – alcanzó una cobertura mensual del 91 % de la audiencia peruana, frente a un 49 % de las plataformas digitales [26].

Por su parte, en Colombia, la Encuesta de Convivencia y Seguridad Ciudadana (ECSC) 2024 del Departamento Administrativo Nacional de Estadística (DANE) indicó que el 8,4 % de las personas de 15 años y más a nivel nacional fue víctima de al menos un delito durante 2023 [27]. En cuanto a la percepción de inseguridad, el 46,3 % de la población de 15 años y más manifestó sentirse insegura en su

ciudad o municipio durante el periodo octubre–diciembre de 2024, reflejando una disminución de 6,6 puntos porcentuales respecto a la medición de 2022 [27]. En relación con el consumo audiovisual, un estudio de Kantar IBOPE Media correspondiente al primer semestre de 2025 reportó que el 69% de los consultados consumió televisión abierta, superando a la televisión paga (17%) y a las plataformas de streaming (14%) (véase en Figura 2.2). Estos resultados reflejan la coexistencia de medios tradicionales y formatos digitales [28].

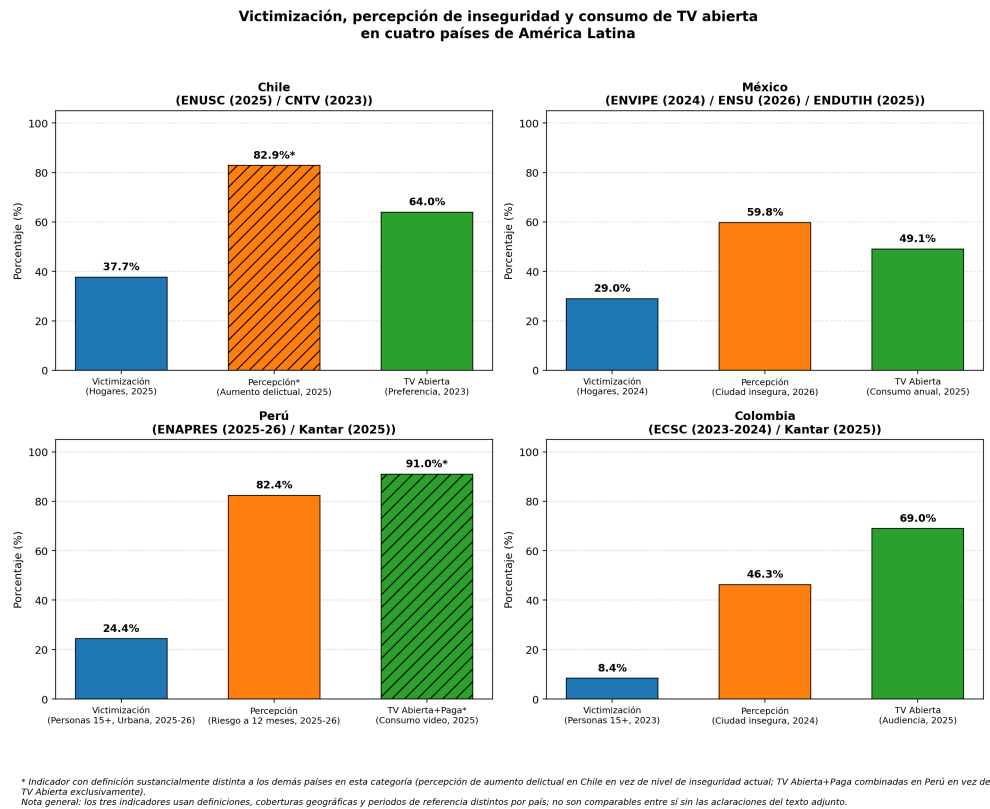


Figura 2.2: Victimización, percepción de inseguridad y consumo de televisión abierta en Chile, México, Perú y Colombia, según encuestas nacionales oficiales (ENUSC, ENVIPE, ENAPRES, ECSC) y estudios de audiencia (CNTV, Kantar IBOPE Media). El asterisco (*) indica indicadores con definición no equivalente entre países; ver nota metodológica en el texto.

Al analizar estos cuatro casos de la región, se puede observar una correlación entre el consumo de medios de comunicación y la percepción de inseguridad reportada en cada país; sin embargo, esto no implica causalidad. Estos resultados dan lugar a plantear que los medios de comunicación y las plataformas digitales pueden contribuir a la construcción social del miedo, especialmente mediante la exposición frecuente a contenidos asociados con la violencia y el delito. Este fenómeno comunicacional genera una marcada brecha perceptiva: el temor ciudadano a convertirse en víctima de un ilícito violento suele superar con creces las probabilidades estadísticas reales calculadas a partir de las denuncias

formales de los organismos policiales (véase en Tabla 2.1).

Tabla 2.1: Resumen comparativo de seguridad ciudadana y consumo audiovisual

Dimensión / Indicador	Chile	México	Perú	Colombia
1. Victimización				
Tasa observada	37,7 %	29,0 %	24,4 %	8,4 %
Unidad de análisis	Hogares	Hogares	Personas (15+)	Personas (15+)
Cobertura geográfica	Nacional	Nacional	Urbana	Nacional
Fuente	ENUSC (2025)	ENVIPE (2024)	ENAPRES (2025-26)	ECSC (2023)
2. Percepción de Inseguridad				
Nivel reportado	82,9 %	59,8 %	82,4 %	46,3 %
Constructo medido	Aumento delictual	Ciudad insegura	Riesgo a 12 meses	Ciudad insegura
Fuente	ENUSC (2025)	ENSU (2026)	ENAPRES (2025-26)	ECSC (2024)
3. Consumo de TV Abierta / Lineal				
Penetración / Audiencia	64,0 %	49,1 %	91,0 %	69,0 %
Tipo de métrica	Preferencia	Consumo anual	Consumo video	Audiencia
Año / Fuente	2023 (CNTV)	2025 (ENDUTIH)	2025 (Kantar)	2025 (Kantar)

Al analizar los indicadores presentados en la tabla resumen (2.1), se evidencian algunas "correlaciones", pero no se "prueba" ningún tipo de "incidencia" de que los medios de comunicación afecten la percepción de inseguridad en estos países de la región. A pesar de ello, mientras algunos países de Centroamérica y del norte de Sudamérica presentan tasas críticas de homicidios, países como Chile exhiben indicadores notablemente inferiores para este delito específico, registrando tasas cercanas a los 3,3 casos por cada 100.000 habitantes (véase en Figura 2.3). Sin embargo, esta baja incidencia en delitos contra la vida no se traduce en una percepción de seguridad generalizada entre sus habitantes, fenómeno que se explica al analizar la tasa de robos violentos de la región (véase en Figura 2.4) [18].

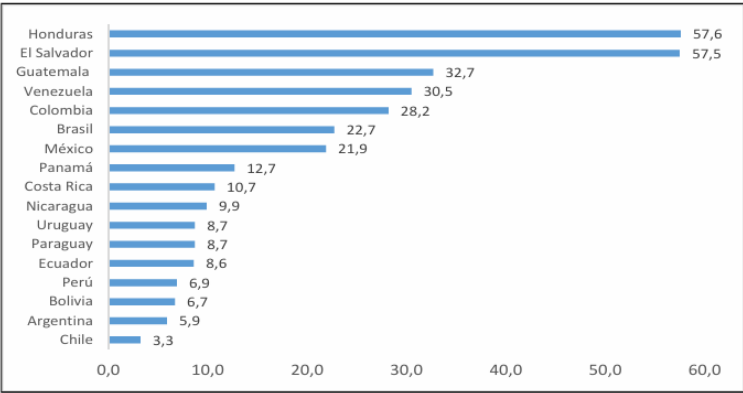


Figura 2.3: Tasa de homicidios por cada 100 mil habitantes, promedio 2011 - 2020

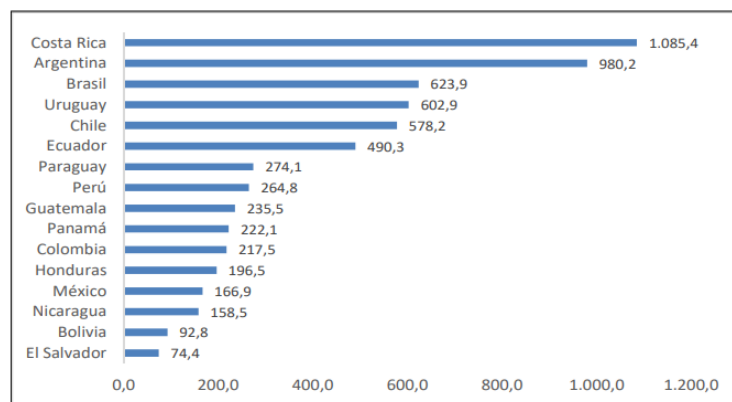


Figura 2.4: Tasa de robos violentos por cada 100 mil habitantes, promedio 2011 - 2020

Al analizar la distribución de estos indicadores, se observa que la violencia no se manifiesta de manera homogénea, sino que tiende a concentrarse críticamente en los principales polos metropolitanos. Estas urbes generan microclimas de criminalidad que superan y, en ocasiones, ocultan las verdaderas proporciones de los promedios nacionales. Por ejemplo, en Ecuador, la ciudad de Guayaquil se ha consolidado como un epicentro de conflictividad urbana extrema; durante 2023, este cantón concentró casi el 29 % de todos los homicidios del país, registrando tasas cercanas a los 90 casos por cada 100.000 habitantes a causa de disputas territoriales del crimen organizado [29]. Un fenómeno de segmentación urbana similar ocurre en Argentina, donde el promedio nacional de homicidios se mantiene en un nivel moderado de aproximadamente 6 casos por cada 100.000 habitantes. Sin embargo, la ciudad de Rosario triplica esta cifra con una tasa local de 16,4 homicidios cada 100.000 habitantes, evidenciando una crisis de seguridad profundamente focalizada [30].

Por otro lado, en metrópolis donde la violencia letal no constituye el indicador predominante, una parte importante de la inseguridad se relaciona con delitos patrimoniales de alta frecuencia en el espacio público. En el caso de Bogotá, Colombia, tras las flexibilizaciones de movilidad post-pandemia, los registros de hurto a personas experimentaron un alza significativa. Los estudios de criminalidad evidencian que en sectores neurálgicos, como los grandes ejes de transporte, las tasas de hurto superaron en algunos meses los 500 casos por cada 100.000 habitantes, convirtiéndose en el factor que más influye estadísticamente en la sensación de desprotección y miedo de los transeúntes [31]. Esta multiplicidad de fenómenos a lo largo de la región ilustra cómo la criminalidad somete a los ciudadanos a distintos niveles de vulnerabilidad dependiendo de las dinámicas propias de la ciudad que habitan.

El contexto delictual chileno y particularmente el de su capital, Santiago, se caracteriza por una alta incidencia de delitos contra la propiedad mediante el uso de fuerza e intimidación. Las estadísticas regionales sitúan a Chile dentro de los países con mayores registros de robos violentos,

alcanzando una tasa de 578,2 casos por cada 100.000 habitantes [18]. Esta realidad contribuye a contextualizar la alta percepción de inseguridad de la población santiaguina: el temor ciudadano no se basa predominantemente en el riesgo de homicidio, sino en la alta probabilidad de ser víctima de delitos patrimoniales violentos en las rutinas de la vida cotidiana.

Frente a este escenario de inseguridad en el entorno público, los ciudadanos pueden modificar sus dinámicas de interacción con el entorno urbano. La vulnerabilidad percibida en las calles y espacios de acceso abierto ha fomentado un repliegue progresivo hacia los espacios residenciales, buscando en el hogar y en las comunidades de acceso cerrado un refugio frente a la criminalidad externa. No obstante, las vulnerabilidades intrínsecas de los sistemas de seguridad residenciales tradicionales han demostrado que estos recintos no están exentos de amenazas, trasladando el desafío del control y la trazabilidad al ámbito privado.

2.2. Problemas derivados de la Inseguridad en Santiago de Chile

La Región Metropolitana concentra los mayores niveles de exposición a hechos de inseguridad del país. Según datos de la Encuesta CASEN 2024, esta región lidera los registros nacionales en prácticamente todos los indicadores analizados: el 46 % de los hogares declara haber presenciado rayados o daños a viviendas y vehículos, el 73 % reporta consumo de drogas o alcohol en su entorno residencial, el 40 % señala haber observado tráfico de drogas, y el 47 % afirma haber presenciado balaceras o disparos en el último mes (véase en Tabla 2.2) [2]. Estas cifras posicionan a la Región Metropolitana como uno de los territorios con mayor exposición a eventos violentos e incivildades del país, configurando un escenario de inseguridad que se manifiesta con intensidades distintas según el ámbito en que los ciudadanos se desenvuelven. En particular, es posible distinguir dos dimensiones centrales donde esta problemática adquiere expresiones concretas y diferenciadas: los espacios de acceso público y los espacios residenciales.

Tabla 2.2: Porcentaje de victimización y percepción por quintil socioeconómico. Comparación CASEN 2022 y CASEN 2024. [2]

Quintil	Rayados o daños a casas o vehículos		Consumo de drogas o alcohol		Traficando drogas		Presenciado balaceras o disparos	
Año	2022	2024	2022	2024 %	2022	2024	2022	2024
I	33,40 %	31,0 %	57,00 %	57,5 %	38,80 %	37,0 %	43,30 %	41,6 %
II	36,80 %	33,1 %	62,30 %	61,2 %	41,50 %	37,8 %	46,00 %	42,5 %
III	37,70 %	34,8 %	63,40 %	61,8 %	39,60 %	36,1 %	44,60 %	40,4 %
IV	39,20 %	36,1 %	62,50 %	61,6 %	36,50 %	33,0 %	42,20 %	36,5 %
V	42,80 %	38,1 %	60,00 %	58,3 %	27,00 %	23,3 %	30,30 %	25,2 %
Nacional	38,0 %	34,6 %	61,0 %	60,1 %	36,7 %	33,4 %	41,3 %	37,3 %

2.2.1. Espacios de Acceso Público

El espacio público constituye uno de los principales ámbitos en los que los habitantes de Santiago experimentan inseguridad. Chile presenta una de las tasas más altas de robos violentos en América Latina, alcanzando 578,2 casos por cada 100.000 habitantes en promedio entre 2011 y 2020, situándose entre los cinco países con mayor registro de este delito en la región [18]. Esta realidad se relaciona con la percepción de riesgo que experimentan los ciudadanos al transitar por la ciudad, alterando sus rutinas y condicionando su relación con el entorno urbano.

Los datos de la Encuesta Nacional Urbana de Seguridad Ciudadana (ENUSC) 2024 refuerzan este diagnóstico. Según sus resultados, el 23,5 % de los hogares chilenos declaró haber sido víctima de algún delito durante el último año, mientras que el 87,5 % de la población percibe que la delincuencia ha aumentado en su barrio [12]. Esta marcada brecha entre victimización real y percepción de inseguridad refleja un fenómeno en que el miedo al delito supera con creces las probabilidades estadísticas de ser víctima, condicionando de manera profunda las decisiones cotidianas de los ciudadanos respecto al uso del espacio público.

Esta evolución ha consolidado a la delincuencia como la principal preocupación de la ciudadanía, superando de forma transversal a otras problemáticas como el desempleo, la salud y la pobreza. Según la Encuesta CEP, entre mayo de 2022 y junio de 2024, la proporción de personas que mencionaban la delincuencia como su principal inquietud aumentó del 50 % al 57 % de la población [2].

Una de las expresiones más visibles de esta inseguridad en el espacio público es el robo violento de vehículos, modalidad que en la Región Metropolitana ha adquirido patrones delictivos altamente organizados: los denominados portonazos, robos que ocurren al momento en que el conductor ingresa o egresa de su domicilio mediante un portón, y las encerronas, interceptaciones vehiculares que ocurren

en la vía pública. Según un informe de la PDI, las comunas de la Región Metropolitana con mayor concentración de encerronas corresponden a sectores de alta densidad vehicular y tránsito intenso, evidenciando que este delito responde a una lógica territorial y de oportunidad deliberadamente planificada por bandas organizadas [32]. Complementariamente, solo en el sector oriente de la Región Metropolitana, bajo la jurisdicción de la Fiscalía Metropolitana Oriente, se registraron 6.769 portonazos entre 2022 y 2025, con un máximo de 2.077 casos en 2022 [1]. Si bien el año 2025 mostró una reducción significativa hasta los 1.149 casos, los registros acumulados (véase en Figura 2.5) dan cuenta de la magnitud del fenómeno y de su capacidad para reorganizarse territorialmente ante medidas preventivas focalizadas.

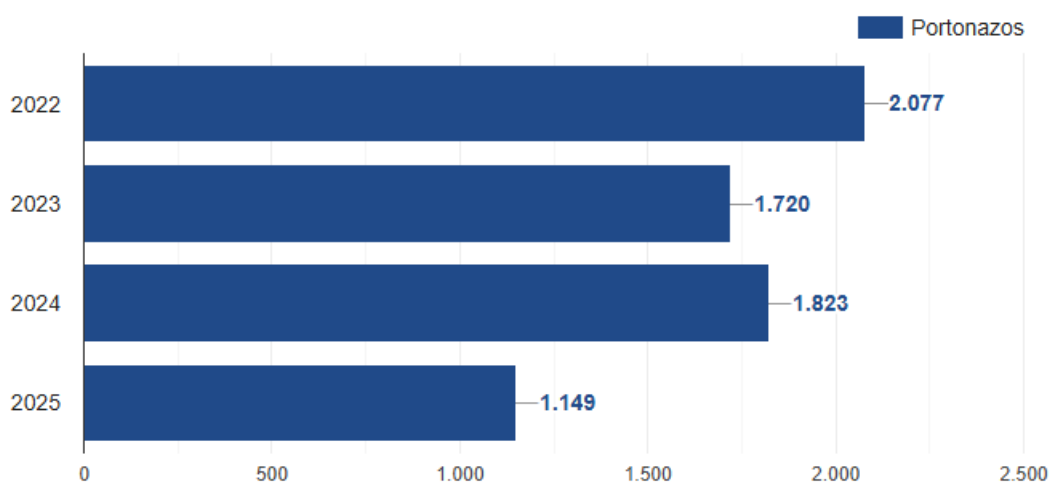


Figura 2.5: Número de portonazos registrados por año en el sector oriente de la Región Metropolitana, período 2022–2025. [1]

Esta forma de criminalidad es especialmente relevante para el presente trabajo, pues opera precisamente en la zona de transición entre el espacio público y el espacio residencial: el acceso vehicular al domicilio. La vulnerabilidad se concentra en ese instante en que el vehículo se detiene frente al portón de ingreso, evidenciando que la frontera entre el peligro exterior y la seguridad del hogar no está determinada únicamente por las paredes del domicilio, sino también por la eficiencia y robustez de sus sistemas de control de acceso.

2.2.2. Espacios Residenciales

Frente a la inseguridad percibida en el espacio público, el entorno residencial puede adquirir un papel de mayor relevancia como espacio de protección, buscando en el hogar y en la comunidad un espacio protegido. Sin embargo, las cifras revelan que esta protección es parcial e inequitativa. Según

los datos de la CASEN 2024, el 10,3 % de los hogares chilenos presenta carencia de seguridad, entendida como la exposición severa y persistente a hechos graves en el entorno residencial como el tráfico de drogas o los disparos [2]. Esta cifra, aunque muestra una leve mejora respecto al 11 % registrado en 2015, evidencia que más de una década de políticas públicas no ha logrado resolver de manera sustantiva la problemática en los entornos habitacionales (véase Figura 2.6).

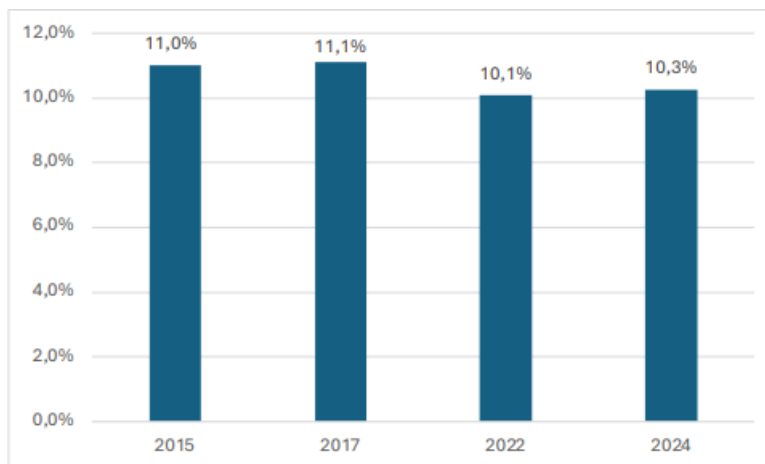


Figura 2.6: Carencias nacionales de seguridad por año. [2]

La distribución de esta carencia no es homogénea. Los hogares de menores ingresos presentan sistemáticamente mayores niveles de exposición a hechos de inseguridad: mientras en el quintil de menores ingresos la carencia alcanza el 12,5 %, en el quintil de mayores ingresos desciende al 4,7 % [2]. Esta brecha se acentúa en hogares con niños, niñas y adolescentes del primer quintil, donde la carencia llega al 14,9 %, triplicando los niveles del mismo grupo en el quintil más alto. La desigualdad territorial en el acceso a entornos seguros es, por tanto, una dimensión estructural del problema habitacional en Santiago.

Esta búsqueda de protección en el espacio residencial ha derivado en respuestas comunitarias diversas. Estudios realizados en barrios de la capital documentan cómo el sentimiento de desprotección que experimentan los habitantes moviliza prácticas de cuidado vecinal orientadas a gestionar la incertidumbre que produce el peligro [33]. Estas prácticas se materializan en esquemas de vigilancia compartida, sistemas de alarmas comunitarias y una mayor organización vecinal, promovidas tanto por iniciativas ciudadanas como por políticas públicas de prevención situacional que buscan reducir las condiciones físico-ambientales que facilitan la ocurrencia del delito.

Santiago concentra una amplia diversidad de tipologías habitacionales que condicionan de manera distinta la forma en que sus habitantes experimentan y gestionan la inseguridad. Desde las tomas, ranchos, cités y conventillos, hasta las villas, blocks, pasajes y condominios, cada modalidad habitacional

presenta dinámicas comunitarias, grados de cohesión social y niveles de control de acceso particulares [34]. En este espectro, las edificaciones residenciales de altura y las comunidades residenciales de acceso cerrado han ganado una presencia creciente en el paisaje urbano de la Región Metropolitana, al ofrecer, al menos en apariencia, un mayor grado de control sobre quién ingresa y egresa del recinto.

2.2.2.1. Control de Acceso en Comunidades Residenciales

El control de acceso constituye uno de los aspectos más críticos de la seguridad en las comunidades residenciales de acceso cerrado. En estas tipologías habitacionales, la frontera entre el espacio público y el privado se materializa en el portón vehicular y en los accesos peatonales, puntos que concentran tanto el riesgo de intrusión como la oportunidad de ejercer un control efectivo sobre el flujo de personas y vehículos.

Sin embargo, en la mayoría de estas comunidades, este control sigue siendo ejercido de manera manual por el personal de conserjería, que debe evaluar en tiempo real la autorización de cada ingreso basándose en criterios subjetivos y sin herramientas de verificación sistemática. Esta dependencia del juicio humano introduce vulnerabilidades concretas: la posibilidad de que personas no autorizadas persuadan al conserje para ingresar, la ausencia de trazabilidad en los registros de acceso, y la dificultad para gestionar eficientemente el uso de los estacionamientos de visita. A esto se suma que el personal de conserjería debe atender múltiples tareas simultáneamente, reduciendo la atención disponible para un control riguroso de los accesos y una posible exposición directa a actos delictuales.

El fenómeno de los portonazos ilustra con precisión el punto de mayor vulnerabilidad de este modelo: el instante en que un vehículo autorizado aguarda la apertura del acceso, momento en que queda expuesto en la vía pública y en que la demora o ineficiencia del sistema de apertura puede ser aprovechada por delincuentes. La tecnificación del control de acceso emerge entonces no solo como una mejora operativa, sino como una respuesta concreta a una vulnerabilidad documentada y recurrente en las comunidades residenciales de la Región Metropolitana.

2.3. Problemática

Las secciones precedentes permiten delimitar con precisión la problemática que aborda este trabajo. En el contexto de Santiago de Chile, las comunidades residenciales de acceso cerrado enfrentan dos desafíos concretos e interrelacionados en materia de seguridad, cuya resolución requiere de una aproximación tecnológica integrada y adaptada al contexto local.

El primero es el control manual e ineficiente de los accesos vehiculares. La ausencia de mecanismos automáticos de identificación y verificación implica que el ingreso de vehículos depende exclusivamente del criterio del personal de conserjería, sin respaldo tecnológico, sin trazabilidad sistemática y sin capacidad de respuesta en tiempo real ante accesos no autorizados. Esta dependencia del factor humano no solo introduce subjetividad en el proceso de autorización, sino que expone a la comunidad a riesgos concretos: un conserje puede ser persuadido para permitir el ingreso de un vehículo no registrado, puede cometer errores de identificación bajo presión o simplemente puede no estar disponible en el momento crítico en que un vehículo intenta acceder. Esta vulnerabilidad se agrava al considerar que el robo violento de vehículos en el acceso al domicilio es una de las modalidades delictivas más frecuentes y organizadas en la Región Metropolitana, como lo evidencian los registros de portonazos y encerronas analizados en las secciones anteriores. La demora en la apertura del portón, producto de un proceso de verificación lento y manual, se convierte así en un factor de riesgo concreto para los residentes.

El segundo desafío es la gestión ineficiente de los estacionamientos de visita. En la mayoría de las comunidades residenciales, el monitoreo de la ocupación de estos espacios se realiza de forma manual o, en ciertas situaciones, simplemente no se lleva a cabo, lo que genera conflictos entre residentes, uso indebido de los espacios comunes y ausencia de información en tiempo real. La falta de un sistema de reservas y monitoreo impide garantizar que los estacionamientos de visita sean utilizados conforme a las políticas internas de la comunidad, generando situaciones de ocupación prolongada, uso no autorizado y disputas vecinales de difícil resolución sin evidencia objetiva.

Ambas problemáticas comparten una causa raíz: la falta de automatización y digitalización de los procesos de control en el acceso residencial. Los sistemas actualmente disponibles en el mercado no ofrecen una solución integral que aborde de manera simultánea el control de accesos vehiculares y la gestión de estacionamientos, y los que existen suelen ser de alto costo, difíciles de adaptar a la infraestructura particular de cada comunidad o no contemplan las normativas locales vigentes. Esta brecha en la oferta tecnológica, sumada a la creciente demanda de seguridad por parte de los habitantes de la Región Metropolitana, configura un escenario propicio para el desarrollo de soluciones innovadoras, modulares y económicamente accesibles.

Ante este escenario, el presente trabajo propone el diseño e implementación de *ResiPark*, un sistema inteligente que integra tecnologías de visión por computador, sensores IoT (del inglés *Internet of Things*) y aplicaciones digitales para automatizar el control de accesos vehiculares y la gestión de estacionamientos en comunidades residenciales de acceso cerrado, contribuyendo a reducir las vulnerabilidades identificadas y a mejorar la seguridad y la calidad de vida de sus habitantes.

Capítulo 3

Estado del Arte

El presente capítulo revisa el estado del arte y las soluciones tecnológicas existentes orientadas a resolver la problemática descrita en el capítulo anterior. El análisis se estructura de manera progresiva: se parte desde los sistemas tradicionales de monitoreo y control de acceso empleados actualmente en comunidades residenciales de acceso cerrado, para luego examinar la evolución hacia modelos de gestión inteligente impulsados por tecnologías de Inteligencia Artificial e Internet de las Cosas. A continuación, se profundizan los subproblemas identificados, incluyendo el reconocimiento automático de patentes y la gestión de estacionamientos mediante visión por computador y sensores IoT. Finalmente, se analizan las principales soluciones comerciales disponibles en el mercado local, identificando sus capacidades, limitaciones y el espacio que dejan para el desarrollo de propuestas más integrales y adaptadas al contexto chileno.

3.1. Sistemas de Monitoreo y Acceso Tradicionales en Comunidades Residenciales de Acceso Cerrado

Durante décadas, la seguridad física y la gestión del flujo vehicular en comunidades residenciales de acceso cerrado, como condominios, pasajes cerrados y edificios residenciales han dependido predominantemente de esquemas de control manual e infraestructura analógica o de baja automatización [35]. En la mayoría de estos recintos, la primera línea de defensa se compone de barreras físicas electromecánicas (portones levadizos, plumas o rejas correderas) operadas de forma directa por el personal de conserjería o mediante dispositivos de radiofrecuencia (RF) (véase en Figura 3.1) de accionamiento manual entregados a los residentes [36].



Figura 3.1: Componentes del sistema de control de acceso por radiofrecuencia (RF) en portones residenciales: (a) control de código fijo [3]; (b) control de código evolutivo [4]; (c) receptor RF [5].

El flujo de control tradicional en estos accesos presenta una alta dependencia de la intervención humana [35]. Cuando un vehículo de visita intenta ingresar al recinto, el procedimiento estándar abarca las siguientes etapas secuenciales:

1. **Detención y verificación inicial:** El vehículo se detiene frente a la barrera vehicular. El personal de conserjería interactúa verbalmente con el conductor para indagar el motivo de la visita y la vivienda de destino [37].
2. **Autorización de ingreso:** El conserje intenta comunicarse con el residente mediante citofonía analógica o llamada telefónica interna. Si el residente confirma la autorización, se procede al registro [37].
3. **Registro de información en bitácora:** El conserje anota manualmente en una libreta, libro de actas físico o planilla básica los datos del vehículo (patente, marca, modelo aproximado), la hora

de entrada, el nombre del conductor y el departamento o casa visitada [35, 38].

4. **Acceso y estacionamiento:** Se acciona la apertura manual de la barrera y se indica verbalmente al conductor la ubicación aproximada de los estacionamientos de visitas disponibles [39].

A pesar de ser el modelo operacional más extendido en entornos residenciales, esta arquitectura tradicional presenta vulnerabilidades estructurales y limitaciones técnicas severas en términos de seguridad, trazabilidad y eficiencia operacional (véase en Figura 3.2).



Figura 3.2: Flujo tradicional de ingreso de visitas y principales vulnerabilidades asociadas a cada etapa del proceso.

3.2. Vulnerabilidades de Seguridad del Modelo Tradicional

El esquema convencional de seguridad y control de acceso en condominios y conjuntos residenciales se basa principalmente en la supervisión humana directa y el registro manual de eventos. Si bien este enfoque ha sido históricamente el estándar operativo en la gestión comunitaria, presenta limitaciones estructurales críticas derivadas de la dependencia exclusiva del factor humano, la ausencia de automatización y la falta de interoperabilidad entre los subsistemas de registro y monitoreo. Estas deficiencias generan brechas de seguridad significativas que comprometen tanto la integridad física del recinto como la disponibilidad de información fidedigna ante incidentes. A continuación, se detallan las principales vulnerabilidades identificadas en el modelo tradicional:

1. **Ingeniería Social:** La evaluación del criterio de acceso recae por entero en la percepción subjetiva y el estado de alerta del personal de turno [36]. Esto expone a la comunidad a técnicas de ingeniería social o suplantación de identidad, donde personas ajenas al recinto ejercen persuasión o engaño sobre el personal para obtener el paso sin una verificación formal [37]. Además, en horarios de alto flujo vehicular (horas punta), la presión por evitar la congestión y la formación de cuellos de botella en la vía pública lleva habitualmente a relajar los protocolos de verificación, permitiendo el ingreso de vehículos sin anotar sus antecedentes [38].
2. **Falta de Trazabilidad e Integridad de Datos:** Las bitácoras físicas en papel y los registros manuales presentan serias deficiencias en la preservación de la información [35]. El registro manuscrito es altamente susceptible a errores de transcripción (patentes mal anotadas, horarios inexactos o ilegibles), deterioro físico y pérdida de datos [38]. Ante la ocurrencia de un siniestro, hurto o delito al interior del condominio, la búsqueda retrospectiva de eventos en libros contables físicos resulta lenta e ineficiente, impidiendo realizar cruces de información de manera rápida para auditorías de la administración o investigaciones policiales [37].
3. **Gestión Ineficiente de Estacionamientos de Visitas:** En el esquema tradicional, no existe un mecanismo en tiempo real que informe sobre la ocupación efectiva de las plazas de estacionamiento compartidas [39]. El personal de conserjería desconoce con precisión qué espacios están desocupados sin realizar rondas presenciales, lo que conduce a situaciones en las que se autoriza el ingreso de vehículos que posteriormente no encuentran lugar disponible, generando bloqueos e infracciones en las vías de circulación interna [40]. Asimismo, la ausencia de trazabilidad sobre el tiempo de permanencia facilita el uso indebido de estos espacios por parte de residentes o visitantes no autorizados que sobrepasan los límites temporales permitidos.
4. **Sistemas Pasivos de Videovigilancia (CCTV):** Si bien la mayoría de los condominios cuenta con circuitos cerrados de televisión (CCTV), estos funcionan predominantemente como herramientas de registro pasivo [41]. Las cámaras graban de forma continua, pero no procesan la información en tiempo real; por lo tanto, no emiten alertas automáticas ante la presencia de patentes no autorizadas, vehículos sospechosos o sobreocupación de espacios [42]. Su utilidad se limita al análisis forense posterior a un incidente, cuando el acceso no autorizado o el delito ya se ha concretado.

3.3. De Edificios Residenciales a Comunidades Inteligentes

La rápida urbanización y el incremento en las exigencias de seguridad y eficiencia operacional han impulsado la evolución de los entornos habitacionales tradicionales hacia el paradigma de los edificios inteligentes (*Smart Buildings*) y las comunidades inteligentes (*Smart Communities*) [43, 44]. Este modelo se fundamenta en el concepto amplio de *Smart City*, definido como un ecosistema urbano sostenible que utiliza las Tecnologías de la Información y Comunicación (TIC) para optimizar la gestión operativa, mejorar la infraestructura física y elevar la calidad de vida de sus habitantes [43]. A escala microbarrial, esta transformación se sustenta en la convergencia de tecnologías emergentes como la Inteligencia Artificial (IA), la Visión por Computador (CV), el Internet de las Cosas (IoT) y el procesamiento de datos en la nube (*Cloud Computing*) [45, 46].

En los entornos residenciales tradicionales, los subsistemas de seguridad, control de acceso, videovigilancia y gestión de estacionamientos operan de manera aislada y fragmentada, conformando sistemas independientes [44]. En contraste, una comunidad inteligente integra estos módulos bajo una arquitectura unificada que permite la captura, transmisión, procesamiento y análisis de datos en tiempo real [43, 45]. La transición desde el modelo tradicional hacia un modelo inteligente se sustenta en dos pilares tecnológicos principales, respaldados por evidencia empírica.

3.3.1. Internet de las Cosas (IoT) e Infraestructura Sensorial

El despliegue de redes de sensores e infraestructura de bajo costo permite dotar de capacidad de cómputo y conectividad a elementos físicos del entorno residencial [45]. En el ámbito de los estacionamientos y accesos, la adopción de microcontroladores y sensores ambientales (ultrasónicos, magnéticos o de presencia) facilita la digitalización del estado del entorno en tiempo real [43, 40].

Un estudio empírico recientemente desarrollado [47] propone e implementa un sistema dinámico de estacionamiento e infraestructura de acceso automatizado mediante una arquitectura IoT de bajo costo, estructurada en tres capas funcionales. La primera, denominada capa de sensado, comprende una red de sensores infrarrojos (IR) dispuestos en cada plaza de estacionamiento y en los puntos de entrada y salida del recinto. La segunda, la capa de procesamiento, está compuesta por un microcontrolador Arduino Uno que actúa como unidad central de decisión, recibiendo las señales de los sensores, ejecutando un algoritmo dinámico de asignación de espacios y accionando servomotores para el control automatizado de las barreras de acceso. La tercera, la capa de interfaz de usuario, se materializa en una pantalla LCD 20x4 I2C que informa en tiempo real la disponibilidad de plazas, permitiendo al conductor conocer el estado del recinto antes de ingresar.

El flujo operativo del sistema elimina por completo la intervención manual: cuando un vehículo es detectado en el punto de entrada, el sistema evalúa automáticamente la disponibilidad de plazas y, de existir espacio libre, acciona la barrera para permitir el ingreso; en caso contrario, informa la saturación del recinto mediante la pantalla. De manera análoga, la salida de un vehículo es detectada por el sensor de egreso, lo que actualiza de forma inmediata el contador de disponibilidad (véase en Figura 3.3). Este ciclo de detección, procesamiento y acción reduce la congestión vehicular en el acceso y evita la circulación innecesaria al interior del recinto [47].

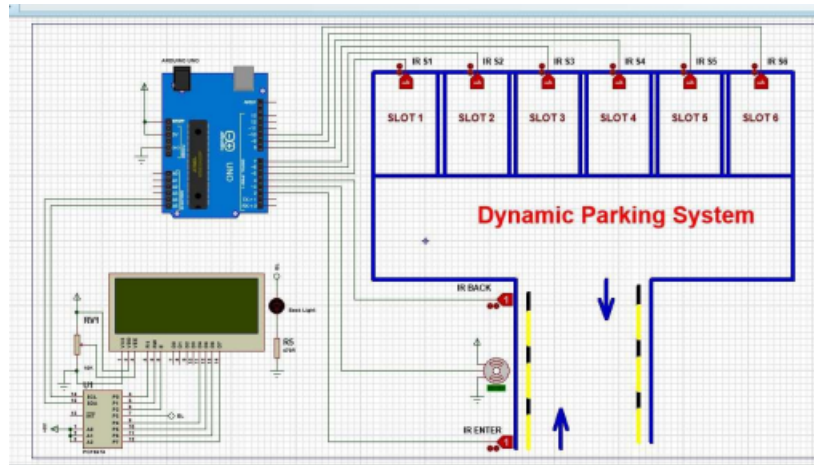


Figura 3.3: Diagrama de circuito sistema de aparcamiento dinámico, que muestra la integración de Arduino Uno, sensores infrarrojos para la detección de plazas, una pantalla LCD para visualización en tiempo real y un servomotor para el control automatizado de la barrera.

Los resultados experimentales validados en un entorno simulado confirmaron la fiabilidad y precisión del sistema en la detección de ocupación y el control de barreras (véase en Figura 3.4). No obstante, los propios autores reconocen limitaciones relevantes en su implementación actual: la ausencia de conectividad en la nube impide el monitoreo remoto del estado del estacionamiento, la falta de una interfaz móvil limita el acceso a la información por parte de los residentes fuera del recinto, y la arquitectura no contempla mecanismos de reserva anticipada ni integración con sistemas de control de acceso basados en identificación vehicular [47].

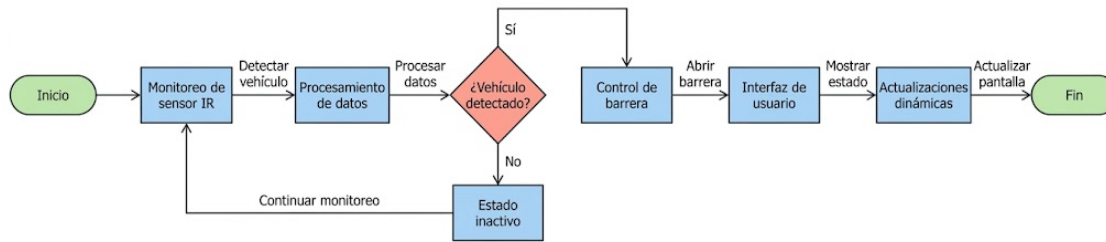


Figura 3.4: Diagrama de flujo del sistema de detección y monitoreo de vehículos mediante sensores infrarrojos, que ilustra el proceso desde la activación del sensor hasta la actualización dinámica del estado y la visualización.

3.3.2. Analítica de Datos y Visión por Computador

Las cámaras pasivas del circuito cerrado de televisión (CCTV) evolucionan hacia sensores analíticos avanzados mediante la integración de algoritmos de aprendizaje profundo (*Deep Learning*) [42, 46]. En lugar de transmitir grandes volúmenes de video para almacenamiento pasivo o inspección manual humana, los sistemas modernos procesan los flujos de video localmente en dispositivos de cómputo en el borde (*Edge Computing*) o servidores dedicados, permitiendo tomar decisiones de acceso en tiempo real sin depender de infraestructura centralizada [42].

Un primer caso empírico que ilustra la evolución hacia interfaces digitales de gestión remota es el sistema desarrollado en [48] (véase en Figura 3.5), que implementa un sistema de gestión de visitas para edificios residenciales que elimina la dependencia del personal de recepción mediante hardware embebido y una aplicación móvil [49, 50]. El sistema captura imágenes del visitante y su documento de identidad mediante módulos ESP32-CAM [51], almacena la información en Firebase Cloud y notifica al residente en tiempo real para que autorice o deniegue el acceso de forma remota. Una vez autorizado, el sistema emite automáticamente una tarjeta RFID temporal [52] mediante un Arduino UNO [53] y un servomotor SG90 [54], permitiendo al visitante entrar y salir durante el período acordado sin registrarse nuevamente. Si bien el sistema representa un avance significativo respecto al modelo de recepción tradicional, las cámaras operan únicamente como herramientas de captura pasiva: la decisión de autorizar el acceso sigue recayendo en el juicio humano del residente, y el sistema no contempla ningún mecanismo de identificación vehicular automática ni gestión de estacionamientos [48].

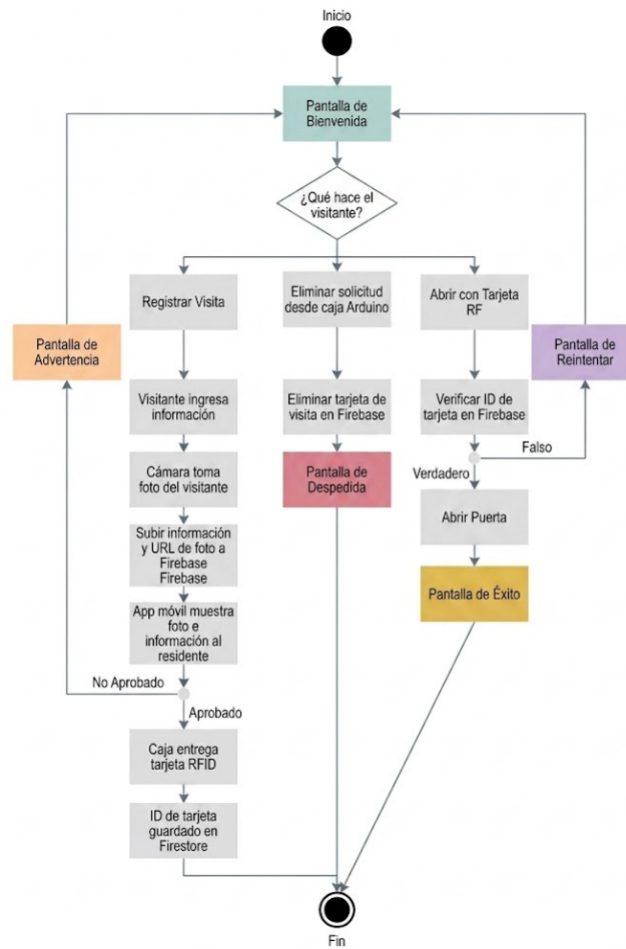


Figura 3.5: Función del flujo principal del sistema.

Esta limitación es precisamente la que abordan los sistemas de visión por computador aplicados al control de acceso vehicular. En [55] se propone una arquitectura de reconocimiento automático de patentes compuesta por tres etapas secuenciales:

- **Detección placa vehicular:** La imagen de entrada en formato RGB es convertida a escala de grises para reducir la carga computacional, tras lo cual se aplica un suavizado gaussiano con kernel 5×5 para eliminar ruido y detalles irrelevantes. A continuación, se emplea umbralización adaptativa [56, 57] para gestionar las variaciones de iluminación entre regiones de la imagen, obteniéndose una imagen binaria sobre la que se extraen los 30 contornos de mayor área y perímetro. De entre estos, se seleccionan únicamente los contornos de cuatro vértices, dado que la patente vehicular posee forma rectangular, utilizando un método de aproximación de contornos para su localización precisa [58] (véase en Figura 3.6).

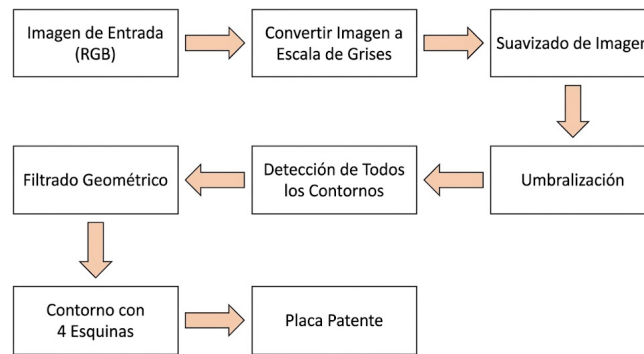


Figura 3.6: Diagrama de flujo del proceso de detección de placa vehicular: desde la imagen RGB de entrada hasta la localización del contorno rectangular de cuatro vértices que delimita la placa.

- **Segmentación de caracteres:** Este paso es crítico para el sistema, ya que cualquier error en la segmentación se propaga directamente a la etapa de reconocimiento [59]. El proceso incluye operaciones morfológicas e histograma de proyección para aislar cada carácter individualmente, generando imágenes de entrada independientes para la etapa clasificadora.
- **Reconocimiento de caracteres:** Implementada mediante una red neuronal convolucional (CNN) desarrollada en TensorFlow [60]. La CNN (véase en Figura 3.7) emplea tres capas convolucionales con kernel 5×5 , submuestreo por *max pooling* con kernel 2×2 , una capa completamente conectada con *dropout* de factor 0,5 para prevenir sobreajuste, y una capa *softmax* final que clasifica 36 clases (10 dígitos y 26 letras). El modelo fue entrenado sobre un conjunto de 1.620 imágenes de caracteres extraídas de su dataset propio [61] y validado con 360 imágenes adicionales, alcanzando una precisión de reconocimiento superior al 95,75 % sobre el conjunto de prueba [55].

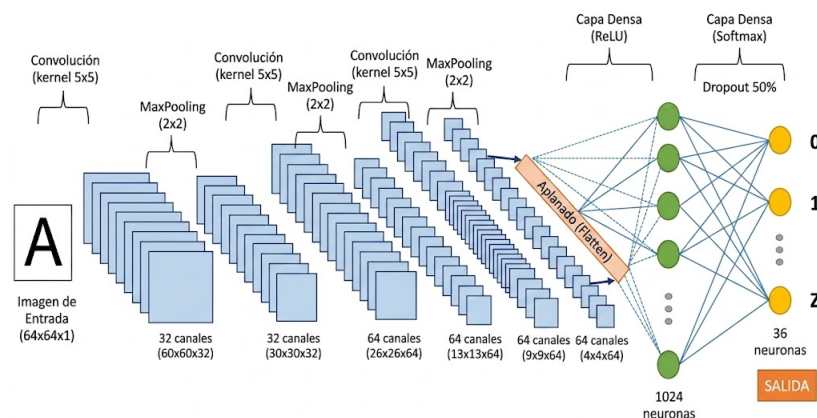


Figura 3.7: Arquitectura de la red neuronal convolucional (CNN) empleada para el reconocimiento de caracteres de patentes vehiculares. La red se compone de tres capas convolucionales y una capa *softmax* de salida con 36 clases (10 dígitos y 26 letras).

El estudio fue implementado y evaluado en Google Colab utilizando TensorFlow y OpenCV, evidenciando que arquitecturas de este tipo son ejecutables en entornos de cómputo de bajo costo, compatibles con el paradigma de procesamiento en el borde (véase en Figura 3.8).

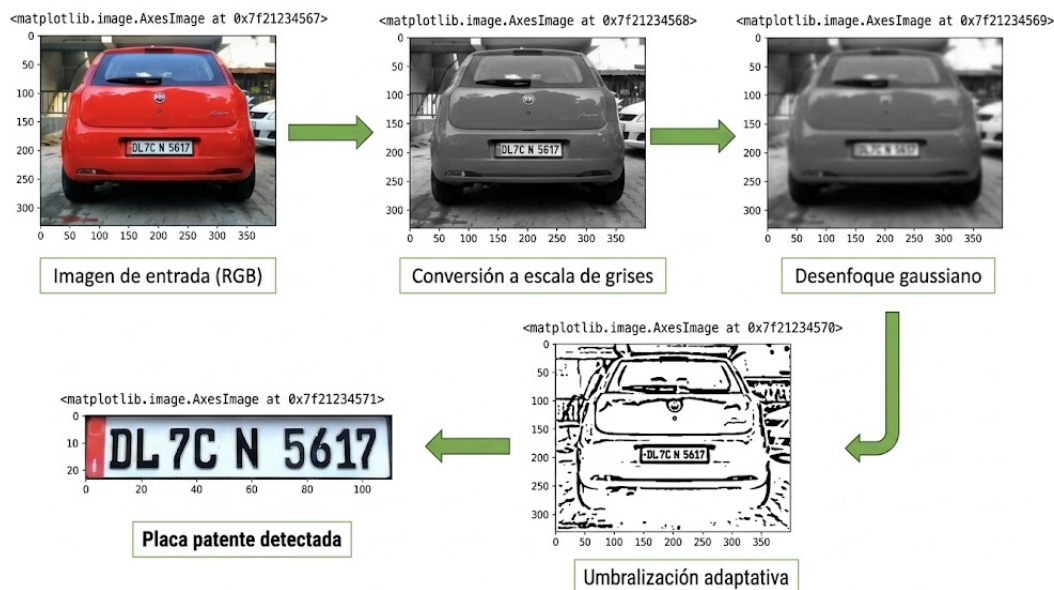


Figura 3.8: Resultado del proceso de detección de placa vehicular sobre una imagen real: (a) imagen de entrada en formato RGB, (b) conversión a escala de grises, (c) aplicación de suavizado gaussiano, (d) umbralización adaptativa y (e) placa vehicular localizada y delimitada.

No obstante, los propios autores identifican limitaciones relevantes: el sistema no considera el efecto del desenfoque por movimiento (*motion blur*), no aborda la detección de patentes con ángulo de inclinación [62], y fue evaluado únicamente sobre datasets estáticos sin validación en escenarios de video en tiempo real [55].

Las patentes son claves para la identificación de los usuarios; sin embargo, no son suficientes de manera aislada. Para lograr una gestión vehicular integral en tiempo real, resulta indispensable detectar y clasificar el vehículo completo en el flujo de video. En esta línea, en [63] se propone un marco avanzado de detección vehicular holística basado en la arquitectura de aprendizaje profundo YOLOv11, orientado a entornos dinámicos de tráfico y sistemas de asistencia a la conducción.

La arquitectura de YOLOv11, ilustrada en la Figura 3.9, puede entenderse como una cadena de tres etapas funcionales. La primera, denominada *Backbone*, es la encargada de analizar la imagen de entrada y extraer sus características visuales relevantes, desde bordes y texturas simples hasta formas complejas como ruedas, ventanas o carrocerías. La segunda etapa, el *Neck*, actúa como un integrador: combina la información extraída a distintas escalas de detalle, lo que permite al modelo identificar tanto vehículos grandes que ocupan gran parte del encuadre como objetos más pequeños o parcialmente

ocultos. Finalmente, la etapa *Head* es la responsable de producir las predicciones finales, determinando para cada región de la imagen si contiene un vehículo, de qué tipo es y con qué nivel de confianza se realiza esa clasificación [63].

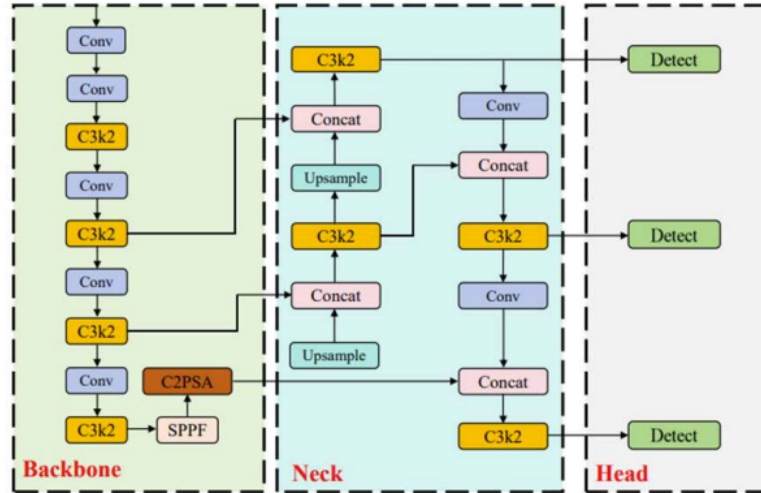


Figura 3.9: Arquitectura general de la red YOLOv11.

El modelo fue entrenado y validado sobre un conjunto de datos de 30.000 imágenes etiquetadas que abarca seis categorías vehiculares (*automóvil*, *autobús*, *camión*, *motocicleta*, *bicicleta* y *patinete eléctrico*).

En términos de rendimiento cuantitativo, la propuesta alcanzó resultados sobresalientes: una precisión del 97,2 %, una sensibilidad (*recall*) del 93,4 %, un $mAP@50$ del 97,0 % y un $mAP@50-95$ del 86,6 %. Este sólido desempeño y el equilibrio entre las métricas se evidencian detalladamente en la curva de confianza F1 (Figura 3.10a) y en la curva de precisión-confianza (Figura 3.10b).

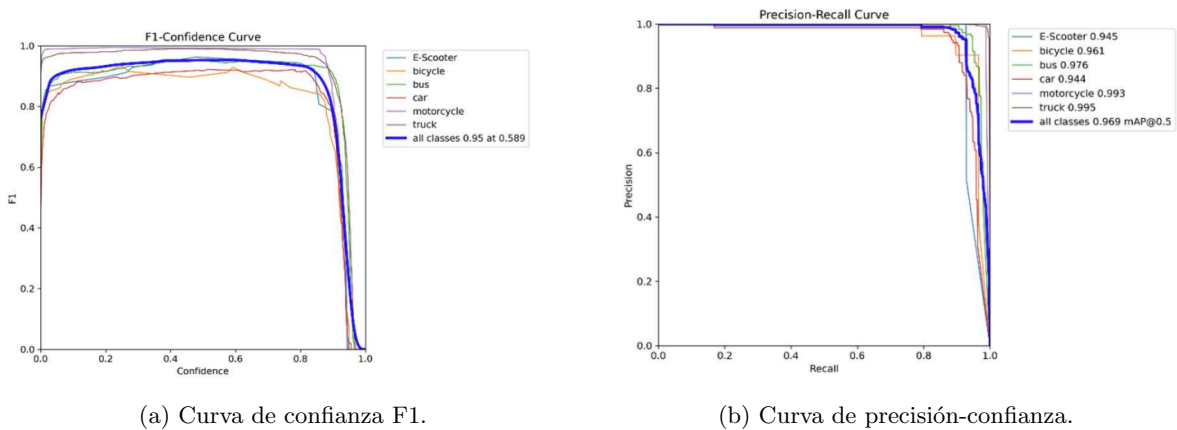


Figura 3.10: Métricas de evaluación del modelo YOLOv11 para detección vehicular en tiempo real.

Asimismo, el sistema demostró una alta eficiencia computacional, logrando una velocidad promedio de inferencia de 56 fotogramas por segundo (FPS) equivalente a 17,8 ms por cuadro procesando imágenes de 640×640 píxeles. Como se detalla en la Tabla 3.1, YOLOv11 supera cuantitativamente en exactitud a arquitecturas previas ampliamente utilizadas en el estado del arte.

Tabla 3.1: Comparación del rendimiento de modelos de detección de última generación.

Model	Size	mAP@50
Faster RCNN	640×640	89.06 %
EfficientDet	640×640	89.4 %
YOLOv5	640×640	85.8 %
YOLO-NAS	640×640	85 %
YOLOv8	640×640	92.6 %
YOLOv9	640×640	92.7 %
YOLOv11	640×640	97.0 %

Las detecciones en escenarios de prueba (véase en Figura 3.11) reafirman la robustez de este enfoque, logrando identificar de manera precisa múltiples clases vehiculares simultáneamente con altos márgenes de certidumbre.



Figura 3.11: Resultados visuales de detección obtenidos con la red YOLOv11.

En conclusión, este trabajo demuestra la viabilidad de ejecutar detección multiclase de vehículos en tiempo real con alta precisión y baja latencia, superando las vulnerabilidades ambientales de los métodos tradicionales de sustracción de fondo. Adicionalmente, consolida a YOLOv11 como una opción estandarizada y óptima para su despliegue en nodos de cómputo en el borde (*Edge Computing*).

3.4. Control de Acceso y Seguridad en Comunidades Inteligentes

La incorporación de tecnologías IoT en infraestructuras residenciales ha transformado profundamente la forma en que se gestiona el control de acceso, desplazando progresivamente los esquemas manuales hacia arquitecturas digitales capaces de automatizar la verificación, el registro y la trazabilidad de los flujos de personas y vehículos [35, 36]. Esta transición, sin embargo, no está exenta de desafíos: la heterogeneidad de dispositivos, la exposición de interfaces de comunicación y las limitaciones computacionales de los nodos de borde introducen nuevas vulnerabilidades que deben ser consideradas desde el diseño del sistema [38].

En el modelo tradicional, la autorización de acceso depende exclusivamente del criterio subjetivo del personal de conserjería, lo que introduce riesgos concretos de ingeniería social, errores de identificación y ausencia de trazabilidad [36, 37]. La transición hacia sistemas digitales de control de acceso busca precisamente eliminar estas vulnerabilidades mediante la automatización del proceso de verificación y el registro sistemático de cada evento de acceso [35].

Desde una perspectiva de diseño, los sistemas modernos de control de acceso en comunidades residenciales inteligentes (véase en Figura 3.12) se articulan en torno a tres componentes funcionales interdependientes: (i) un módulo de identificación, encargado de verificar la identidad del visitante o vehículo; (ii) un módulo de autorización, que contrasta la identidad verificada contra una base de datos de accesos permitidos; y (iii) un módulo de actuación, que ejecuta la apertura o denegación del acceso y registra el evento con su marca temporal [37, 38]. La robustez del sistema depende de la fiabilidad de cada uno de estos componentes y de la integridad de la comunicación entre ellos.

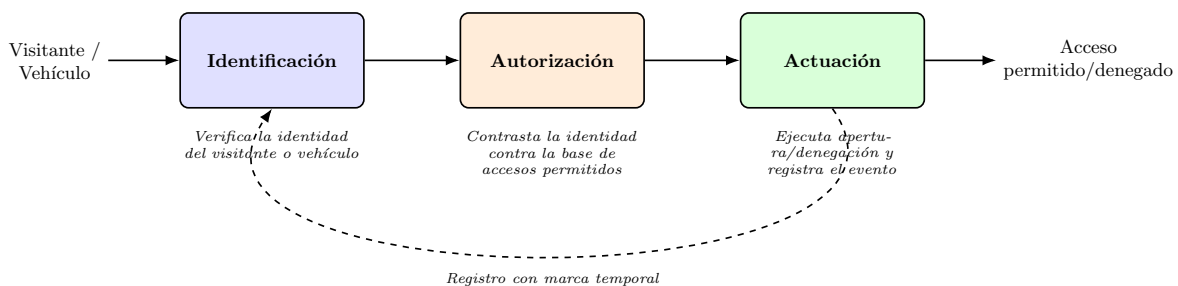


Figura 3.12: Componentes funcionales de un sistema de control de acceso en comunidades residenciales inteligentes: identificación, autorización y actuación.

Es en este contexto donde las tecnologías de visión por computador y sensores IoT adquieren un rol central, al proveer los mecanismos de identificación automática que los sistemas tradicionales no pueden ofrecer. Las siguientes secciones analizan en detalle las soluciones tecnológicas disponibles para cada uno de los subproblemas identificados: el reconocimiento automático de patentes vehiculares y la gestión de estacionamientos mediante sensores IoT.

3.4.1. Reconocimiento Automático de Patentes ALPR

Los sistemas ALPR (del inglés *Automatic License Plate Recognition*) constituyen el núcleo tecnológico del control de acceso vehicular automatizado. Su evolución ha seguido una trayectoria progresiva desde técnicas clásicas de procesamiento de imágenes hacia arquitecturas basadas en aprendizaje profundo, impulsada por la necesidad de operar con precisión en condiciones reales de variabilidad ambiental [42].

3.4.1.1. Enfoques Tradicionales

Los métodos clásicos de ALPR se fundamentan en una secuencia de etapas de preprocesamiento, que incluye la conversión a escala de grises, el suavizado y la umbralización adaptativa, seguidas de la localización de la placa mediante detección de bordes, contornos o líneas, y un reconocimiento final apoyado en clasificadores estadísticos, de correlación o plantillas [56, 57, 58]. El sistema presentado en [55], ejemplifica un caso híbrido dentro de este paradigma: si bien sus etapas de detección de placa y segmentación de caracteres se apoyan en técnicas clásicas de procesamiento de imágenes, el reconocimiento final se resuelve mediante una CNN, alcanzando una precisión del 95,75 % sobre conjuntos de datos estáticos, aunque sin abordar el desenfoque por movimiento ni la inclinación de la placa.

Una variante metodológica distinta es la propuesta en [64], quienes localizan la placa mediante la transformada de Hough para la detección de líneas rectas, seguida de reconocimiento de caracteres por *template matching* sobre patentes de formato estandarizado. Si bien este enfoque demuestra robustez para localizar la geometría rectangular de la placa bajo condiciones controladas, su dependencia de plantillas predefinidas limita su generalización a formatos de patente no contemplados en el conjunto de plantillas. En ambos sistemas, no se evalúa el desempeño ante desenfoque por movimiento ni ante ángulos de captura oblicuos.

De manera complementaria, en [65] se propone un sistema ALPR optimizado para dispositivos de bajo costo como Orange Pi, basado en visión por computadora tradicional con OpenCV, logrando

una tasa de reconocimiento del 96 % pero con limitaciones ante condiciones adversas de iluminación y perspectiva.

En conjunto, estos enfoques evidencian una limitación estructural compartida por el paradigma clásico: los mecanismos para abordar directamente el desenfoque por movimiento y la inclinación de la placa, factores inevitables en escenarios de video en tiempo real, no forman parte del repertorio de técnicas de procesamiento de imagen tradicional. Frente a esta limitación, la literatura reciente recurre de manera consistente a arquitecturas de aprendizaje profundo, ya sea mediante redes de deconvolución para corregir el desenfoque o redes de transformación espacial (*Spatial Transformer Networks*) para corregir la inclinación de la placa antes del reconocimiento, lo que constituye el punto de inflexión que motiva la adopción de este nuevo paradigma respecto al tradicional.

3.4.1.2. Enfoques Basados en Aprendizaje Profundo

La incorporación de redes neuronales convolucionales (CNN) representa un salto cualitativo en la robustez de los sistemas ALPR [60]. La introducción de arquitecturas de detección de una sola etapa como YOLO permitió unificar la localización y clasificación de objetos en un único proceso, reduciendo significativamente la latencia de inferencia y habilitando el procesamiento en tiempo real [66]. Estudios como el presentado en [59] confirmaron además que los enfoques basados en aprendizaje profundo eliminan la necesidad de segmentación manual de caracteres, simplificando el pipeline completo. Entre los antecedentes directos de esta generación de sistemas se encuentran la propuesta de [67], que integra YOLOv5 con un mecanismo de atención de canal y una etapa de reconocimiento basada en GRU con función de pérdida CTC, y el sistema presentado en [68], cuyo enfoque de dos etapas de detección y clasificación del layout de la placa alcanzó una tasa de reconocimiento end-to-end del 96,9 % evaluada sobre ocho conjuntos de datos públicos de cinco regiones distintas, superando tanto a trabajos previos como a sistemas comerciales. No obstante, ambos enfoques dependen de infraestructura de cómputo con recursos considerables (servidor o nube), lo que limita su despliegue directo en dispositivos de borde de bajo costo.

Un caso representativo de esta generación es VehicleTracker, propuesto en [69], un sistema ALPR end-to-end de despliegue nativo en el borde (*edge-native*) que integra YOLOv5 para la detección de placas con un motor OCR híbrido (EasyOCR y Tesseract) para el reconocimiento de caracteres. El sistema (véase en Figura 3.13) procesa cada fotograma mediante un pipeline de seis etapas: (i) captura y gestión de sesión, (ii) preprocesamiento de imagen, (iii) detección de placa mediante YOLOv5, (iv) postprocesamiento y deduplicación mediante distancia de Levenshtein, (v) almacenamiento en base de datos MongoDB, y (vi) visualización de resultados en un dashboard web. Los resultados experimentales

reportan una precisión de reconocimiento del 88,9 %, una tasa de falsos positivos del 2,1 % y una latencia de procesamiento de aproximadamente 35 ms por fotograma; según reportan los propios autores, esta cifra supera a sistemas basados en pipelines multi-etapa, que alcanzarían latencias de hasta 140 ms [69].

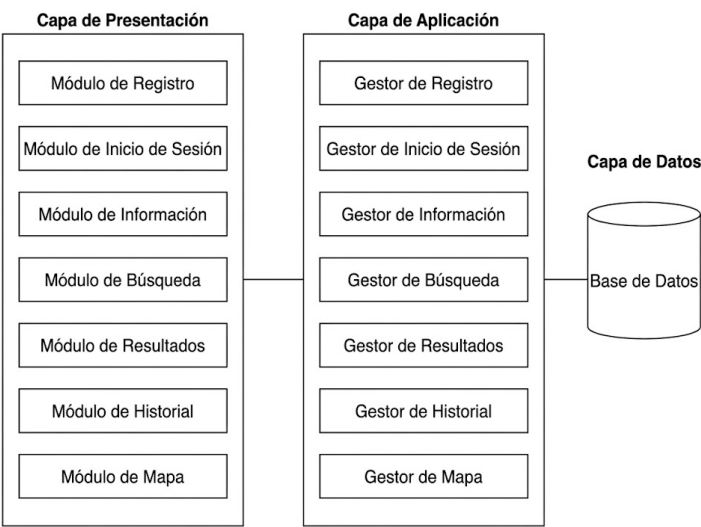


Figura 3.13: Arquitectura integral del sistema VehicleTracker, ilustrando la separación en capas de presentación, aplicación y datos.

Un aspecto arquitectónico relevante de VehicleTracker es su capacidad de operar sin conectividad a internet, procesando los flujos de video localmente mediante un diseño edge-native que garantiza privacidad de datos y continuidad operacional ante fallos de red [69]. Esta característica es especialmente pertinente para aplicaciones de control de acceso residencial, donde la dependencia de servicios en la nube puede introducir latencias inaceptables o puntos únicos de fallo. No obstante, el sistema fue evaluado sobre videos pregrabados y no contempla la integración con sistemas de gestión de acceso, reservas de estacionamiento ni notificaciones a usuarios.

En síntesis, la evolución de los sistemas ALPR evidencia un desplazamiento desde enfoques secuenciales basados en reglas hacia arquitecturas de aprendizaje profundo integradas, capaces de operar en tiempo real sobre dispositivos de cómputo de bajo costo. Para consolidar los avances recientes en esta área, la Tabla 3.2 presenta una comparativa de los principales enfoques analizados, destacando sus diferencias clave en diseño, latencia y modos de despliegue.

Tabla 3.2: Resumen comparativo de enfoques ALPR modernos basados en aprendizaje profundo.

Arquitectura Base	Alcance del Sistema	Capacidad de Rastreo	Latencia Media	Modo de Despliegue	Referencia
YOLOv5 + GRU	Solo Algorítmico	No reportada	No reportada	Nube / Servidor	[67]
YOLO Multi-etapa	Pipeline Algorítmico	No reportada	No reportada	Servidor (Alto Cómputo)	[68]
YOLOv5 + OCR Híbrido	Aplicación End-to-End	Sesión Multipunto	~35 ms	Nativo en Borde (Offline)	[69]

3.4.2. Gestión de Estacionamientos: Visión por Computador vs. Sensores IoT

La administración eficaz de los estacionamientos de visita requiere monitorear la ocupación, controlar los tiempos de uso y facilitar la gestión de reservas. Para abordar estos requerimientos, la literatura identifica dos enfoques tecnológicos principales: sistemas basados en visión por computador y sistemas basados en sensores IoT dedicados, cada uno con ventajas y limitaciones específicas según el entorno de despliegue.

3.4.2.1. Sistemas Basados en Visión por Computador

Los sistemas de monitoreo de estacionamientos basados en cámaras aprovechan la infraestructura de videovigilancia existente para inferir el estado de ocupación de cada plaza mediante algoritmos de detección de objetos [66]. Este enfoque presenta ventajas económicas en espacios abiertos con buenas condiciones de iluminación, donde una sola cámara puede cubrir múltiples plazas simultáneamente, reduciendo el costo de hardware por espacio monitorizado [40].

Un caso representativo de este paradigma, que además ilustra su complementariedad con sensores IoT en lugar de una sustitución total, es el sistema propuesto en [66]. Su arquitectura combina un microcontrolador NodeMCU con sensores ultrasónicos de proximidad para la detección primaria de vehículos en cada plaza, mientras que las cámaras CCTV existentes en el recinto cumplen un rol de verificación secundaria: el flujo de video se emplea específicamente para contrastar y validar las lecturas entregadas por los sensores IoT, eliminando falsos positivos que estos últimos podrían generar de forma aislada. Adicionalmente, el sistema incorpora bolardos de seguridad automatizados para reforzar la protección física del vehículo estacionado, y una aplicación móvil que permite a los conductores consultar la disponibilidad de espacios de forma remota. Este diseño evidencia que, en la práctica, visión por computador y sensores IoT no son necesariamente paradigmas excluyentes entre sí: la cámara aporta una capa de verificación semántica que compensa la limitación estructural de los sensores de proximidad, que no distinguen vehículos de otros objetos, mientras que el sensor IoT ofrece la detección primaria de bajo costo computacional que la videovigilancia continua no puede igualar [66].

Sin embargo, el enfoque basado predominantemente en visión por computador presenta limitaciones significativas en entornos complejos. Las oclusiones producidas por pilares estructurales y la variabilidad de sombras en estacionamientos subterráneos degradan considerablemente la precisión de detección. Incluso en sistemas híbridos CV-IoT recientes, los propios autores reconocen que en entornos de baja iluminación es necesario recurrir a algoritmos complementarios de detección nocturna

para mantener la fiabilidad del reconocimiento visual [66]. Adicionalmente, el procesamiento continuo de flujos de video demanda recursos computacionales elevados, lo que puede resultar prohibitivo en despliegues de bajo costo [40]. Estas limitaciones se acentúan precisamente en el tipo de entorno más común en comunidades residenciales de acceso cerrado en Chile: estacionamientos subterráneos con conectividad limitada y condiciones de iluminación no deseadas.

3.4.2.2. Sistemas Basados en Sensores IoT

En contraste, los sistemas basados en sensores IoT emplean tecnologías de detección no ópticas, como sensores ultrasónicos, infrarrojos o magnéticos, cuyo principio de funcionamiento no depende de la iluminación ambiental ni de la complejidad visual de la escena. Estos sistemas despliegan un sensor dedicado por cada plaza de estacionamiento [39], lo que permite una detección binaria (ocupado/libre) de alta fiabilidad con bajo consumo energético y mínimos requerimientos de procesamiento centralizado.

El sistema SPARK, propuesto en [70], representa uno de los casos de referencia más citados en esta línea. Si bien corresponde a un trabajo fundacional que antecede en más de una década a la consolidación del paradigma IoT tal como se entiende actualmente, se incluye como antecedente seminal de la arquitectura WSN aplicada a estacionamientos, y no como representante del estado del arte más reciente. Su arquitectura basada en redes de sensores inalámbricos (WSN) permite la monitorización remota de espacios, la guía automatizada hacia plazas disponibles y mecanismos de reserva, sin necesidad de excavaciones ni instalaciones de cableado costosas (véase en Figura 3.14). Los nodos de la WSN integran unidades de detección, cómputo y comunicación que cooperan para recopilar, procesar y transmitir el estado de ocupación de cada espacio en tiempo real [70].

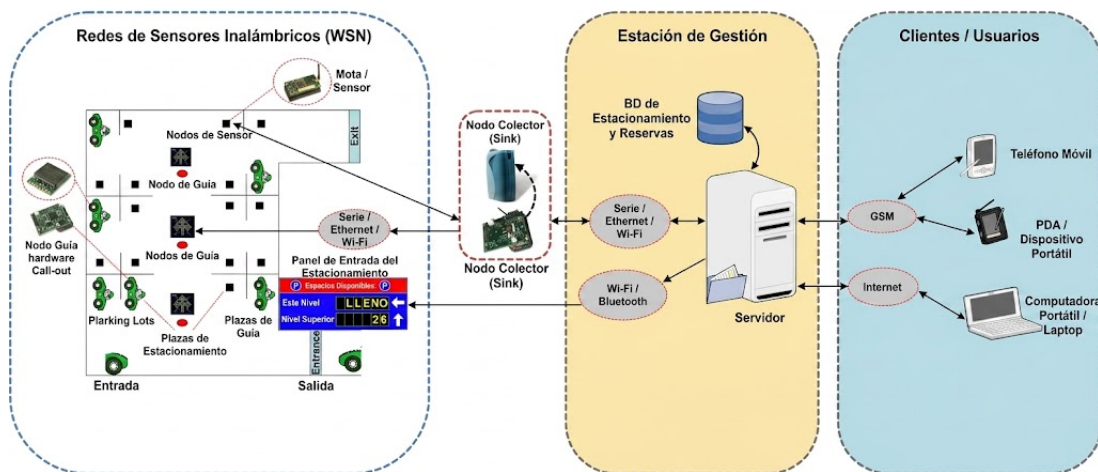


Figura 3.14: Arquitectura general del sistema SPARK, compuesta por seis subsistemas interconectados: WSN, Sink, Gestión de Estacionamiento, Guía Automatizada, Pantalla de Entrada y Cliente.

Desde una perspectiva arquitectónica más reciente, los trabajos de [6] y [71] describen que los sistemas de estacionamiento inteligente basados en WSN e IoT se estructuran en cuatro capas funcionales interconectadas (véase en Figura 3.15). La primera es la Capa de Detección o Sensores (*Sensing Layer/Perception Layer*), responsable de la recopilación de datos brutos mediante sensores desplegados en los espacios de estacionamiento para detectar la presencia de vehículos. Estos sensores de carretera pueden ser instalados dentro, sobre o al lado de la superficie de la vía [6]. La segunda es la Capa de Red (*Networking Layer/Communication Layer*), que gestiona la transmisión de los datos recopilados por los sensores a otras partes del sistema, utilizando infraestructuras de comunicación robustas; las WSN facilitan en particular la comunicación vehículo-infraestructura y los sistemas de localización en tiempo real [71]. La Capa de Middleware se encarga del procesamiento, la gestión y el análisis de los datos brutos de los sensores, empleando algoritmos para predecir la disponibilidad de estacionamiento, optimizar el uso del espacio y preparar los datos para la capa de aplicación [6]. Finalmente, la Capa de Aplicación (*Application Layer*) utiliza los datos procesados para presentar información útil a los conductores a través de aplicaciones móviles, interfaces web u otros medios, permitiendo a los usuarios verificar la disponibilidad de estacionamientos en tiempo real, realizar reservas y recibir orientación hacia los espacios disponibles [6, 71].

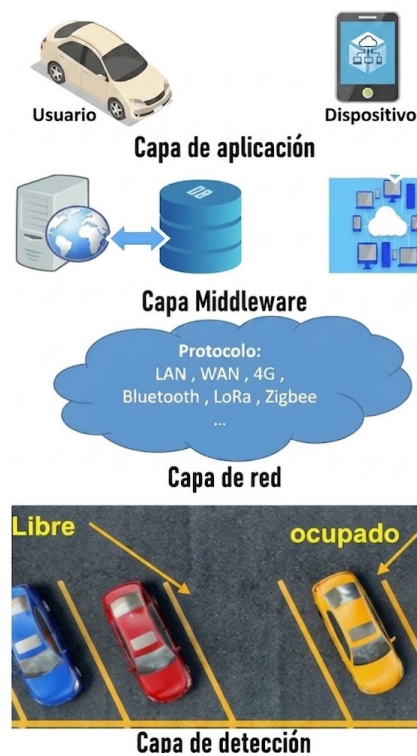


Figura 3.15: Capas funcionales de un sistema de estacionamiento inteligente basado en WSN e IoT: Capa de Detección o Sensores, Capa de Red, Capa de Middleware y Capa de Aplicación [6].

La integración de la tecnología IoT con WSN en sistemas de estacionamiento inteligente permite la detección de la presencia de vehículos y ofrece funcionalidades avanzadas como la monitorización remota y la gestión de reservas [71]. Este enfoque proporciona una solución automatizada, en tiempo real, para la gestión de estacionamientos con una robustez ante condiciones ambientales que la visión por computador no logra igualar en entornos de iluminación heterogénea u oclusión estructural.

No obstante, esta robustez tiene como contrapartida una escalabilidad de costos distinta a la de los sistemas basados en cámaras: mientras que una sola cámara puede cubrir múltiples plazas de estacionamiento simultáneamente, los sistemas IoT requieren, por diseño, un sensor dedicado por cada espacio a monitorizar [40]. Esto implica que el costo de hardware e instalación de un sistema IoT crece de forma prácticamente lineal con el número de plazas del recinto, a diferencia del costo marginal decreciente que ofrece una arquitectura basada en cámaras a medida que aumenta la superficie cubierta. En consecuencia, la elección entre ambos paradigmas no admite una respuesta única, sino que depende de las condiciones específicas del entorno de despliegue, tamaño del estacionamiento, condiciones de iluminación y presupuesto disponible, lo que motiva la evaluación de ambos enfoques en el contexto particular de las comunidades residenciales de acceso cerrado. La Tabla 3.3 sintetiza los principales criterios de comparación entre ambos paradigmas a partir de la evidencia revisada.

Tabla 3.3: Comparación entre sistemas de gestión de estacionamientos basados en visión por computador y en sensores IoT.

Criterio	Visión por Computador	Sensores IoT
Escalabilidad de costo	Marginal decreciente: una cámara cubre múltiples plazas [40]	Lineal: requiere un sensor dedicado por plaza [40]
Robustez ante iluminación/occlusión	Baja en entornos subterráneos: sombras, pilares y baja iluminación degradan la precisión, requiriendo algoritmos complementarios [66]	Alta: sensores no ópticos, independientes de condiciones visuales del entorno
Demanda computacional	Elevada: procesamiento continuo de flujos de video [40]	Baja: detección binaria por nodo, procesamiento distribuido [70]
Instalación	Reutiliza infraestructura de videovigilancia existente [66]	Sin excavaciones, pero requiere despliegue de un nodo por plaza [70]
Caso representativo	Dixit et al. [66]: CV + IoT híbrido para verificación de falsos positivos	Srikanth et al. [70] (SPARK): WSN con guía y reserva

3.5. Soluciones Comerciales de Gestión Residencial

Habiendo caracterizado los componentes funcionales que debe cubrir un sistema de control de acceso moderno de identificación, autorización y actuación (véase en Figura 3.12), corresponde ahora

examinar en qué medida las soluciones disponibles comercialmente satisfacen dicha arquitectura. Este análisis resulta relevante tanto para dimensionar el estado de adopción tecnológica en el mercado nacional como para identificar brechas funcionales que justifiquen el desarrollo de una solución propia.

En el mercado chileno coexisten dos categorías de soluciones tecnológicas orientadas a comunidades residenciales, que difieren sustancialmente en su alcance funcional. La primera corresponde a plataformas de gestión administrativa, cuyo propósito principal es digitalizar procesos de cobro, comunicación y reserva de espacios comunes, sin abordar directamente la seguridad física del acceso vehicular. La segunda corresponde a sistemas de control de acceso físico, que sí incorporan hardware especializado, lectores de patente, barreras, y tarjetas de proximidad para automatizar el ingreso de personas y vehículos. A estas dos categorías se suman referentes internacionales especializados específicamente en reconocimiento de patentes vehiculares (ALPR), cuyo análisis permite contrastar el estado de la técnica local con soluciones de mercados con mayor madurez en esta tecnología específica.

3.5.1. Plataformas de gestión administrativa

La primera categoría agrupa plataformas orientadas principalmente a la gestión administrativa de condominios, cuyo módulo de control de acceso cuando existe se ofrece como una funcionalidad adicional y no como el eje central de la solución. Se destacan las siguientes plataformas:

1. **Edifito:** Es un software 100 % en línea orientado a administradores que gestionan múltiples comunidades desde un panel centralizado. Sus funcionalidades principales incluyen la automatización de gastos comunes, la comunicación con copropietarios y, en su versión más reciente, la incorporación de inteligencia artificial para automatizar tareas administrativas y responder consultas vía WhatsApp. No ofrece un módulo de control de acceso físico propio [72].
2. **Comunidad Feliz:** Según cifras propias del proveedor, la aplicación de administración de condominios más descargada de Latinoamérica, con más de un millón de descargas y 7.000 comunidades activas. A diferencia de Edifito, Comunidad Feliz sí ofrece un módulo de control de acceso contratable por separado, que incluye código QR dinámico, antenas TAG para acceso vehicular, lectores de reconocimiento facial y lectura de carnet chileno, además de citofonía digital integrada al mismo equipo de control de acceso. No obstante, este módulo requiere hardware adicional cotizado aparte y su función principal sigue siendo la gestión administrativa (gastos comunes, reserva de espacios, votaciones en línea) [73].
3. **Kastor:** Es un ecosistema de administración de comunidades con más de 3.000 comunidades activas en Chile, centrado en el cumplimiento de la Ley de Copropiedad Inmobiliaria [16], la

transparencia financiera y la comunicación entre administradores, comité y residentes. Su oferta se organiza en dos módulos *Kastor Operaciones* y *Kastor Conserjería* sin evidencia pública de integración con hardware de reconocimiento vehicular o control de acceso automatizado [74].

4. **ComunidadActiva:** Plataforma orientada a administradoras con múltiples condominios, que permite gestionar gastos comunes, morosidad, reservas, incidencias y comunicaciones desde un solo panel. Su módulo de accesos se limita a la generación de códigos QR para visitas, sin incorporar hardware de reconocimiento vehicular ni lectura de patentes [75].

En conjunto, estas cuatro plataformas evidencian que el mercado de gestión administrativa ha comenzado a incorporar funcionalidades de control de acceso – particularmente Comunidad Feliz – pero de forma accesoria y dependiente de hardware propietario contratado por separado, sin que la identificación vehicular automatizada constituya una capacidad nativa del producto. La Tabla 3.4 resume estas funcionalidades para las cuatro soluciones revisadas, evidenciando que el control de acceso es, en el mejor de los casos, un módulo opcional o limitado a códigos QR, nunca una capacidad equiparable a la gestión de gastos comunes o la reserva de espacios que sí ofrecen de forma nativa.

Tabla 3.4: Funcionalidades de plataformas de gestión administrativa

Solución	Gastos comunes	Reserva espacios	Votaciones	Control de acceso
Edifito	✓	✓	✓	–
Comunidad Feliz	✓	✓	✓	Opcional ^a
Kastor	✓	✓	✓	–
ComunidadActiva	✓	✓	–	QR

^a Contratable por separado, con hardware propietario adicional (QR, TAG vehicular, reconocimiento facial).

3.5.2. Soluciones de control de acceso físico

La segunda categoría agrupa soluciones cuyo producto central es, precisamente, la automatización del control de acceso mediante hardware dedicado. Se destacan:

1. **SafeCard:** Aplicación móvil nativa (iOS/Android) con panel de administración, que integra acceso vehicular y peatonal mediante códigos QR dinámicos, reconocimiento facial con auto-enrolamiento del usuario en la app, y lectura de patente con “tecnología anticlonación”. Ofrece tres planes escalonados (básico, intermedio y full) según la cantidad de puntos de acceso y equipamiento contratado (barreras, lectores peatonales y vehiculares), con un enfoque de comercialización por suscripción mensual medida en UF [76].

2. **Federal Access:** Integra barreras, lectores de patente, torniquetes y códigos QR en un sistema que verifica credenciales, carnet de identidad, QR, biometría o patente contra una base de datos local, destacando el almacenamiento de la información dentro de territorio chileno como argumento de cumplimiento normativo de protección de datos personales. Su propuesta de valor se centra en la velocidad de apertura y la disponibilidad de reportes en tiempo real para la administración [77].
3. **Smartki:** Ofrece control de acceso vehicular y peatonal mediante aplicación móvil, con generación de invitaciones digitales y un modo de operación con tarjeta física para comunidades medianas y pequeñas, orientado a que guardias y conserjes mantengan un registro digital de visitas y proveedores desde un dispositivo móvil [78].
4. **Citofon:** Se especializa en detección fotográfica de patentes, manteniendo un registro de las imágenes capturadas en cada ingreso, disponible para consulta en el portal del administrador. El enrolamiento de patentes autorizadas (lista blanca) se realiza a través de la aplicación de residentes, y el proveedor destaca la dificultad de falsificación de este mecanismo frente a otros métodos de autenticación, así como su utilidad para prevenir ingresos por suplantación de vehículo (“portonazos”) [79].
5. **VisitPass:** Plataforma de control de acceso con código QR, biometría y aplicación móvil, orientada tanto a condominios como a oficinas e industrias, sin que su sitio público detalle mecanismos de reconocimiento vehicular específico [80].

A diferencia de las plataformas administrativas, estas cinco soluciones sí resuelven de forma nativa la automatización del ingreso vehicular. Sin embargo, en todos los casos revisados, el mecanismo de reconocimiento, ya sea mediante lectura de patente, biometría o códigos QR, corresponde a hardware o software propietario cerrado, sin documentación técnica pública sobre la arquitectura o desempeño del modelo utilizado, y ninguna de ellas ofrece un módulo de gestión de reservas de estacionamiento de visita para residentes (véase en Tabla 3.5).

Tabla 3.5: Funcionalidades de soluciones de control de acceso físico

Solución	Ident. vehicular	QR/TAG	Biometría	Citofonía/video	Modelo documentado
SafeCard	ALPR	✓	✓	–	–
Federal Access	ALPR	✓	✓	–	–
Smartki	–	✓	–	–	–
Citofon	ALPR	–	–	–	–
VisitPass	–	✓	✓	–	–

Cabe destacar que, además de las soluciones empaquetadas de marca única revisadas, el mercado chileno también presenta un modelo alternativo basado en componentes: fabricantes globales de hardware biométrico como **ZKTeco** [81] ofrecen terminales de reconocimiento facial, lectores RFID y sistemas de lectura de patentes que son integrados a medida por terceros – instaladores o integradores locales – en proyectos residenciales específicos, en lugar de comercializarse como un producto único.

3.5.3. Referentes internacionales

A nivel internacional, plataformas como **Proptia** [82] y **placa.ai** [83] representan soluciones especializadas en reconocimiento de patentes (ALPR) para comunidades cerradas de tipo HOA (del inglés *Homeowners Association*), predominantes en el mercado estadounidense. Proptia integra el reconocimiento de patentes directamente con un sistema de gestión de visitantes y control de portón, comparando cada vehículo detectado contra registros de residentes, visitantes, proveedores y contratistas, y generando un registro auditable de cada evento. Por su parte, placa.ai propone un enfoque particularmente relevante: en lugar de requerir hardware ALPR dedicado, cuyo costo oscila entre USD 742 y USD 2.500 por punto de acceso, opera mediante software que procesa el video de cámaras IP ya instaladas en la comunidad, reduciendo significativamente la barrera de adopción.

Al igual que las soluciones chilenas revisadas, ambas plataformas internacionales constituyen tecnología propietaria cerrada, sin documentación técnica ni evidencia empírica publicada sobre la arquitectura, desempeño o proceso de entrenamiento de sus modelos de detección y reconocimiento. Tampoco integran un módulo de gestión de reservas de estacionamiento de visita orientado a residentes (véase en Tabla 3.6).

Tabla 3.6: Funcionalidades de referentes internacionales de reconocimiento de patentes

Solución	País	Ident. vehicular	Hardware dedicado	Modelo documentado
Proptia	EE.UU.	ALPR	✓	–
placa.ai	EE.UU.	ALPR	– ^a	–

^a No requiere hardware ALPR dedicado; opera sobre cámaras IP genéricas ya instaladas en la comunidad.

3.5.4. Síntesis comparativa

Se observa una separación funcional relativamente marcada entre plataformas de gestión administrativa y soluciones de control de acceso físico: las primeras priorizan la digitalización de procesos financieros y comunicacionales, incorporando control de acceso como un módulo opcional dependiente de hardware contratado por separado (caso de Comunidad Feliz); las segundas resuelven de forma nativa

la automatización del ingreso vehicular y peatonal, pero no ofrecen herramientas de administración financiera de la comunidad. A esta separación se suma un tercer modelo, de naturaleza distinta a los dos anteriores: la integración por proyecto de hardware de fabricantes globales como ZKTeco, ensamblada a medida por instaladores locales en lugar de comercializarse como producto único.

En segundo lugar, tanto las soluciones nacionales de control de acceso físico como los referentes internacionales especializados en reconocimiento de patentes (Proptia, placa.ai) comparten dos limitaciones transversales. Por un lado, en ningún caso se documenta públicamente la arquitectura, el desempeño o el proceso de entrenamiento del modelo de reconocimiento utilizado, tratándose en todos los casos de tecnología propietaria cerrada, limitación que se acentúa aún más en el modelo de integración por componentes, donde ni siquiera existe un producto único cuyo desempeño documentar. Por otro lado, ninguna de las siete soluciones empaquetadas revisadas, ni nacionales ni internacionales, integra un módulo de gestión de reservas de estacionamiento de visita orientado a residentes, limitándose a la autorización y el registro del ingreso vehicular.

Una excepción parcial a la dependencia de hardware propietario dedicado es placa.ai, que opera mediante software sobre cámaras IP genéricas ya instaladas en la comunidad, reduciendo así la barrera de adopción asociada al costo del equipamiento. No obstante, esta ventaja arquitectónica tampoco viene acompañada de documentación técnica pública sobre el modelo empleado ni de un módulo de reservas de estacionamiento.

En conjunto, el panorama revisado evidencia que la automatización del control de acceso vehicular y la gestión administrativa de comunidades residenciales se han desarrollado como categorías de producto mayormente independientes, y que la transparencia técnica sobre los modelos de reconocimiento, así como la integración de dichos modelos con la gestión de estacionamientos de visita, permanece como un área escasamente cubierta por la oferta comercial actual, tanto en el mercado chileno – ya sea empaquetado o por integración de componentes – como en los referentes internacionales analizados.

Capítulo 4

Metodología y Desarrollo del Sistema

La solución propuesta consiste en un sistema inteligente orientado a automatizar el control de acceso vehicular en comunidades residenciales de acceso cerrado, así como a optimizar la gestión de los estacionamientos de visita. Para ello, se integrarán tecnologías de reconocimiento automático de patentes (ALPR), sensores IoT y aplicaciones digitales para facilitar la interacción entre los distintos usuarios.

4.1. Requisitos del Sistema

En esta sección se detallan las especificaciones técnicas y operativas necesarias para el correcto funcionamiento de la solución propuesta. Estos requisitos han sido definidos a partir de las necesidades de gestión de accesos vehiculares y administración de estacionamientos de visita en las comunidades residenciales de acceso cerrado, y se estructuran en cuatro categorías principales: (i) funcionales, (ii) no funcionales, (iii) de interfaz, y (iv) de ambiente. Esta clasificación permite separar las acciones y servicios específicos que el sistema debe proveer de los atributos de calidad y restricciones tecnológicas que garantizarán su viabilidad, seguridad y rendimiento a largo plazo.

4.1.1. Requisitos Funcionales

Los requisitos funcionales describen las acciones, comportamientos y servicios específicos que el sistema debe ser capaz de ejecutar para satisfacer las necesidades de sus diferentes usuarios (residentes, conserjes y administradores). Estas especificaciones abarcan desde la detección automatizada de vehículos hasta la gestión centralizada de datos, control de roles y administración de reservas. A continuación, se detallan estos requisitos organizados según sus respectivos módulos operativos:

- **RF01:** El sistema debe detectar automáticamente los vehículos autorizados mediante reconocimiento de patentes.
- **RF02:** El sistema debe tener un registro de accesos vehiculares que incluya fecha, hora, patente y resultado del intento de acceso.
- **RF03:** El sistema debe dar acceso a los conserjes al registro de accesos vehiculares.
- **RF04:** El sistema debe notificar, en tiempo real, al personal de conserjería ante accesos vehiculares no autorizados o no reconocidos.
- **RF05:** El sistema debe detectar los espacios de estacionamiento de visita disponibles u ocupados mediante sensores IoT.
- **RF06:** El sistema debe permitir modificar ciertos parámetros de las políticas internas de uso de estacionamientos: duración máxima, horarios permitidos y tolerancia.
- **RF07:** El sistema debe permitir a los residentes visualizar el estado actual de los estacionamientos de visita.
- **RF08:** El sistema debe permitir a los residentes, propietarios y administradores realizar reservas de estacionamientos para visitas.
- **RF09:** El sistema debe registrar un historial del uso de los estacionamientos, incluyendo vehículo, hora de entrada, salida y duración.
- **RF10:** El sistema debe generar alertas cuando un vehículo permanezca más tiempo del permitido en un estacionamiento de visitas.
- **RF11:** El sistema debe permitir la recuperación de contraseñas para los usuarios registrados.
- **RF12:** El sistema debe validar los roles de los usuarios (residente, conserje o administrador) para controlar el acceso a las funcionalidades correspondientes.

- **RF13:** El sistema debe registrar información clave como usuarios, vehículos, accesos y reservas en una base de datos centralizada.
- **RF14:** El sistema debe permitir la gestión de usuarios (creación, lectura, actualización y eliminación) por parte de los administradores.
- **RF15:** El sistema debe permitir al administrador configurar el número total de estacionamientos de visita disponibles.

4.1.2. Requisitos No Funcionales

Los requisitos no funcionales establecen las restricciones tecnológicas y los atributos de calidad que el sistema debe cumplir para asegurar una operación robusta y una experiencia de usuario óptima. A diferencia de los requisitos funcionales, estos no definen *qué* hace el sistema, sino *cómo* lo hace, abordando dimensiones críticas como el rendimiento bajo distintas condiciones, la usabilidad de las interfaces, la seguridad de los datos, el cumplimiento normativo y la escalabilidad de la arquitectura. Los criterios definidos para este proyecto se detallan a continuación:

- **RNF1:** El sistema de detección de accesos vehiculares debe ser capaz de operar de manera robusta bajo diversas condiciones ambientales, tales como variaciones de iluminación, lluvia moderada u obstrucciones parciales, garantizando un reconocimiento confiable de las patentes en contextos típicos de operación.
- **RNF2:** El sistema de detección de disponibilidad en los estacionamientos de visitas debe funcionar adecuadamente en distintos entornos físicos, incluyendo espacios subterráneos o con condiciones estructurales complejas. Debe ser capaz de detectar de forma confiable la presencia de vehículos en cada espacio, sin verse afectado por factores como iluminación, temperatura o interferencias electromagnéticas propias del entorno urbano.
- **RNF3:** La aplicación móvil para residentes debe tener una interfaz intuitiva y fácil de usar para los usuarios del sistema.
- **RNF4:** La aplicación web para administradores debe permitir un monitoreo en tiempo real o con mínima latencia de los eventos del sistema.
- **RNF5:** El sistema debe cumplir con la normativa de protección de datos personales vigente (por ejemplo, la Ley N° 19.628 en Chile), garantizando la confidencialidad y seguridad de la información procesada.

- **RNF6:** La base de datos debe implementar medidas básicas de seguridad, incluyendo encriptación de contraseñas y validación de datos de entrada.
- **RNF7:** Las aplicaciones móviles y web deben ser compatibles con las versiones más comunes de navegadores (Chrome, Firefox, Edge) y sistemas operativos móviles (Android 10+ y iOS 13+).
- **RNF8:** La arquitectura del sistema debe ser escalable, permitiendo integrar nuevos módulos o extender su cobertura a múltiples comunidades de acceso cerrado sin reestructurar el sistema base.

4.1.3. Requisitos de Interfaces

El sistema debe proporcionar diversas interfaces para interactuar con los usuarios y otros componentes técnicos, garantizando una operación fluida y eficiente.

4.1.3.1. Interfaces de Usuario

Estas interfaces representan el principal punto de contacto e interacción entre los distintos actores (residentes, administradores y personal de conserjería) y las funcionalidades del sistema. Su diseño y desarrollo están orientados a satisfacer las necesidades operativas específicas de cada uno, priorizando en todo momento la usabilidad, la accesibilidad y una experiencia de usuario (UX) intuitiva. El objetivo central de esta capa es presentar la información en tiempo real de manera clara, minimizando la curva de aprendizaje y permitiendo la ejecución de tareas de forma rápida y eficiente desde múltiples plataformas. A continuación se detallan las interfaces consideradas:

Aplicación Web (Panel de Administración y Conserjería): Se debe proveer una aplicación web responsiva destinada a administradores y conserjes. Sus principales funcionalidades incluyen:

- Monitoreo en tiempo real de las cámaras de acceso vehicular (portón).
- Gestión integral de usuarios del sistema (altas, bajas, modificaciones de roles y permisos).
- Visualización de registros históricos (logs) de acceso vehicular, detallando las patentes, fechas, horas y estado de los accesos.

Aplicación Móvil (Residentes y Propietarios): Se debe desarrollar una aplicación orientada a residentes y propietarios, compatible con plataformas móviles modernas. La interfaz debe permitir:

- Registro de nuevos usuarios e inicio de sesión seguro.

- Visualización de la disponibilidad actual de los estacionamientos de visita en tiempo real.
- Creación, modificación y cancelación de reservas de estacionamientos, especificando fechas y horarios.

4.1.3.2. Interfaces Técnicas y de Comunicación

Estas interfaces establecen los canales, protocolos y estándares necesarios para garantizar una interconexión segura, eficiente y de baja latencia entre los diversos módulos físicos y lógicos del sistema. Su correcta implementación es crítica para orquestar el flujo de datos continuo entre los dispositivos de hardware desplegados en terreno (como cámaras, sensores y microcontroladores de borde) y la lógica de negocio central en el backend, manteniendo la integridad operativa en todo momento.

Interfaz con Cámaras de Vigilancia IP: El sistema se integrará con el hardware de captura de video para alimentar el módulo de visión artificial.

- Se plantea utilizar protocolos de comunicación estándar y seguros para transmisión de video e interoperabilidad (por ejemplo, RTSP, ONVIF).
- La conexión a la red de datos se debe realizar mediante medios cableados (por ejemplo, Ethernet) o inalámbricos, según la factibilidad técnica del recinto.

API RESTful: Se debe desarrollar una API robusta que actúe como puente de comunicación entre los clientes (aplicación móvil y web) y los servicios centrales del backend.

- Su función es centralizar las operaciones lógicas, tales como autenticación, gestión de bases de datos y la orquestación de los modelos de IA.
- Se deben implementar estándares de seguridad y cifrado (HTTPS/TLS) para proteger la confidencialidad e integridad de la información transmitida.

4.1.3.3. Interfaces con Servicios Externos

Con el fin de potenciar sus capacidades, optimizar recursos locales y garantizar un alto nivel de escalabilidad, el sistema requiere interoperar con plataformas e infraestructuras provistas por terceros. Estas interfaces permiten delegar operaciones especializadas o de alta demanda, tales como el almacenamiento seguro y redundante, la inferencia de modelos complejos y la mensajería omnicanal, hacia ecosistemas externos robustos, asegurando así una mayor disponibilidad y facilidad de mantenimiento.

Servicios de Almacenamiento en la Nube: Interfaz para el respaldo persistente y seguro de datos críticos (registros de acceso, credenciales cifradas y auditorías), facilitando la recuperación ante desastres y el cumplimiento normativo.

Plataformas de Cómputo e IA: En caso de delegar el procesamiento, el sistema deberá ser capaz de enviar datos (imágenes o flujos de video) a plataformas externas o servicios de inferencia en la nube mediante APIs seguras, y recibir los resultados estructurados de vuelta con mínima latencia.

Sistemas de Notificaciones (Push/Email): Integración con proveedores de mensajería (por ejemplo, Firebase Cloud Messaging, SendGrid, Amazon SES) para alertar a los usuarios de manera oportuna.

- Envío de notificaciones push a la aplicación móvil (por ejemplo, confirmación de reservas, alertas de tiempo excedido en estacionamientos).
- Envío de correos electrónicos transaccionales para recuperación de cuentas, validación de identidad o reportes periódicos.

4.1.4. Requisitos de Ambiente

A continuación, se describe el entorno tecnológico (hardware y software) en el que el sistema será desarrollado, desplegado y operado, considerando diversas alternativas para garantizar flexibilidad, escalabilidad e independencia de un proveedor único.

4.1.4.1. Hardware

El sistema requiere una infraestructura física que contemple capacidades de borde (edge), transmisión y procesamiento central:

- **Dispositivos de Borde y Microcontroladores:** Equipos para la recolección de datos de sensores y procesamiento. Se contemplan desde microcontroladores de bajo consumo (por ejemplo, ESP32, Arduino, familias STM32) para gestión de estacionamientos, hasta computadoras de placa única (SBC) o aceleradores edge para tareas de IA local (por ejemplo, familia Raspberry Pi, NVIDIA Jetson, Coral Dev Board, entre otros).
- **Sensores de Ocupación:** Dispositivos instalados en los espacios de visita. La tecnología elegida (magnéticos, ultrasónicos, infrarrojos o visión artificial dedicada) dependerá de las condiciones

ambientales e infraestructura del estacionamiento.

- **Cámaras de Vigilancia:** Hardware para la captura de video del control de accesos (ALPR). Se requiere conectividad IP mediante PoE o conexión inalámbrica, resolución adecuada (Full HD o superior), control de exposición para lectura de patentes y soporte para visión nocturna (IR).
- **Infraestructura de Red:**
 - **Redes IoT:** Protocolos inalámbricos de bajo consumo para la comunicación de sensores de estacionamiento (por ejemplo, LoRaWAN, ZigBee, 6LoWPAN, Bluetooth Mesh, o Wi-Fi optimizado).
 - **Redes de Banda Ancha:** Infraestructura local Ethernet y Wi-Fi estándar para interconectar cámaras, nodos edge y gateways con acceso a internet.
- **Servidores y Procesamiento Central:**
 - **Backend y Base de Datos:** Infraestructura local (on-premise) o instancias en la nube (por ejemplo, AWS, Azure, Google Cloud, DigitalOcean) dimensionadas según el tráfico de la comunidad residencial.
 - **Aceleración de IA:** Hardware con soporte para GPU o TPU en caso de requerir inferencia de modelos complejos en tiempo real a nivel de servidor central.
- **Dispositivos Cliente:** Smartphones estándar y computadoras (con navegadores web actualizados) para los usuarios finales y administradores.
- **Respaldo Energético:** Se recomienda el uso de sistemas UPS para mantener operativos gateways, switches y cámaras críticas frente a cortes eléctricos.

4.1.4.2. Software

El sistema se apoyará en un stack tecnológico moderno, admitiendo diversas alternativas comerciales y de código abierto:

- **Sistemas Operativos:**
 - **Servidores:** Distribuciones Linux (por ejemplo, Ubuntu, Debian, CentOS/RHEL), sistemas Unix-like (por ejemplo, FreeBSD) o entornos Windows Server.
 - **Dispositivos Edge/Microcontroladores:** Sistemas operativos ligeros (por ejemplo, Raspberry Pi OS, Armbian), sistemas de tiempo real (RTOS como FreeRTOS) o firmware bare-metal.

- **Clientes:** iOS y Android para ecosistemas móviles; Windows, macOS y Linux para escritorio.
- **Desarrollo Backend y APIs:** Lenguajes y frameworks orientados a servicios, tales como Python (Flask, FastAPI, Django), Node.js (Express, NestJS), Java (Spring Boot), C# (.NET) o Go.
- **Desarrollo Frontend:**
 - **Web:** HTML5/CSS3/JS complementado con frameworks reactivos (por ejemplo, React.js, Angular, Vue.js, Svelte).
 - **Móvil:** Frameworks multiplataforma/híbridos (por ejemplo, Flutter, React Native, MAUI) o desarrollo nativo (Kotlin/Java, Swift) según los requerimientos de rendimiento.
- **Bases de Datos:** Motores relacionales (por ejemplo, PostgreSQL, MySQL, SQL Server) para asegurar la integridad de reservas y usuarios, con la posibilidad de integrar bases NoSQL (por ejemplo, MongoDB, Redis) para el manejo de logs y telemetría de sensores.
- **Visión por Computador e IA:** Uso de librerías estándar de procesamiento de imágenes (por ejemplo, OpenCV) combinadas con frameworks de Machine/Deep Learning (por ejemplo, TensorFlow/Keras, PyTorch, o modelos basados en arquitecturas YOLO para la detección de objetos).
- **Herramientas DevOps y Versionamiento:** Uso de Git (GitHub, GitLab, Bitbucket) para el control de código fuente, apoyado opcionalmente por herramientas de contenerización (Docker) para un despliegue estandarizado en múltiples entornos.

4.1.5. Perfiles de Usuario

El sistema está diseñado para integrarse a la dinámica de comunidades residenciales de acceso cerrado. En este contexto, se identifican diversos perfiles de usuarios que desempeñan roles clave dentro de la operación cotidiana de las comunidades residenciales de acceso cerrado. A continuación, se describen estos perfiles desde una perspectiva organizacional y comunitaria:

- **Residentes y Propietarios:** Corresponden a las personas que habitan o son propietarias de los departamentos dentro del complejo residencial. En su mayoría, cuentan con uno o más vehículos asociados y, en algunos casos, reciben visitas que requieren el uso de los estacionamientos comunes. Son usuarios frecuentes de los espacios compartidos y actores centrales en la vida de la comunidad residencial.
- **Conserjes:** Son responsables de las labores operativas y de vigilancia en el acceso a la comunidad residencial. Tienen contacto directo con residentes y visitantes, y tradicionalmente desempeñan

el rol de intermediarios en la autorización de accesos vehiculares. Su función es clave para el cumplimiento de las normas internas y la seguridad del recinto.

- **Administradores:** Encargados de la gestión general de la comunidad. Representan a la administración en la toma de decisiones sobre el uso de espacios, la política de accesos y la relación con los residentes. Su rol es garantizar el funcionamiento eficiente de la comunidad residencial, el cumplimiento del reglamento interno y la resolución de conflictos comunitarios.
- **Visitantes:** Son personas ajenas a la comunidad que acceden temporalmente a la comunidad residencial, generalmente por invitación de residentes. Aunque no tienen un vínculo permanente, su acceso al recinto debe ser gestionado de manera controlada para resguardar la seguridad de la comunidad y el uso adecuado de los recursos compartidos, como los estacionamientos de visita.

4.2. Arquitectura del Sistema

Con base en los requisitos del sistema, se plantea un diseño basado en una arquitectura de capas jerárquicas, lo que permite desacoplar la obtención de datos físicos, la comunicación de red y la lógica de negocio, facilitando la escalabilidad, el mantenimiento y la seguridad de ResiPark de extremo a extremo. Esta estructura jerárquica corresponde a un **modelo de cuatro capas para sistemas IoT** (véase en Figura 4.1), este fue abordado en [6], por lo que supone una solución viable al flujo del sistema. A continuación, se detallan las capas y sus respectivas funciones:

1. **Capa de Obtención de Datos:** Esta capa es responsable de la digitalización del entorno físico. Para abordar el control de acceso vehicular y la gestión de estacionamientos de visita, se proponen dos subsistemas principales: el **módulo de control de acceso** y el **módulo de monitoreo de estacionamientos**.
2. **Capa de Comunicación:** La capa de comunicación es responsable de transportar la información recolectada por los módulos físicos hacia la capa de soporte, seleccionando en cada caso el protocolo más adecuado según los requerimientos de latencia, alcance y consumo energético del subsistema involucrado. En el caso de los sensores IoT, se debe priorizar un protocolo que permita reducir el consumo energético y cubrir las distancias de comunicación requeridas. Para el caso del acceso vehicular, la información proveniente de las cámaras debe transmitirse con baja latencia y de forma íntegra, por lo que es necesario un protocolo que cumpla con estas especificaciones.
3. **Capa de Soporte (Middleware):** La capa de soporte constituye el middleware del sistema, responsable de la lógica de negocio que conecta la información capturada en la capa física con

las decisiones de autorización y registro de acceso. Esta capa debe componerse de tres servicios interdependientes que operen de manera coordinada. El primero corresponde a un **microservicio de detección y reconocimiento**, el cual debe ser capaz de procesar las imágenes provenientes de las cámaras de vigilancia mediante modelos de visión por computador, tanto para el reconocimiento automático de patentes (ALPR) como para la clasificación complementaria del vehículo. El segundo corresponde a una **API de verificación y registro**, respaldada por una base de datos relacional centralizada, la cual debe ser capaz de validar los resultados entregados por el microservicio de detección contra los registros de vehículos y usuarios autorizados, de aplicar las políticas de acceso y estacionamiento definidas por los administradores, y de dejar constancia de cada evento (accesos, reservas, uso de estacionamientos) para su posterior consulta. El tercero corresponde a un **orquestador de borde**, el cual debe ser capaz de coordinar el flujo de información entre la capa de comunicación y la capa de soporte, gestionando la latencia de los eventos provenientes de los sensores IoT y las cámaras. Dado que esta capa concentra información sensible de los residentes, su diseño debe incorporar mecanismos de seguridad como la encriptación de datos en tránsito y en reposo, la autenticación entre servicios internos, y una arquitectura desacoplada que permita escalar horizontalmente o incorporar nuevos módulos sin afectar el funcionamiento del sistema base.

4. **Capa de Aplicación:** La capa de aplicación constituye el punto de contacto entre el sistema y sus usuarios finales, y debe ser capaz de traducir la información procesada por la capa de soporte en interfaces accesibles y comprensibles para cada perfil. Para los conserjes y administradores, se debe disponer de una **aplicación web** que permita el monitoreo en tiempo real de los accesos vehiculares, la consulta de registros históricos y la gestión de usuarios y políticas del sistema. Para los residentes y propietarios, se debe ofrecer una **aplicación móvil** orientada a permitir la visualización del estado de los estacionamientos de visita y la gestión de reservas para sus visitantes. Ambas interfaces deben consumir la información expuesta por la API RESTful de la capa de soporte, respetando los roles y permisos definidos para cada tipo de usuario. De esta forma, la capa de aplicación debe ser capaz de cerrar el ciclo del sistema, convirtiendo los datos capturados y procesados en las capas inferiores en decisiones y acciones concretas para la comunidad residencial.

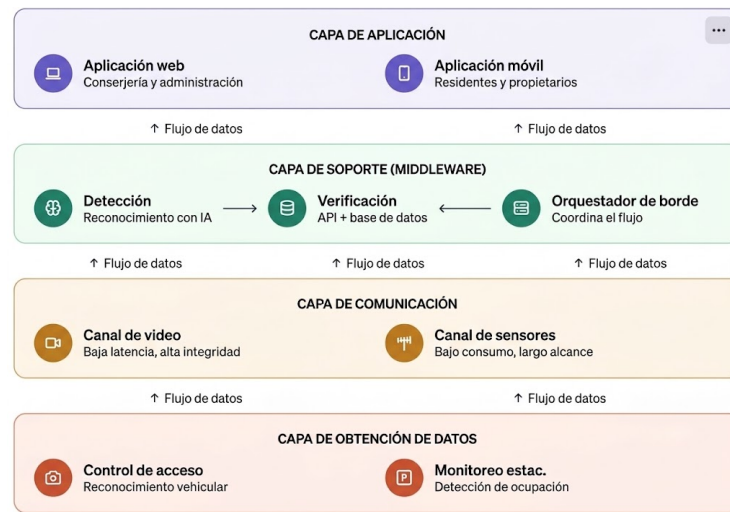


Figura 4.1: Arquitectura de Sistema ResiPark, **modelo 4 capas** sistema IoT

4.3. Módulos de Desarrollo

El sistema permitirá el ingreso automatizado de vehículos cuyas patentes estén previamente registradas como autorizadas. Estas podrán corresponder a vehículos de residentes, propietarios, visitas o personal autorizado por la administración. La validación se realizará mediante cámaras y algoritmos de reconocimiento de patentes vehiculares, eliminando la necesidad de intervención manual en la mayoría de los casos.

Además, se incorporará la detección y el monitoreo del uso de los estacionamientos de visita. A través de una red de sensores conectados mediante un protocolo de comunicación, el sistema será capaz de detectar la presencia de vehículos en cada espacio individual, registrar el tiempo de ocupación y generar alertas en caso de uso indebido o excesivo. Esto permitirá una supervisión eficiente, precisa y escalable en las comunidades residenciales de acceso cerrado. El sistema contará con dos interfaces principales: (i) una **Aplicación Móvil**, destinada a residentes, propietarios, desde la cual podrán realizar reservas de estacionamiento de visitas y gestionarlas, y (ii) una **Aplicación Web**, con foco principal en el personal de conserjería, que les permitirá visualizar eventos del sistema en tiempo real y recibir notificaciones sobre accesos no autorizados o infracciones.

Para estructurar el desarrollo del sistema de forma clara y modular, se identifican cuatro componentes principales, cada uno con funcionalidades específicas que contribuyen al funcionamiento global de la solución. Estos *módulos* permiten una organización más eficiente del trabajo, además de facilitar el análisis y la definición de requisitos para cada parte del sistema:

- **Módulo 1 - Gestión de Accesos Vehiculares (M1):** Encargado de gestionar el ingreso y salida de vehículos al recinto residencial. Este módulo se apoya en tecnologías de visión por computador y reconocimiento automático de patentes para permitir o denegar accesos en función de las autorizaciones registradas en el sistema. A nivel de hardware, considera cámaras IP (Ethernet/PoE o Wi-Fi) con resolución Full HD y visión nocturna, integradas a un dispositivo de procesamiento en el borde tipo Raspberry Pi; a nivel de software, contempla el uso de Python con OpenCV, frameworks de aprendizaje profundo como TensorFlow o PyTorch, y plataformas en la nube para entrenamiento e inferencia (Google Cloud AI Platform, AWS SageMaker, Azure Machine Learning o Google Colab).
- **Módulo 2 - Gestión de Estacionamientos de Visitantes (M2):** Permite monitorear y administrar el uso de los estacionamientos destinados a visitas. Utiliza una red de sensores IoT para detectar la presencia de vehículos en tiempo real. El sistema registra el tiempo de ocupación de cada espacio y genera alertas en caso de infracciones o uso prolongado no autorizado, facilitando así una gestión eficiente y transparente de estos recursos compartidos. A nivel de hardware, considera sensores de presencia por estacionamiento, microcontroladores y gateways de bajo consumo, comunicados mediante un protocolo de red de bajo consumo y largo alcance como LoRaWAN.
- **Módulo 3 - Interfaces de Usuario (M3):** Incluye las aplicaciones web y móviles destinadas a los distintos perfiles de usuarios. A través de estas interfaces, residentes, conserjes y administradores podrán interactuar con el sistema, realizar gestiones, recibir notificaciones y visualizar información relevante. A nivel de hardware, considera smartphones (iOS y Android) y computadores con navegadores modernos como dispositivos cliente; a nivel de software, contempla frameworks web como React.js, Angular o Vue.js junto a HTML5/CSS3 para la aplicación web, y frameworks híbridos como React Native o Flutter, o desarrollo nativo (Kotlin/Java, Swift/Objective-C) para la aplicación móvil.
- **Módulo 4 - Servicios Centrales del Sistema (M4):** Este módulo corresponde al núcleo de integración tecnológica del sistema. Incluye las API, la base de datos y la lógica de negocio que permiten que los distintos componentes funcionen de manera coherente y centralizada, actuando como intermediario entre los módulos de percepción, gestión e interfaces. A nivel de hardware, considera servidores de backend y base de datos – desplegados localmente o en la nube (AWS, GCP, Azure) – e infraestructura de red (Ethernet, Wi-Fi, routers, switches y gateways); a nivel de software, contempla lenguajes y frameworks de backend como Python (Flask o FastAPI), Node.js (Express.js) o Java, sistemas gestores de bases de datos relacionales como PostgreSQL o MySQL, y control de versiones mediante Git sobre plataformas como GitHub, GitLab o Bitbucket.

4.4. Diagrama de Contexto

El diagrama de contexto (véase en Figura 4.2) proporciona una vista de alto nivel de la arquitectura de la solución, definiendo los límites del sistema y estableciendo las interacciones fundamentales con las entidades externas, ya sean usuarios o dispositivos de hardware.

El núcleo de la solución está orquestado por el módulo de **Servicios Centrales (M4)**, el cual se encarga de almacenar la información en la Base de Datos, procesar la Gestión de Accesos (M1), gestionar los Estacionamientos (M2) y soportar las Interfaces (M3).

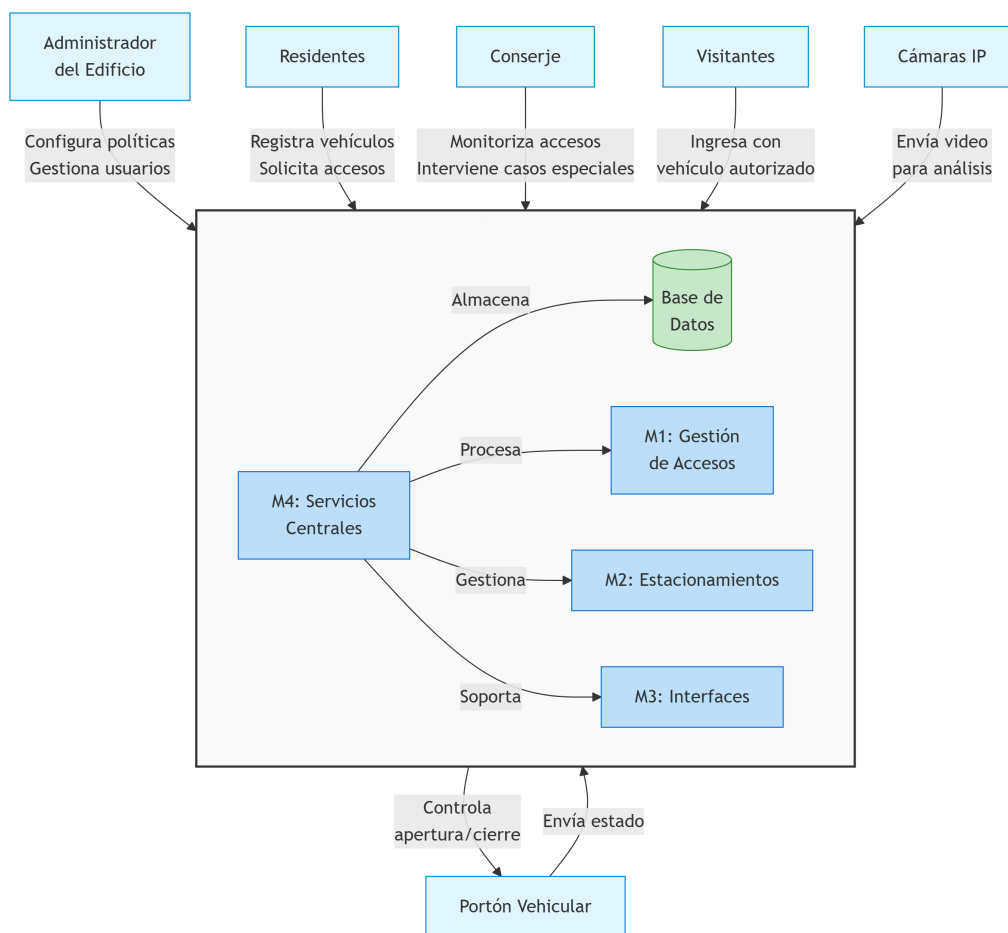


Figura 4.2: Diagrama de Contexto del Sistema.

Las interacciones del sistema con su entorno se estructuran según los siguientes actores y componentes externos:

- **Administrador:** Interactúa con el sistema para configurar las políticas operativas y gestionar a los usuarios.
- **Residentes:** Utilizan la plataforma para registrar sus vehículos y solicitar los accesos de sus visitas.
- **Conserje:** Emplea el sistema para monitorizar los accesos vehiculares en tiempo real e intervenir de forma manual en casos especiales o excepciones.
- **Visitantes:** Su interacción es física, ingresando al recinto únicamente cuando cuentan con un vehículo autorizado.
- **Cámaras IP:** Actúan como periféricos de entrada continua, enviando el flujo de video hacia el sistema para el análisis y reconocimiento de patentes.
- **Portón Vehicular:** Mantiene una comunicación bidireccional; el sistema envía instrucciones para controlar su apertura y cierre, mientras que el hardware informa su estado físico actual.

4.4.1. Matriz de Requisitos Funcionales y Módulos

La siguiente matriz de trazabilidad establece la relación directa entre los requisitos levantados (tanto funcionales como no funcionales) y los distintos módulos arquitectónicos propuestos para el sistema (M1, M2, M3 y M4). El propósito fundamental de esta herramienta es garantizar que cada requerimiento esté respaldado y cubierto por al menos un componente operativo, asegurando así la completitud del diseño. Adicionalmente, esta vista cruzada permite identificar el nivel de responsabilidad e impacto que posee cada módulo dentro de la solución, lo cual resulta crítico para planificar las fases de desarrollo, priorizar esfuerzos de prueba y evaluar el alcance de futuros mantenimientos o actualizaciones (véase en Tabla 4.1).

Tabla 4.1: Matriz de Trazabilidad de Requisitos y Módulos

Requisito	M1	M2	M3	M4
RF01: Detección de vehículos autorizados	X			X
RF02: Registro de accesos	X		X	X
RF03: Visualización del registro de accesos	X		X	X

Requisito	M1	M2	M3	M4
RF04: Notificación de accesos no autorizados	X		X	X
RF05: Detección de estacionamientos disponibles		X	X	X
RF06: Modificación de políticas de estacionamiento			X	X
RF07: Visualización del estado de los estacionamientos		X	X	X
RF08: Reserva de estacionamientos de visitantes	X	X	X	X
RF09: Registro de uso de estacionamientos		X		X
RF10: Alertas por exceso de tiempo		X	X	X
RF11: Recuperación de contraseñas			X	X
RF12: Validación de roles de usuario			X	X
RF13: Registro de información centralizada			X	X
RF14: Gestión de usuarios			X	X
RF15: Configuración de estacionamientos		X	X	X
RNF01: Robustez en reconocimiento vehicular	X			
RNF02: Robustez en detección de estacionamientos		X		
RNF03: Usabilidad de la aplicación móvil			X	
RNF04: Monitoreo en tiempo real	X	X	X	X
RNF05: Cumplimiento de normativas de privacidad	X	X	X	X
RNF06: Seguridad de la información				X
RNF07: Compatibilidad multiplataforma			X	
RNF08: Escalabilidad del sistema	X	X	X	X

4.5. Diseño de Interfaces

En esta sección se define la arquitectura de la información y la experiencia de usuario (UX) que estructuran la interacción con el sistema. El diseño de interfaces tiene como objetivo principal garantizar que la comunicación entre los distintos actores (residentes, personal de conserjería y administradores) y la plataforma sea intuitiva, eficiente y visualmente coherente, minimizando la curva de aprendizaje y facilitando la ejecución rápida de tareas críticas como la gestión de accesos y reservas.

4.5.1. Modelo de Navegación

A continuación, se presenta un diagrama de flujo que ilustra el modelo de navegación propuesto para la aplicación móvil “ResiPark”. Este modelo visualiza el recorrido lógico del usuario a través de las distintas pantallas, estableciendo una separación clara entre las vistas públicas y las privadas.

El diagrama (véase en Figura 4.3) se divide en dos contextos principales que encapsulan las funcionalidades accesibles según el estado de la sesión del usuario: el Flujo No Autenticado y el Flujo Autenticado.

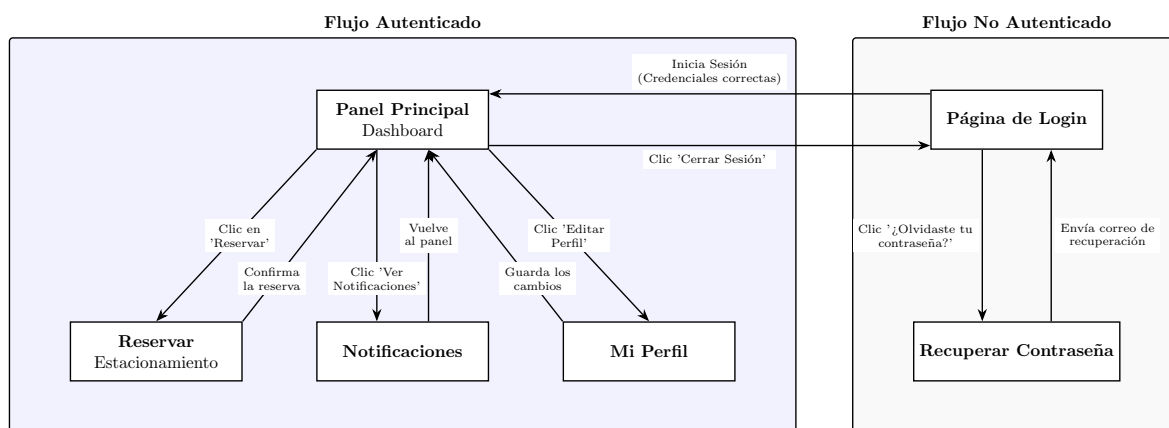


Figura 4.3: Diagrama de flujo rediseñado que representa la navegación del usuario en la aplicación ResiPark.

4.5.1.1. Flujo No Autenticado

Este flujo representa la experiencia inicial del usuario antes de validar su identidad en el sistema.

- **Página de Login:** Es el punto de entrada principal a la aplicación. El usuario debe ingresar sus credenciales (correo y contraseña) para acceder al sistema.

- **Recuperación de Contraseña:** Si el usuario olvida sus credenciales, puede acceder a un formulario para solicitar la recuperación mediante la opción “¿Olvidaste tu contraseña?”. Una vez enviado el correo de recuperación, el sistema lo redirige de vuelta a la pantalla de inicio de sesión.

El único camino para salir de este flujo y acceder a las funcionalidades del sistema es mediante un inicio de sesión exitoso con credenciales correctas.

4.5.1.2. Flujo Autenticado

Una vez que se inicia sesión correctamente, se ingresa a este flujo privado, cuyo eje central es el Panel de Residente.

- **Panel Principal (Dashboard):** Actúa como el centro neurálgico desde donde se puede acceder a todas las funcionalidades principales de la aplicación.
- **Secciones Principales:** Desde el panel central, el usuario puede navegar hacia las siguientes vistas:
 1. **Reservar Estacionamiento:** Interfaz para gestionar y crear nuevas reservas. Tras confirmar la reserva, el sistema retorna al Dashboard.
 2. **Notificaciones:** Espacio para visualizar avisos y comunicados importantes. El usuario puede volver al panel central fácilmente tras revisarlas.
 3. **Mi Perfil:** Sección para actualizar la información personal y preferencias. Al guardar los cambios, se redirige nuevamente al panel principal.
- **Cierre de Sesión:** El usuario puede cerrar su sesión en cualquier momento desde el panel principal, acción que revoca su acceso y lo devuelve al “Flujo No Autenticado” (la página de login).

Este modelo estructurado asegura una navegación intuitiva y mantiene al usuario autenticado dentro de un ciclo cerrado de herramientas útiles y protegiendo la información mediante un control de acceso centralizado.

4.6. Mockup

Con el propósito de validar la propuesta de interacción antes de su implementación final, se desarrollan mockups de las dos interfaces principales del sistema: la aplicación web, orientada a conserjes y administradores, y la aplicación móvil, orientada a residentes y propietarios. Estos prototipos visuales

permiten representar la disposición de los elementos de cada pantalla, los flujos de navegación entre módulos y los datos que el sistema debe capturar o mostrar a cada perfil de usuario, sirviendo como referencia para el desarrollo posterior del frontend.

4.6.1. Interfaz Usuario – App Web

La pantalla de inicio de sesión de la interfaz web solicita el número de departamento y la contraseña del usuario (conserje o administrador) para acceder al panel mediante el botón “Ingresar” (véase en Figura 4.4).

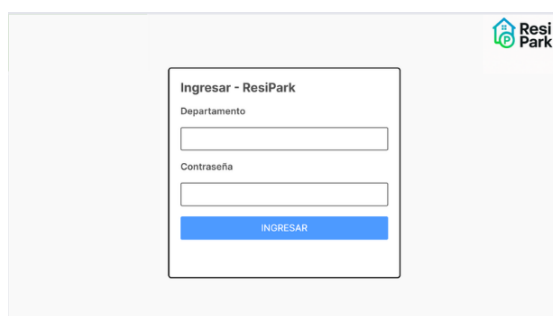


Figura 4.4: Pantalla de inicio de sesión, interfaz web.

El panel de estacionamientos muestra, en su parte superior, una grilla con el estado de cada espacio (E01 a E12), con cada espacio diferenciado según su disponibilidad, junto a una vista de cámara del área de estacionamientos. Bajo la grilla, una tabla detalla, para cada estacionamiento, su estado (ocupado, libre o reservado), el tiempo restante de ocupación y el departamento asociado. El menú lateral izquierdo permite navegar entre los módulos de Estacionamientos, Registro de Patentes y Portón (véase en Figura 4.5).

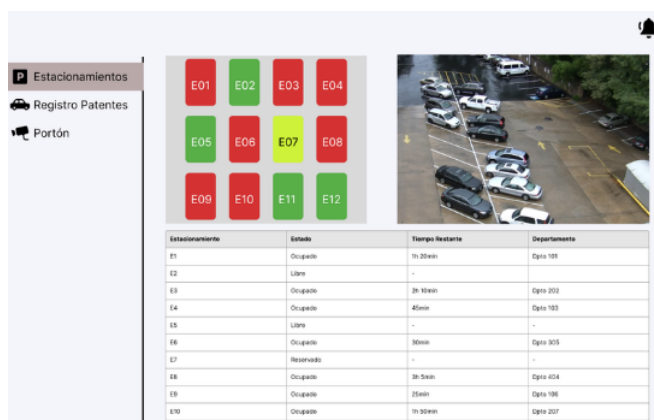
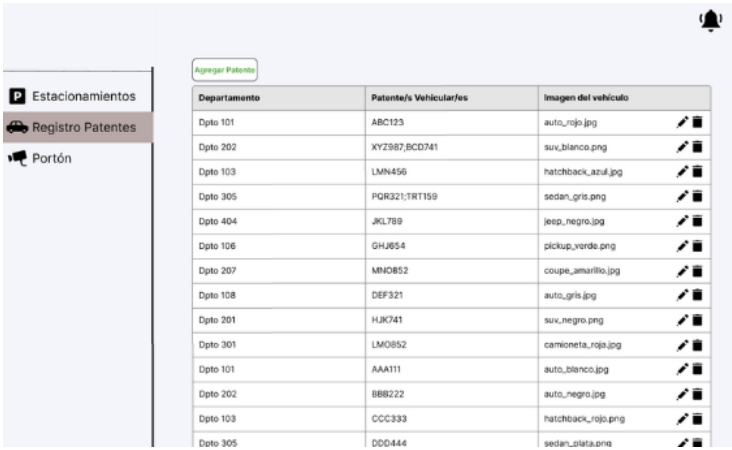


Figura 4.5: Panel de ocupación de estacionamientos, interfaz web.

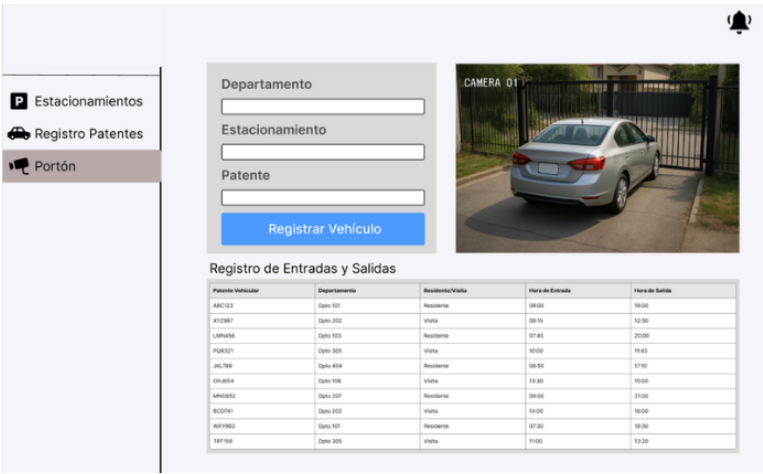
El módulo de Registro de Patentes presenta una tabla con las patentes vehiculares asociadas a cada departamento, permitiendo múltiples patentes por unidad. Cada fila incluye el departamento, la o las patentes registradas, una referencia a la imagen del vehículo asociado y acciones de edición y eliminación del registro. Un botón “Agregar Patente” en la parte superior permite incorporar nuevos registros (véase en Figura 4.6).



Departamento	Patente/s Vehicular/es	Imagen del vehículo
Dpto 101	ABC123	auto_rojo.jpg
Dpto 202	XYZ987,BCD741	suv_blanco.png
Dpto 103	LMN456	hatchback_azul.jpg
Dpto 305	PQR321,TRT159	sedan_gris.png
Dpto 404	JKL789	jeep_negro.jpg
Dpto 106	GHI654	pickup_verde.png
Dpto 207	MNO852	coupe_amarillo.jpg
Dpto 108	DEF321	auto_gris.jpg
Dpto 201	HJK741	suv_negro.png
Dpto 301	LMO852	camioneta_roja.jpg
Dpto 101	AAA111	auto_blanco.jpg
Dpto 202	BBB222	auto_negro.jpg
Dpto 103	CCC333	hatchback_rojo.png
Dpto 305	DDD444	sedan_azul.png

Figura 4.6: Módulo de registro de patentes por departamento, interfaz web.

El módulo de Portón dispone, en su parte superior, un formulario para el registro manual de un vehículo visitante – departamento, estacionamiento asignado y patente – junto a una vista en tiempo real de la cámara del acceso vehicular. Debajo, una tabla de “Registro de Entradas y Salidas” detalla, para cada evento, la patente del vehículo, el departamento, el tipo de acceso (residente o visita), y las horas de entrada y salida registradas (véase en Figura 4.7).



Patente Vehicular	Departamento	Residente/Visita	Hora de Entrada	Hora de Salida
ABC123	Dpto 101	Residente	08:00	18:00
XYZ987	Dpto 202	Visita	09:15	12:30
LMN456	Dpto 103	Residente	07:45	20:00
PQR321	Dpto 305	Visita	10:00	16:45
JKL789	Dpto 404	Residente	06:30	19:00
GHI654	Dpto 106	Visita	13:00	15:00
MNO852	Dpto 207	Residente	09:00	21:00
DEF321	Dpto 202	Visita	14:00	16:00
HJK741	Dpto 101	Residente	07:30	18:30
LMO852	Dpto 305	Visita	11:00	13:20

Figura 4.7: Módulo de portón y registro de accesos vehiculares, interfaz web.

4.6.2. Interfaz Usuario – App Móvil

La pantalla de inicio de sesión de la aplicación móvil presenta los campos de correo electrónico y contraseña, opción de recuperación de contraseña y el botón “Iniciar Sesión”. Bajo el formulario se muestran tres tarjetas informativas que resumen las funcionalidades principales del sistema: reserva de estacionamientos, seguridad automatizada mediante reconocimiento de patentes, y monitoreo en tiempo real (véase en Figura 4.8).

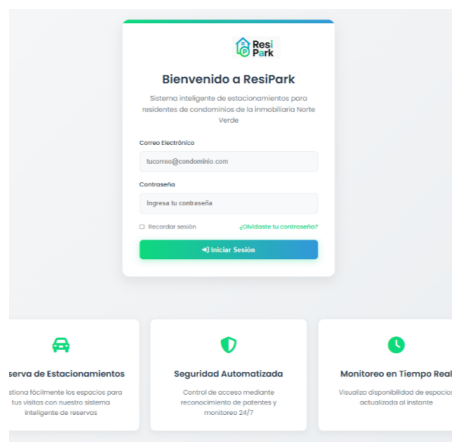


Figura 4.8: Pantalla de inicio de sesión, aplicación móvil.

El panel principal de la aplicación móvil despliega un saludo personalizado y tres accesos directos: (i) Reserva de Estacionamiento, para consultar disponibilidad y gestionar espacios para visitas; (ii) Notificaciones, para mantenerse informado sobre eventos del estacionamiento; y (iii) Mi Perfil, para actualizar la información personal, los vehículos registrados y las preferencias del usuario (véase en Figura 4.9).

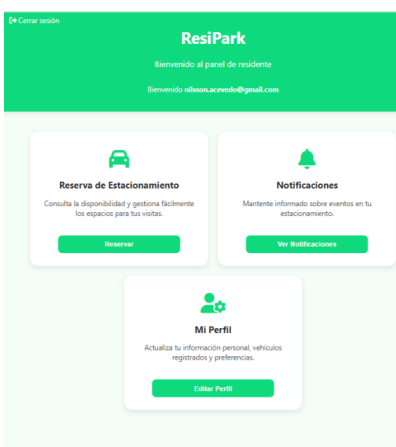


Figura 4.9: Panel principal de residente, aplicación móvil.

La interfaz de reserva de estacionamientos solicita, mediante un formulario, el nombre completo del visitante, su RUT, la marca y patente del vehículo, y un menú desplegable para seleccionar el estacionamiento deseado, confirmándose mediante el botón “Confirmar Reserva”. Bajo el formulario, una sección de “Disponibilidad de Estacionamientos” muestra en tarjetas individuales el estado (ocupado o disponible) de cada espacio, permitiendo al usuario elegir directamente entre las opciones libres (véase en Figura 4.10).

Reservar Estacionamiento
Reservar un espacio para tu visita

Formulario de Reserva

Nombre del visitante
Nombre completo

RUT del visitante
12345678-9

Marca del vehículo
Ej: Toyota, Chevrolet,...

Patente del vehículo
Ej: A81234

Estacionamiento a reservar
-- Selecciona un espacio --

Confirmar Reserva

Disponibilidad de Estacionamientos
Selecciona un espacio disponible para reservar

E1 Ocupado	E2 Disponible	E3 Disponible	E4 Disponible	E5 Ocupado	E6 Disponible
E7 Disponible	E8 Disponible				

Figura 4.10: Interfaz de reserva de estacionamientos, aplicación móvil.

Capítulo 5

Resultados

El presente capítulo expone los resultados obtenidos durante la fase de pruebas y despliegue del sistema ResiPark, con el propósito de evaluar su desempeño tanto en condiciones controladas como en un entorno de uso real.

5.1. Caso de Estudio

Con el objetivo de validar el funcionamiento del sistema ResiPark en un entorno residencial real, se realizó un despliegue piloto en un pasaje cerrado ubicado en la comuna de Cerrillos, Región Metropolitana. El recinto corresponde a una comunidad residencial de acceso cerrado, compuesta por entre 10 y 20 viviendas, cuyo único punto de ingreso vehicular y peatonal es un portón operado tradicionalmente mediante control remoto de radiofrecuencia (RF) de accionamiento manual, sin la presencia de personal de conserjería permanente ni ningún mecanismo digital de registro de accesos.

Este perfil de recinto resulta representativo del modelo de control de acceso predominante en comunidades residenciales chilenas, caracterizado en el Capítulo 3: dependencia de dispositivos RF entregados a los residentes, ausencia de registro sistemático de accesos y nula trazabilidad ante eventos de seguridad. A diferencia de los condominios de mayor escala, los pasajes cerrados de menor tamaño carecen por completo de presupuesto o infraestructura para adoptarlas, quedando excluidos de la transición hacia modelos de gestión inteligente. El despliegue de ResiPark en este contexto permite, por lo tanto, evaluar el sistema precisamente en el segmento que representa la brecha identificada: recintos pequeños, sin personal de seguridad dedicado y fuera del alcance práctico de las soluciones existentes.

5.1.1. Evaluación de tecnologías

Con base en lo revisado en el Capítulo [4](#) se realiza la evaluación de las tecnologías seleccionadas mediante la **Matriz de Decisión Ponderada**, la cual es una herramienta analítica utilizada para evaluar y comparar múltiples alternativas de forma objetiva. Su propósito principal es facilitar la toma de decisiones complejas al asignar valores numéricos a criterios subjetivos, lo que permite una comparación cuantitativa y fundamentada. Este método reduce la subjetividad al forzar al equipo a definir explícitamente qué es más importante a través de la asignación de pesos a diferentes criterios de evaluación. La matriz de decisión ponderada se compone de los siguientes elementos:

- **Alternativas:** Son las diferentes opciones o soluciones que se están considerando (por ejemplo, FastAPI, Express.js, etc.). Generalmente se listan como columnas.
- **Criterios de Evaluación:** Son los atributos, características o requisitos que se utilizarán para juzgar las alternativas (por ejemplo, Rendimiento, Escalabilidad, Costo). Se listan como filas.
- **Peso del Criterio (%)**: Representa la importancia relativa de cada criterio en la decisión final. A cada criterio se le asigna un porcentaje, y la suma de todos los pesos debe ser igual a 100 % (o 1.0).
- **Puntuación:** Es la calificación que recibe cada alternativa para un criterio específico, típicamente en una escala predefinida (por ejemplo, de 1 a 10). Esta puntuación evalúa qué tan bien una alternativa cumple con un criterio.
- **Puntuación Ponderada:** Es el resultado de multiplicar la *Puntuación* de una alternativa por el *Peso del Criterio*. Este valor refleja la calificación de la alternativa ajustada por la importancia del criterio.

5.1.2. Metodología de Cálculo

El proceso para utilizar la matriz consiste en, primero, listar todas las alternativas viables a comparar; luego, definir los criterios de evaluación más relevantes para los objetivos del proyecto y asignar a cada uno un peso porcentual según su prioridad; posteriormente, puntuar cada alternativa frente a cada criterio; y, finalmente, calcular la puntuación ponderada resultante para cada celda de la matriz (véase en Figura [5.1](#)).

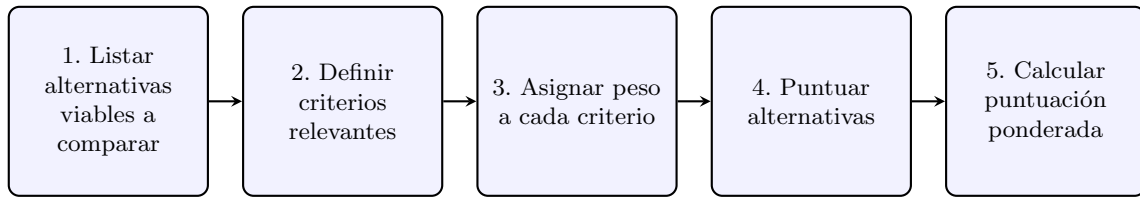


Figura 5.1: Diagrama de flujo de la metodología de cálculo para la matriz de decisión.

La fórmula para obtener la puntuación ponderada es:

$$PM = P_s \times W_c \quad (5.1)$$

donde PM corresponde a la puntuación ponderada, P_s a la puntuación asignada, y W_c al peso del criterio.

Finalmente, para obtener el resultado final de cada alternativa, se suman todas sus puntuaciones ponderadas:

$$P_a = \sum_{i=1}^n PM_i \quad (5.2)$$

donde P_a corresponde a la puntuación total de la alternativa, y PM_i a cada una de sus puntuaciones ponderadas individuales.

La alternativa con la puntuación total más alta constituye, bajo los criterios y ponderaciones definidos, la opción mejor evaluada.

5.1.3. Evaluación de Opciones por Componente

A continuación se presentan los cinco análisis comparativos realizados mediante la Matriz de Decisión Ponderada, correspondientes a cada componente tecnológico crítico del sistema: hardware de procesamiento en el borde, modelos de detección de objetos, tecnología de sensores, protocolo de comunicación IoT y sistema de gestión de base de datos.

5.1.3.1. Hardware para Procesamiento en el Borde

Para seleccionar la plataforma de hardware más adecuada para el procesamiento en el borde (*Edge Computing*), se compararon tres alternativas mediante una Matriz de Decisión Ponderada: **NVIDIA Jetson Nano** [84], **Raspberry Pi 4** [85] y una solución híbrida **Raspberry Pi + Google Coral**

USB [86]. Se definieron cuatro criterios: **Compatibilidad con CUDA (40 %)**, dado el beneficio directo del ecosistema NVIDIA; **Costo (30 %)**; **Consumo Energético (20 %)**; y **Facilidad de Uso (10 %)**. Cada alternativa fue puntuada de 1 a 10 por criterio (véase en Tabla 5.1).

Tabla 5.1: Matriz de decisión ponderada para hardware de procesamiento en el borde

Alternativa	Puntuación Total (P_a)
NVIDIA Jetson Nano	7,00
Raspberry Pi 4B	8,00
Raspberry Pi + Coral USB	7,20

La Tabla 5.1 muestra que la **Raspberry Pi 4B** obtiene la mayor puntuación total con un valor de 8,00. Si bien la NVIDIA Jetson Nano presenta una ventaja significativa en compatibilidad con CUDA, esta alternativa obtiene una valoración inferior en costo y consumo energético. Por su parte, la solución basada en Raspberry Pi y Coral USB destaca por su eficiencia energética, pero presenta una mayor complejidad de integración. En consecuencia, la **Raspberry Pi 4B** se selecciona como plataforma de procesamiento en el borde debido al equilibrio alcanzado entre costo, consumo energético, facilidad de uso y capacidades de procesamiento requeridas por el sistema.

5.1.3.2. Modelos de Detección de Objetos

Se compararon seis modelos de detección orientados a dispositivos embebidos: YOLOv5n [87], YOLOv8s [88], YOLOv4-Tiny [89], SSD-MobileNet v2 [90], OpenCV + Haar/Cascade [91] y YOLOv11s [92], mediante los criterios: **Precisión (30 %)**, **Requisitos Computacionales (20 %)**, **Velocidad de Procesamiento (15 %)**, **Adaptabilidad (15 %)**, **Facilidad de Entrenamiento (10 %)**, **Licencia/Costo (5 %)** y **Tiempo de Implementación (5 %)** (véase en Tabla 5.2).

Tabla 5.2: Puntuación ponderada total por modelo de detección de objetos

Modelo	Puntuación Total (P_a)
YOLOv11s	8,60
SSD-MobileNet v2	7,93
YOLOv5n	7,90
YOLOv8s	7,85
YOLOv4-Tiny	7,67
OpenCV + Haar/Cascade	6,25

YOLOv11s obtiene la mayor puntuación al combinar una precisión superior (mAP 47,0 frente a 44,9 de YOLOv8s) con menores requisitos computacionales, logrando el mejor equilibrio entre exactitud

y eficiencia. SSD-MobileNet v2 y YOLOv5n son alternativas livianas competitivas de bajo consumo. YOLOv8s obtiene una puntuación inferior por su mayor demanda computacional frente a las alternativas livianas. YOLOv4-Tiny sobresale en velocidad y bajo consumo con precisión moderada, mientras que OpenCV + Haar/Cascade, el más liviano, es el menos preciso y solo apto para casos básicos.

La elección final de modelos para el módulo de visión por computador (M1) combina ambos resultados según la subtarea: **YOLOv11s** se utilizó para la detección de patentes, consistente con su mejor puntuación; **YOLOv8s** se utilizó para la detección de vehículos, decisión motivada por la disponibilidad previa de un modelo YOLOv8 ya entrenado sobre un conjunto propio de más de 10.000 imágenes, sobre el cual se realiza un reentrenamiento (*fine-tuning*) utilizando otro conjunto de datos, implicando un menor costo que entrenar YOLOv11 desde cero para esta subtarea.

5.1.3.3. Tecnologías de Sensores para Detección Vehicular

Para detectar la ocupación de los estacionamientos de visita se compararon **sensor ultrasónico**, **sensor magnético**, **sensor de presión** y **sensor infrarrojo (IR)**, según: **Precisión (30 %)**, **Costo (25 %)**, **Funcionamiento en Subterráneos (25 %)** y **Facilidad de Instalación (20 %)** (véase en Tabla 5.3).

Tabla 5.3: Matriz de decisión ponderada para tecnologías de sensores

Alternativa	Puntuación Total (P_a)
Sensor ultrasónico	7,95
Sensor magnético	8,75
Sensor de presión	6,60
Sensor infrarrojo	6,90

Los sensores IR son económicos y fáciles de instalar, pero de precisión limitada y dependientes de la luz ambiental [93]. Los de presión son muy precisos pero de instalación compleja y costosa bajo la superficie [94]. Los **sensores magnéticos** combinan instalación sencilla, alta precisión y excelente desempeño en subterráneos [95], resultando la alternativa con mejor desempeño según los criterios definidos. Los sensores ultrasónicos ofrecen buena precisión y bajo costo, aunque son sensibles a la humedad y a los ecos en entornos cerrados [96]. No obstante, debido a las restricciones de costo y disponibilidad de componentes, se opta finalmente por el sensor ultrasónico.

5.1.3.4. Protocolos de Comunicación Inalámbrica para IoT

Se evaluaron **Wi-Fi** [97], **LoRaWAN** [98], **Zigbee** [99], **Thread** [100] y **BLE** [101] según: **Alcance (30 %)**, **Consumo Energético (25 %)**, **Facilidad de Implementación (20 %)**, **Seguridad Integrada (15 %)** y **Velocidad de Transmisión (10 %)** (véase en Tabla 5.4).

Tabla 5.4: Matriz de decisión ponderada para protocolos de comunicación inalámbrica

Alternativa	Puntuación Total (P_a)
Wi-Fi	7,60
LoRaWAN	8,25
Zigbee	7,55
Thread	7,50
BLE	7,65

Wi-Fi destaca en velocidad y facilidad de uso, pero con alto consumo y alcance limitado [97]. Zigbee y Thread ofrecen arquitectura mallada y seguridad integrada, a costa de mayor complejidad [99]. BLE tiene bajo consumo pero alcance insuficiente para una red dispersa [101]. **LoRaWAN** ofrece el mejor equilibrio entre largo alcance, bajo consumo y cifrado robusto (AES-128) [98], siendo la opción más adecuada para el proyecto.

5.1.3.5. Sistema de Gestión de Base de Datos (SGBD)

Se evaluaron siete SGBD: MySQL [102], PostgreSQL [103], MariaDB [104], SQLite [105], Oracle DB [106], MongoDB [107] y Firebase [108], en dos fases: una evaluación cualitativa general, y una Matriz de Decisión Ponderada sobre los finalistas tras descartar Oracle DB, SQLite, MongoDB y Firebase.

Los tres finalistas obtuvieron puntuaciones muy similares (véase en Tabla 5.5), indicando que cualquiera representa una elección técnicamente viable:

Tabla 5.5: Puntuación ponderada total por SGBD finalista

SGBD	Puntuación Total (P_a)
MySQL	8,7
MariaDB	8,6
PostgreSQL	8,5

A partir de los cinco análisis comparativos precedentes, la Tabla 5.6 consolida los componentes tecnológicos finalmente seleccionados para cada uno de los cuatro módulos del sistema.

Tabla 5.6: Resumen de componentes seleccionados por módulo

Módulo	Componentes seleccionados
M1 – Visión por Computador y OCR	YOLOv11s (detección de patentes); YOLOv8s (detección de vehículos, reentrenado sobre conjunto propio de más de 10.000 imágenes); Florence-2 (OCR)
M2 – Sensado	ESP32 + módulo LoRa + sensor ultrasónico de proximidad (ocupación de estacionamientos); Raspberry Pi 4B + cámara IP (captura de vehículos y patentes)
M3 – Interfaces de Usuario	React Native (app móvil: registro, reservas, historial); React.js, HTML5 y CSS3 (web de monitoreo y control para conserjería)
M4 – Servicios Centrales	Flask y FastAPI (Python), Express.js (Node.js); Supabase (base de datos SQL); GitHub (control de versiones); Google Colab (entrenamiento de modelos); Uvicorn (servidor local); CloudFlare (túnel público)

5.2. Módulos del Sistema

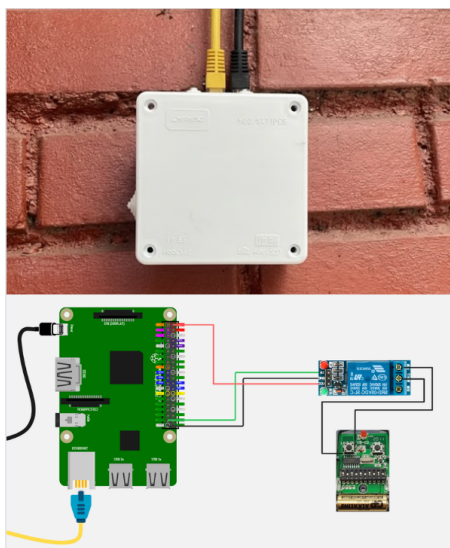
A partir de la selección tecnológica, esta sección describe la implementación concreta de los cuatro módulos que componen *ResiPark*: (i) el módulo de gestión de accesos vehiculares (M1), (ii) el módulo de sensado para la gestión de estacionamientos de visitantes (M2), (iii) las interfaces de usuario web y móvil (M3), y (iv) los servicios centrales que integran al sistema completo (M4). Para cada uno se detalla tanto su arquitectura de hardware como el flujo lógico que gobierna su funcionamiento.

5.2.1. Módulo 1 - Gestión de Accesos Vehiculares (M1):

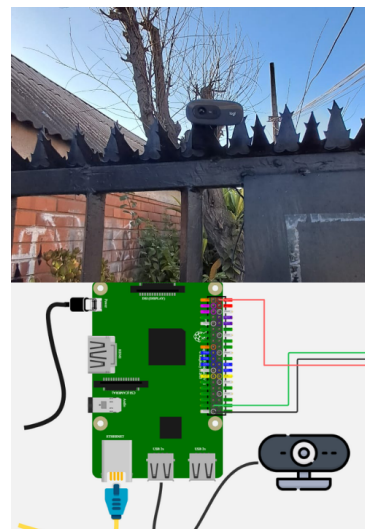
Este módulo se encarga de gestionar el ingreso y salida de vehículos del recinto residencial. Este módulo se apoya en tecnologías de visión por computador y reconocimiento automático de patentes para permitir o denegar accesos, en función de las autorizaciones registradas en el sistema.

5.2.1.1. Control de Acceso

El control de acceso se compone de dos integraciones principales sobre una Raspberry Pi 4B: (i) un **relé** de un canal encargado de accionar el control remoto del portón, y (ii) una **cámara IP/webcam** encargada de captar imágenes de los vehículos que ingresan. La Figura [5.2](#) muestra ambos componentes.



(a) Raspberry Pi 4B con relé de 1 canal para el control remoto del portón.



(b) Raspberry Pi 4B con cámara IP/webcam para la captura de imágenes de vehículos.

Figura 5.2: Componentes del módulo de control de acceso: (a) integración con relé para accionamiento del portón; (b) integración con cámara para captura de imágenes vehiculares.

Estos componentes son responsables de la recolección de información del sistema en cuestión, obteniendo así las imágenes de los vehículos presentes en el portón (véase en Figura 5.3).

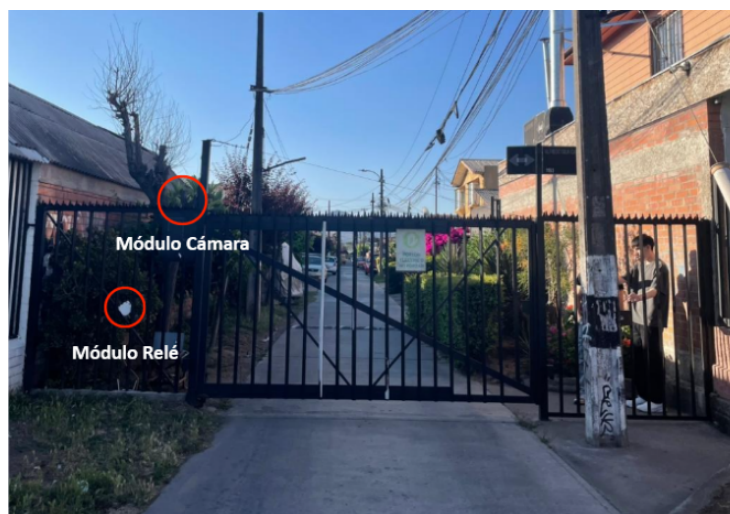


Figura 5.3: Entorno real de los módulos actuando en la recolección de información.

5.2.1.2. Servicio de Detección y Reconocimiento

Para el correcto funcionamiento del control de acceso vehicular, se implementa un pipeline de cinco etapas sobre cada imagen o fotograma recibido (véase en Figura 5.4):

1. **Definición de zona de interés (ROI):** se delimita una región rectangular en la porción central-inferior del fotograma, correspondiente al área donde habitualmente circulan los vehículos frente a la cámara. El margen horizontal se fija en un 10 % a cada lado (cubriendo el 80 % central del ancho de la imagen), mientras que verticalmente la zona abarca desde el 40 % hasta el 90 % de la altura del fotograma, descartando detecciones fuera de esta zona.
2. **Detección vehicular:** el modelo YOLOv8s procesa el fotograma completo; solo se conservan las detecciones cuya caja delimitadora intersecta la zona de interés con un área mínima configurable, evitando procesar vehículos en tránsito por el fondo de la escena.
3. **Detección de patente:** sobre el recorte del vehículo válido, el modelo YOLOv11s localiza la región de la patente, seleccionando la detección de mayor confianza y aplicando un margen de padding del 8 % para evitar recortes ajustados que afecten el reconocimiento posterior.
4. **Transcripción OCR:** el recorte de la patente se procesa mediante Florence-2 con la tarea <OCR>, generando el texto transcrito.
5. **Validación heurística:** el texto generado se normaliza (mayúsculas, remoción de espacios) y se valida mediante una expresión regular que exige una longitud de 5 a 8 caracteres alfanuméricos con al menos una letra y un dígito, descartando transcripciones inválidas antes de retornar el resultado.

El servicio retorna, para cada detección válida, el tipo de vehículo, las confianzas de detección vehicular y de patente, y el texto de la patente transcrita, en formato JSON.

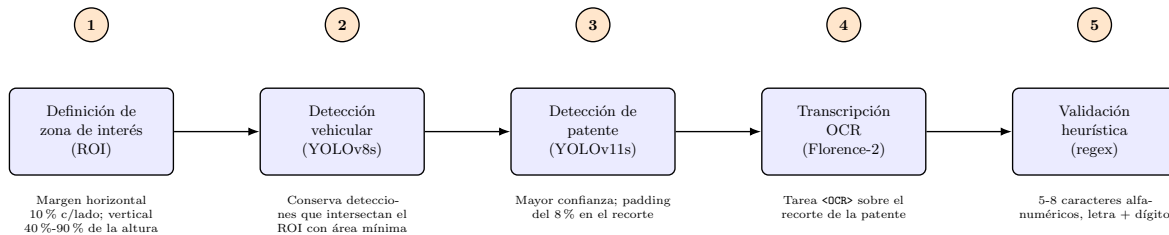


Figura 5.4: Pipeline de cinco etapas del endpoint `/predict`: desde la definición de la zona de interés hasta la validación heurística del texto transcrito.

5.2.2. Módulo 2 - Gestión de Estacionamientos de Visitantes (M2):

Permite monitorear y administrar el uso de los estacionamientos destinados a visitas. Utiliza una red de sensores IoT para detectar la presencia de vehículos en tiempo real. El sistema registra el tiempo de ocupación de cada espacio y genera alertas en caso de infracciones o uso prolongado no autorizado, facilitando así una gestión eficiente y transparente de estos recursos compartidos.

5.2.2.1. Módulo Monitoreo Estacionamientos

Para la gestión de estacionamientos de visita, se utiliza un sensor ultrasónico en conjunto con un microcontrolador ESP32 (véase en Figura [5.5](#)). El sensor tiene una capacidad máxima de **60 cm** de detección, permitiendo así un correcto funcionamiento a la hora de monitorear los estacionamientos.

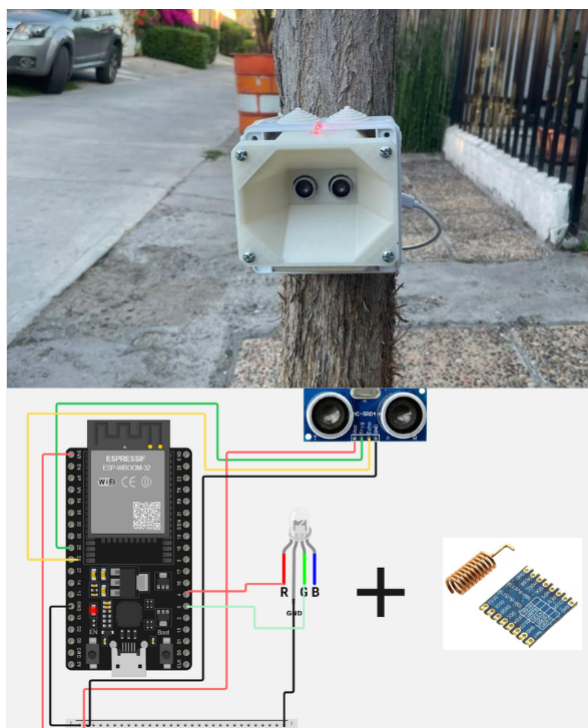


Figura 5.5: Sensor ultrasónico integrado con módulo de radio LoRa y ESP32 para la detección de estacionamientos.

A nivel de hardware, la comunicación LoRaWAN se implementa mediante un módulo de radio LoRa SX1276/78 integrado al microcontrolador ESP32. La Figura [5.6](#) detalla el flujo completo de comunicación del sensor de estacionamiento.

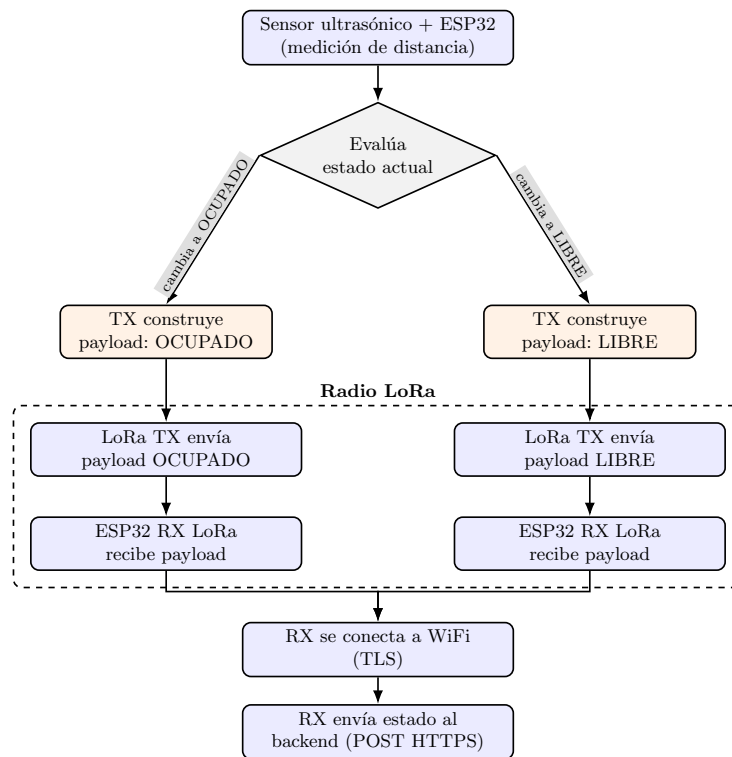


Figura 5.6: Flujo de comunicación del sensor de ocupación de estacionamientos: el ESP32 transmisor evalúa el estado a partir del sensor ultrasónico, construye payload correspondiente (libre/ocupado) y lo transmite vía LoRa; el ESP32 receptor recibe el payload, se conecta a la red WiFi mediante TLS y reenvía el estado al backend del sistema mediante una solicitud HTTPS.

5.2.3. Módulo 3 - Interfaces de Usuario (M3):

Incluye las aplicaciones web y móviles destinadas a los distintos perfiles de usuarios. A través de estas interfaces, residentes, conserjes y administradores podrán interactuar con el sistema, realizar gestiones y visualizar información relevante.

5.2.3.1. Interfaz web

Para el desarrollo de la aplicación web se utiliza una API REST desarrollada con FastAPI, encargada de gestionar el intercambio de información entre la interfaz de usuario, la base de datos y los servicios centrales del sistema.

Inicio de Sesión

Para la pestaña de ingreso de los usuarios (véase en Figura 5.7) se solicitan el RUT del conserje y la contraseña. En caso de tener problemas con la contraseña, se dispone de un link de recuperación.

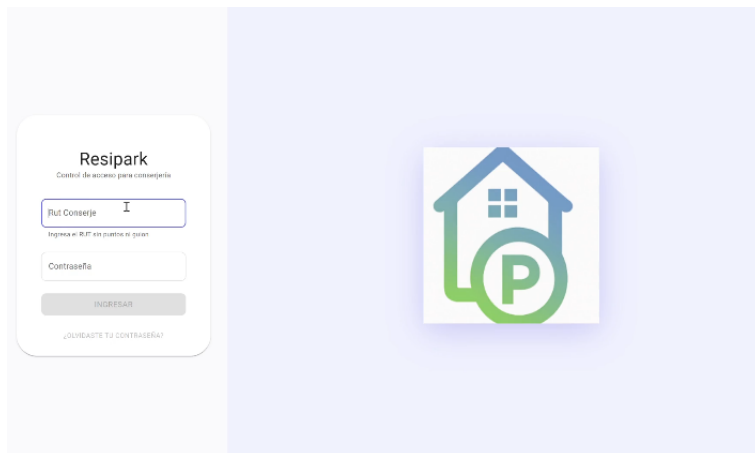


Figura 5.7: Pestaña de ingreso

Pantalla de Inicio

La pantalla de inicio (véase en Figura 5.8) ofrece una vista centralizada de las operaciones principales del estacionamiento. En el lado izquierdo se muestra la cámara de entrada, lo que permite al conserje visualizar en tiempo real los accesos al recinto. En el centro se encuentra la grilla de disponibilidad de estacionamientos, organizada con un sistema de colores que distingue entre espacios libres, ocupados o reservados, acompañados de la hora desde la cual están disponibles. A la derecha se habilita un módulo para gestionar reservas, con un botón destacado que permite liberar un estacionamiento en caso de ser necesario.

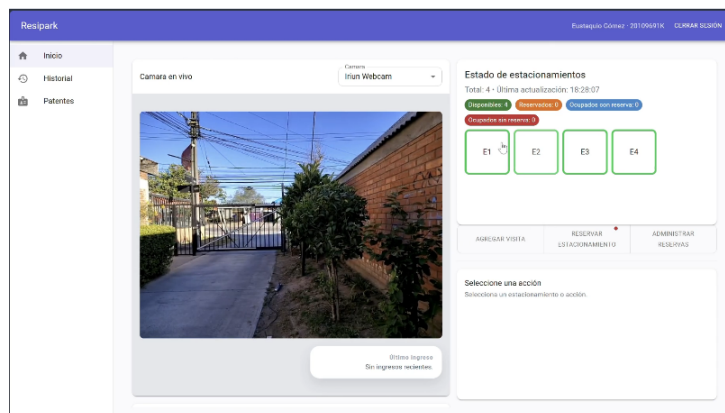


Figura 5.8: Pestaña de inicio

Pantalla de Historial

Esta pantalla corresponde al módulo de historial, donde el conserje puede revisar los registros de acceso al estacionamiento (véase en Figura 5.9). Se muestra una tabla con columnas para la patente del vehículo, el tipo de usuario, la hora de entrada y la hora de salida. En la parte superior se incluye una barra de búsqueda que facilita la localización de registros específicos. Esta interfaz permite llevar un control ordenado y detallado de los movimientos de los vehículos que ingresan y salen de la comunidad residencial.

Patente	Tipo Usuario	Depto	Entrada	Salida	Evento
879F45	Visitante	202A	09-11-2023 18:20:07	-	Ingreso

Figura 5.9: Pestaña historial de ingresos

Administración de Patentes

En esta pantalla, el conserje puede gestionar las patentes registradas en el sistema, asociándolas a cada vivienda/departamento de la comunidad residencial. Se presenta una tabla con dos columnas: el número del departamento y la patente vinculada. Además, existe la opción de editar los registros directamente mediante un ícono de lápiz en cada fila. Este módulo está orientado a mantener un control actualizado de los vehículos autorizados para ingresar al estacionamiento (véase en Figura 5.10).

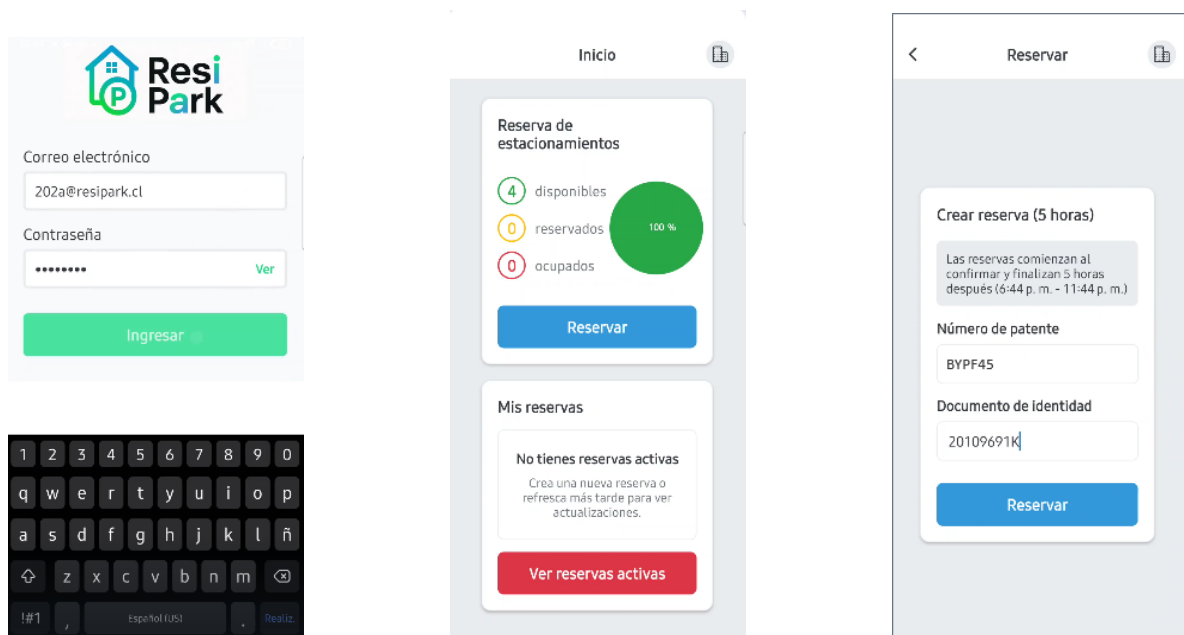
Departamento	Patente	Tipo
201A	KW0508	Auto
201B	BK183	Auto
202A	JJ0048	Auto
202B	GRK255	Auto
301A	HZP649	Auto
301B	PLJ538	Auto
302A	JTH842	Auto
302B	HCCL44	Auto

Figura 5.10: Pestaña Administración de Patentes

5.2.3.2. Aplicación móvil

Para el desarrollo de la aplicación móvil se utiliza una API REST desarrollada con FastAPI, encargada del intercambio de información correspondiente al estado de los estacionamientos.

La interfaz de inicio de sesión tiene como propósito autenticar a los usuarios residentes de forma segura. Se presentan campos para ingresar correo electrónico y contraseña, junto a un botón de inicio de sesión. La vista principal de reserva presenta un resumen del estado de los estacionamientos de visita, indicando la cantidad de espacios disponibles, reservados y ocupados, junto con un gráfico circular que representa dicha proporción. Bajo este resumen, un botón permite iniciar una nueva reserva, mientras que la sección “Mis reservas” muestra las reservas activas del residente o, en su ausencia, un mensaje indicando que no existen reservas vigentes, junto a un botón para consultar el detalle de las reservas activas. La interfaz para agendar un estacionamiento solicita el número de patente del vehículo visitante y el documento de identidad asociado a la reserva, indicando explícitamente que esta tendrá una duración fija de cinco horas a partir del momento de su confirmación. Una vez completados los campos, el botón “Reservar” concreta la solicitud (véase en Figura 5.11).



(a) Pantalla de inicio de sesión.

(b) Vista residente para agendar estacionamiento.

(c) Interfaz para agendar estacionamiento.

Figura 5.11: Flujo de reserva de estacionamientos en la aplicación móvil implementada: (a) inicio de sesión, (b) panel principal del residente, (c) formulario de reserva.

5.2.4. Módulo 4 - Servicios Centrales del Sistema (M4):

Este módulo corresponde al núcleo de integración tecnológica del sistema. Incluye las API, la base de datos y la lógica de negocio que permiten que los distintos componentes funcionen de manera coherente y centralizada, actuando como intermediario entre los módulos de percepción, gestión e interfaces.

5.2.4.1. Servicio de Verificación y Registro

El servicio de detección se implementa como una API desarrollada en FastAPI, servida mediante Uvicorn como servidor ASGI en un hilo de ejecución independiente, alojada en un entorno con aceleración por GPU (Google Colab) y expuesta públicamente mediante un túnel Cloudflare, lo que permite su consumo remoto desde el dispositivo de borde sin requerir una dirección IP pública fija. Durante el arranque del servicio (*lifespan*), se cargan en memoria dos modelos de detección – YOLOv8s para la localización de vehículos y YOLOv11s para la localización de patentes – junto con el modelo Florence-2 para transcripción óptica de caracteres (OCR) (véase en Figura 5.12).

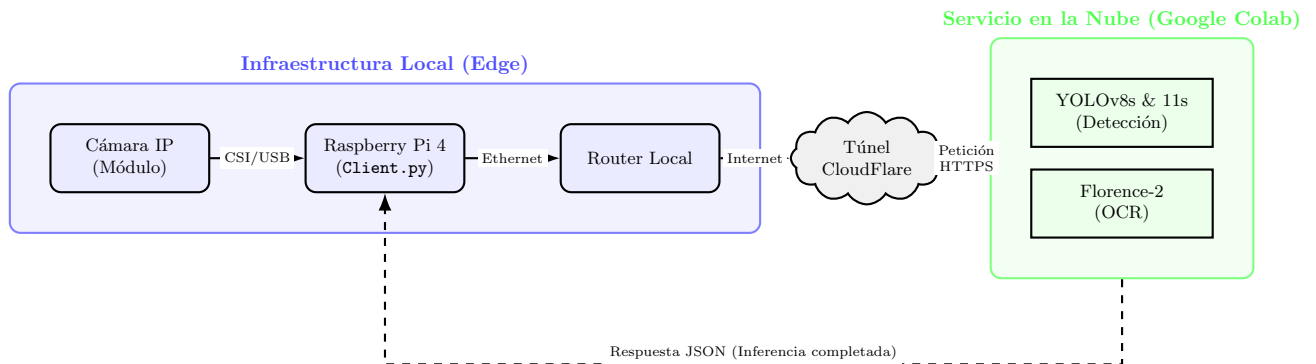


Figura 5.12: Arquitectura del sistema de control de acceso: El flujo separa la recolección local de imágenes (Edge) del procesamiento de redes neuronales profundas en la nube, ilustrando el ciclo completo de petición y respuesta.

El segundo componente es una API REST desarrollada en FastAPI con SQLAlchemy como capa de acceso a datos, responsable de contrastar cada patente detectada contra la base de datos del sistema y registrar los eventos de ingreso autorizados. Expone tres endpoints principales, cuya lógica de negocio se resume en la Tabla 5.7

Tabla 5.7: Endpoints del servicio de verificación y registro

Endpoint	Método	Lógica de negocio
<code>/verificar-patente/{patente}/{tipo}</code>	GET	Busca la patente en la tabla de vehículos residentes de la BD Supabase con coincidencia exacta de tipo; si no coincide el tipo, informa el conflicto; si no es residente, busca una reserva de visita activa ($\text{hora_inicio} \leq \text{ahora} \leq \text{hora_termino}$) asociada a esa patente.
<code>/registrar-ingreso-residente/{patente}</code>	POST	Confirma que la patente corresponde a un residente e inserta un evento de acceso en <code>registro_evento_acceso</code> , asociado directamente al vehículo.
<code>/registrar-ingreso-visitante/{patente}</code>	POST	Confirma la existencia de una reserva activa e inserta un evento de acceso asociado al identificador de dicha reserva, en lugar de a un vehículo registrado permanentemente.

Esta diferenciación entre residentes y visitantes con reserva – resuelta mediante dos rutas de verificación distintas dentro del mismo endpoint – permite que el sistema aplique reglas de autorización diferenciadas sin requerir que el cliente conozca de antemano la categoría del vehículo detectado.

5.2.4.2. Orquestador de Borde

El tercer componente, ejecutado en la Raspberry Pi, coordina el flujo completo del sistema (véase en Figura 5.13): (i) captura fotogramas de la cámara IP a intervalos regulares (cada diez fotogramas, para limitar la carga sobre el servicio de detección), (ii) los envía al servicio de detección y reconocimiento, y (iii) para cada patente transcrita, invoca secuencialmente el servicio de verificación y – de resultar válida – el de registro de ingreso. La apertura física del portón es delegada a un módulo de control de relé invocado por el orquestador una vez confirmado el registro exitoso del ingreso.

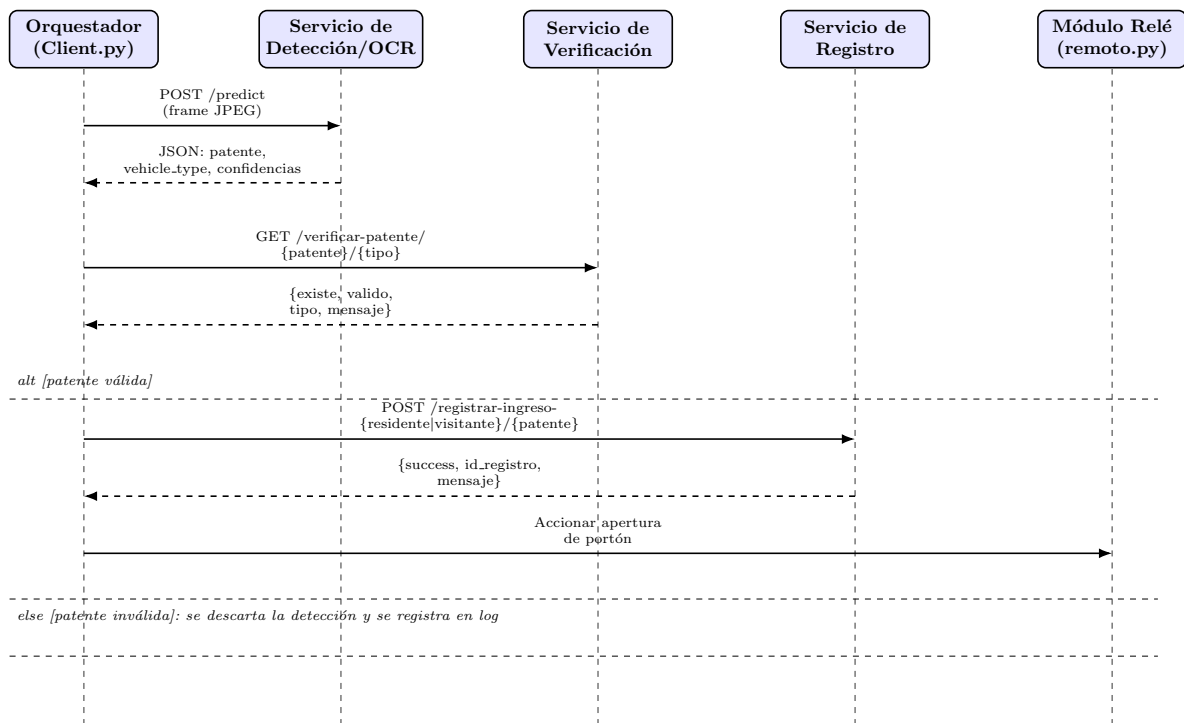


Figura 5.13: Diagrama de secuencia del flujo de comunicación en la Capa de Soporte: desde la captura del fotograma hasta la apertura del portón.

5.3. Revisión del Prototipo

Esta sección presenta una revisión general del prototipo implementado, describiendo sus funcionalidades principales, la arquitectura de componentes que lo conforman y el rol específico que cumple cada uno dentro del flujo completo del sistema.

5.3.1. Contexto General del Prototipo

El prototipo permite la captura de video en tiempo real desde cámaras, el procesamiento en Colab de modelos de detección para patentes y tipo de vehículo, y la validación de dichos resultados contra una base de datos SQL. Asimismo, habilita la reserva de estacionamientos por parte de los residentes a través de una interfaz web y una aplicación móvil, y el monitoreo de la ocupación de los estacionamientos de visita mediante sensores ESP32 con radio LoRa.

5.3.2. Esquema General de Componentes del Prototipo

El prototipo (véase en Figura 5.14) se compone de una cámara IP con comunicación directa al Backend; una API REST desarrollada en FastAPI que gestiona la comunicación con Colab y con la base de datos; aplicaciones web y móvil que permiten a los usuarios visualizar y gestionar accesos; y un sensor IoT para el monitoreo de estacionamientos, conectado al Backend.

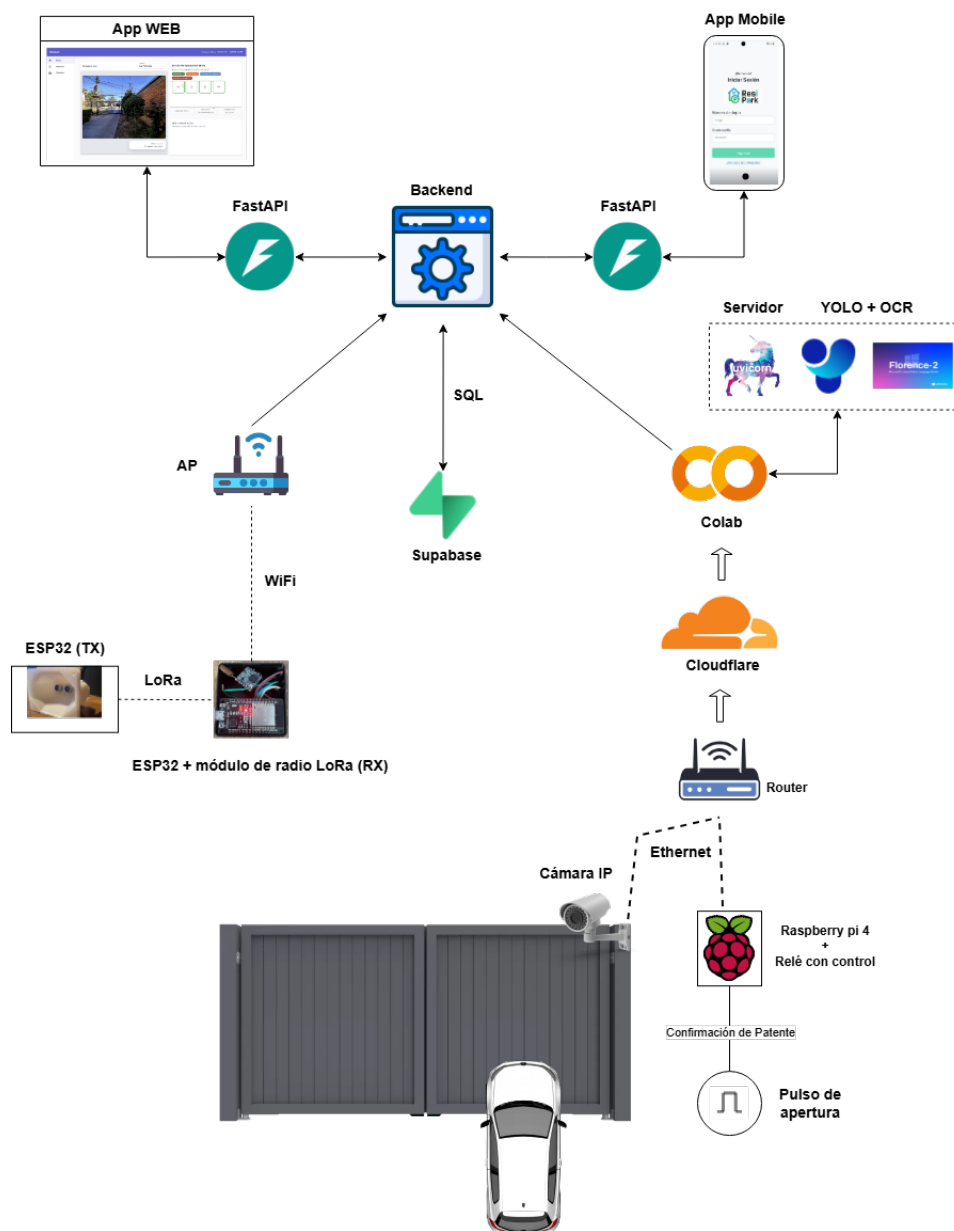


Figura 5.14: Esquema general prototipo

5.3.3. Descripción de Componentes

La Figura 5.14 presenta el esquema general del prototipo implementado. La Tabla 5.8 describe el rol de cada componente señalado en dicho esquema.

Tabla 5.8: Descripción de componentes del esquema general del prototipo

Componente	Descripción / Rol en el sistema
App WEB	Interfaz destinada a conserjería: monitoreo de cámara de entrada, grilla de disponibilidad de estacionamientos, historial de accesos y administración de patentes. Se comunica con el Backend a través de su propia instancia de FastAPI.
App Mobile	Interfaz destinada a residentes: inicio de sesión, reserva de estacionamientos de visita, historial y gestión de reservas activas. Se comunica con el Backend a través de su propia instancia de FastAPI.
FastAPI (Web / Mobile)	Dos instancias de la API REST que exponen los endpoints consumidos por la aplicación web y la aplicación móvil respectivamente, actuando como capa de comunicación entre las interfaces de usuario y el Backend.
Backend	Núcleo de lógica de negocio del sistema: procesa las solicitudes provenientes de ambas interfaces, coordina la verificación y registro de accesos, y gestiona la comunicación con la base de datos, el módulo de sensado y el servicio de detección en la nube.
Supabase	Base de datos SQL del sistema; almacena usuarios, vehículos, reservas y el registro de eventos de acceso, comunicándose con el Backend mediante consultas SQL.
AP (Access Point)	Punto de acceso WiFi que recibe la información del nodo receptor ESP32+LoRa y la reenvía al Backend.
ESP32 (TX)	Nodo emisor equipado con sensor ultrasónico de proximidad (rango máximo 60 cm); detecta la ocupación del estacionamiento de visitas y transmite el estado (libre/ocupado) vía LoRa.
ESP32 + módulo de radio LoRa (RX)	Nodo receptor que recibe el payload LoRa transmitido por el ESP32 (TX) y lo reenvía al AP mediante WiFi.

Componente	Descripción / Rol en el sistema
Servidor – YOLO + OCR	Servicio desplegado sobre Uvicorn que aloja los modelos de detección (YOLOv8s para vehículos, YOLOv11s para patentes) y el modelo Florence-2 para la transcripción OCR, ejecutando el pipeline completo de reconocimiento sobre cada fotograma recibido.
Colab	Entorno de ejecución en la nube con aceleración por GPU, sobre el cual se aloja el Servidor YOLO + OCR.
Cloudflare	Túnel público que expone el servicio alojado en Colab hacia Internet, permitiendo su consumo remoto desde el dispositivo de borde sin requerir una dirección IP pública fija.
Router	Dispositivo de red local que conecta la Raspberry Pi (control de acceso) a Internet, habilitando la comunicación con el túnel Cloudflare.
Cámara IP	Captura en tiempo real las imágenes de los vehículos que se aproximan al portón, transmitiéndolas a la Raspberry Pi mediante conexión Ethernet.
Raspberry Pi 4 + Relé con control	Ejecuta el orquestador de borde: captura fotogramas de la cámara, los envía al Servidor YOLO + OCR, y –tras la confirmación de patente– acciona el relé para generar el pulso de apertura del portón.
Confirmación de Patente / Pulso de apertura	Señal generada por la Raspberry Pi hacia el relé una vez validado el ingreso, que se traduce en el pulso eléctrico que acciona la apertura física del portón.

5.3.4. Funcionamiento del sistema

Esta sección describe el funcionamiento del sistema a través de las vistas implementadas, organizadas según el perfil de usuario que las opera. En primer lugar, se presenta el flujo de conserjería a través de la interfaz web; luego, el flujo de reserva desde la aplicación móvil de residentes; y finalmente, el ciclo de asignación y ocupación de un estacionamiento, que involucra la interacción conjunta entre ambos perfiles y la red de sensores IoT.

5.3.4.1. Flujo de Conserjería (Interfaz Web)

El conserje inicia sesión en la interfaz web mediante sus credenciales de acceso (véase en Figura 5.15). Esta pantalla constituye el punto de entrada al sistema para el personal de conserjería y administración, restringiendo el acceso a las funcionalidades de monitoreo y gestión únicamente a usuarios autenticados.

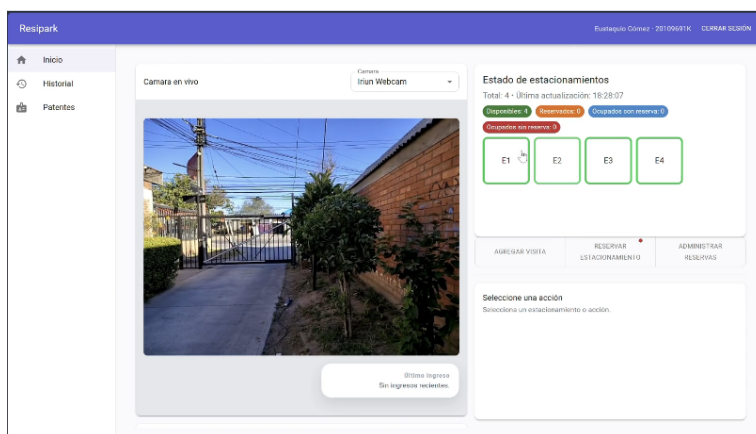


Figura 5.15: Pestaña de inicio, interfaz web.

Una vez autenticado, el conserje puede registrar manualmente el ingreso de una nueva visita al sistema, en aquellos casos en que un vehículo no cuenta con una reserva previa realizada por el residente (véase en Figura 5.16). Esta funcionalidad permite mantener un registro centralizado de todos los accesos, incluso aquellos gestionados de forma manual, sin comprometer la trazabilidad del sistema.

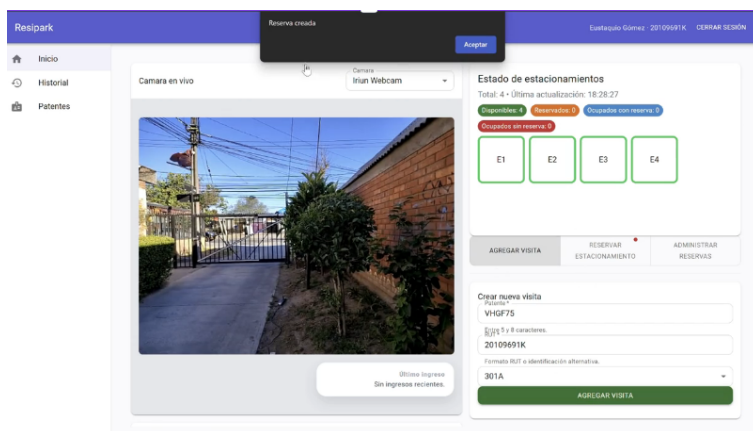


Figura 5.16: Registro de visita, interfaz web.

5.3.4.2. Flujo de Reserva (Aplicación Móvil de Residentes)

Un residente desea realizar una reserva de estacionamiento de visitas. Una vez autenticado, el usuario accede al menú principal, desde donde puede visualizar opciones como reservar estacionamiento, visualizar la disponibilidad en tiempo real de los estacionamientos, ver historial de reservas, gestionar datos personales y cerrar sesión. La interfaz está diseñada para facilitar la navegación, manteniendo la información clara y accesible. El usuario selecciona la opción “Reservar”, y posteriormente indica la patente y el RUT de la visita. Este flujo de tres pasos – inicio de sesión, selección desde el panel principal y llenado del formulario de reserva – resume la interacción mínima que debe completar un residente para habilitar el acceso de un visitante (véase en Figura 5.17).

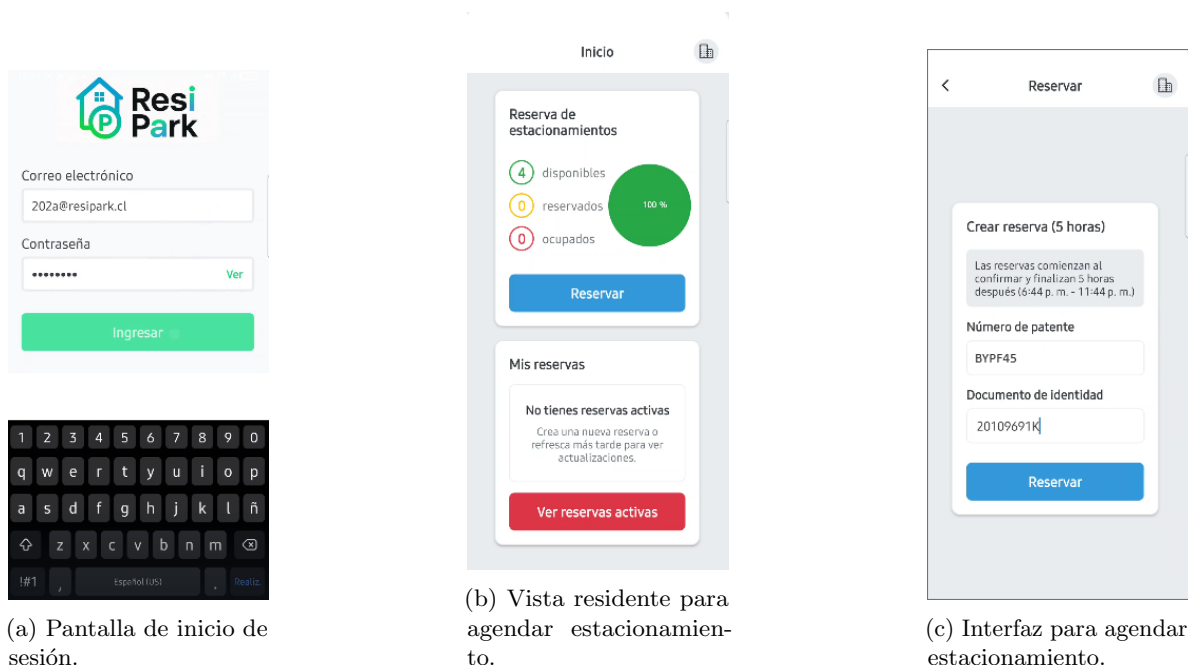


Figura 5.17: Flujo de reserva de estacionamientos en la aplicación móvil implementada: (a) inicio de sesión, (b) panel principal del residente, (c) formulario de reserva.

Al confirmar la reserva, el sistema despliega una ventana emergente de confirmación y el usuario puede visualizar sus reservas activas directamente desde la aplicación. Desde esta misma vista, el residente también tiene la posibilidad de anular una reserva vigente si la visita ya no será necesaria. Posteriormente, en la pantalla de inicio se actualizan las reservas activas y el número de estacionamientos disponibles, reflejando el estado más reciente del sistema (véase en Figura 5.18).

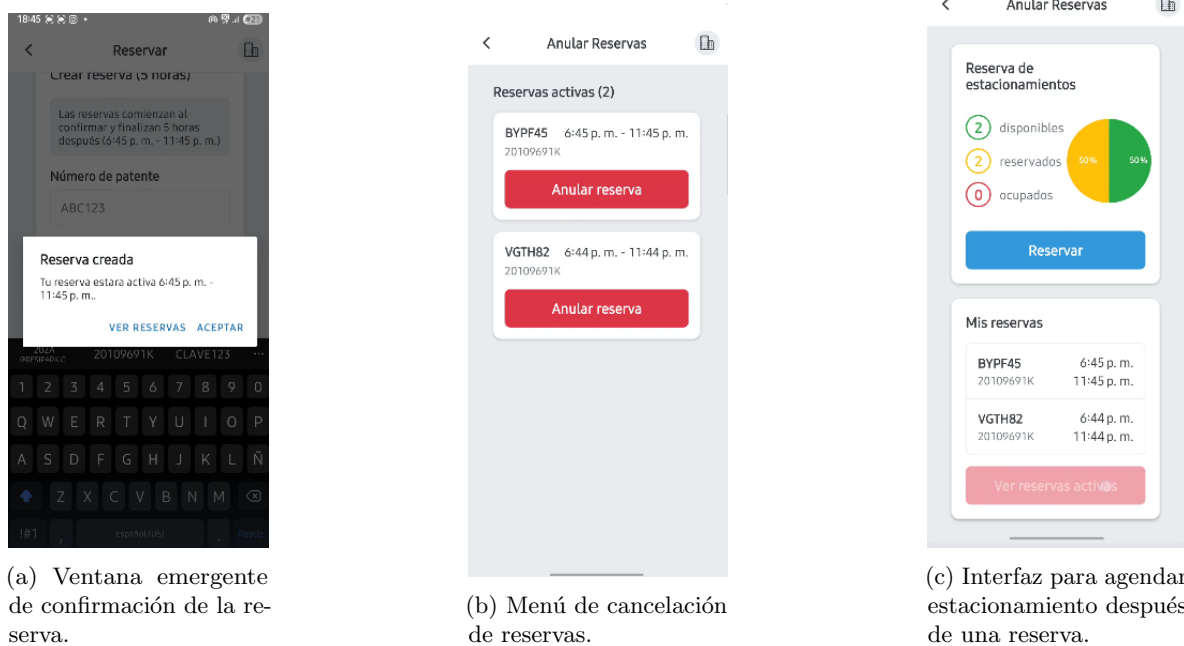


Figura 5.18: Confirmación, cancelación y estado posterior de una reserva de estacionamiento: (a) ventana emergente de confirmación, (b) menú de cancelación de reservas activas, (c) panel principal del residente tras registrar una reserva.

5.3.4.3. Ciclo de Asignación y Ocupación de Estacionamientos

Una vez que el residente ha generado la reserva, el conserje debe asignar manualmente un estacionamiento disponible a dicha solicitud desde la interfaz web (véase en Figura 5.19). Esta etapa constituye el punto de enlace entre el flujo de reserva del residente y el flujo de acceso vehicular gestionado por conserjería.

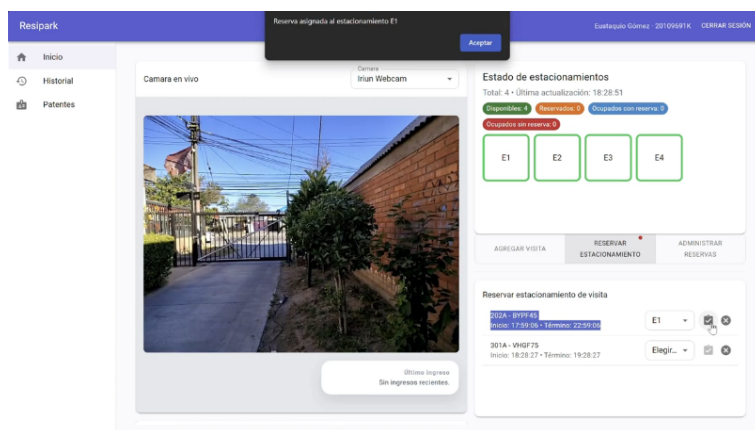


Figura 5.19: Asignación de estacionamientos, interfaz web.

Al completarse la asignación, el estado del estacionamiento cambia de disponible (color verde) a reservado (color amarillo) en la interfaz de conserjería. Cuando el vehículo visitante se presenta en el portón y es reconocido por el pipeline de detección, el sistema despliega en tiempo real la patente detectada, el tipo de acceso (visita) y la hora de ingreso, accionando de manera automática el relé para abrir el portón en función de la reserva activa (véase en Figura 5.20).

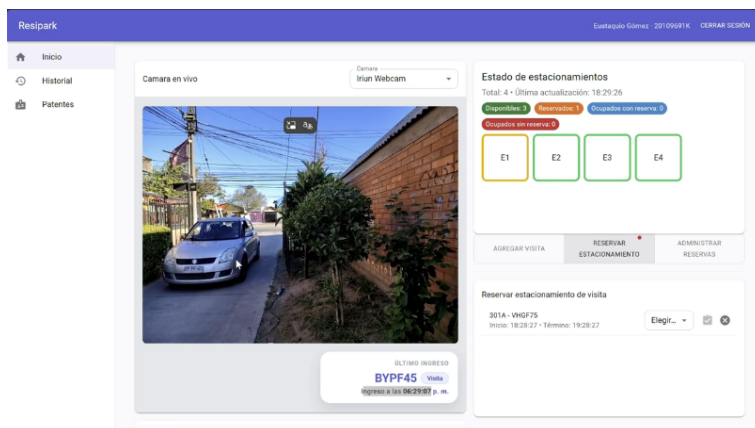


Figura 5.20: Estado y visualización de acceso vehicular, interfaz web.

Posteriormente, cuando el vehículo se estaciona físicamente en el espacio asignado – en este caso, el estacionamiento E1 – el sensor ultrasónico detecta su presencia y enciende el LED rojo correspondiente en el módulo de sensado. Esta señal es transmitida vía LoRa hacia el backend, actualizando el estado del estacionamiento a color azul en la interfaz web, indicando que se encuentra ocupado con una reserva activa asociada (véase en Figura 5.21). Este cambio de estado evidencia la sincronización entre la información lógica de la reserva (gestionada por el residente y el conserje) y la ocupación física real del espacio (reportada por el sensor IoT).

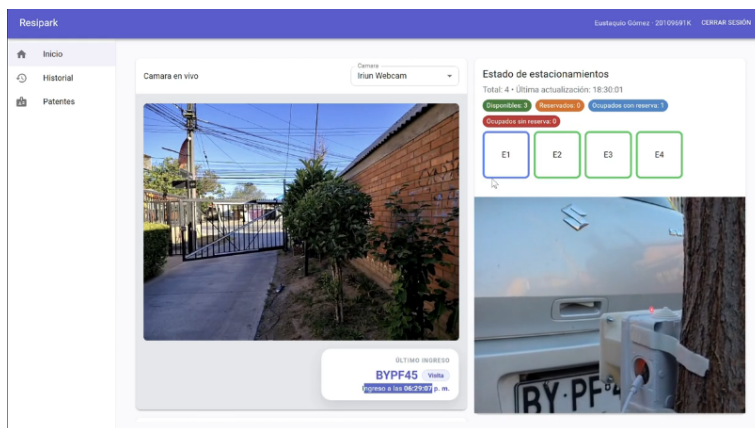


Figura 5.21: Estado de estacionamiento ocupado con reserva, interfaz web.

Finalmente, el sistema contempla la opción de que el conserje remueva la asignación de un estacionamiento a una reserva – por ejemplo, si la visita finalizó anticipadamente o si es necesario reasignar el espacio – desde la misma interfaz web (véase en Figura 5.22).

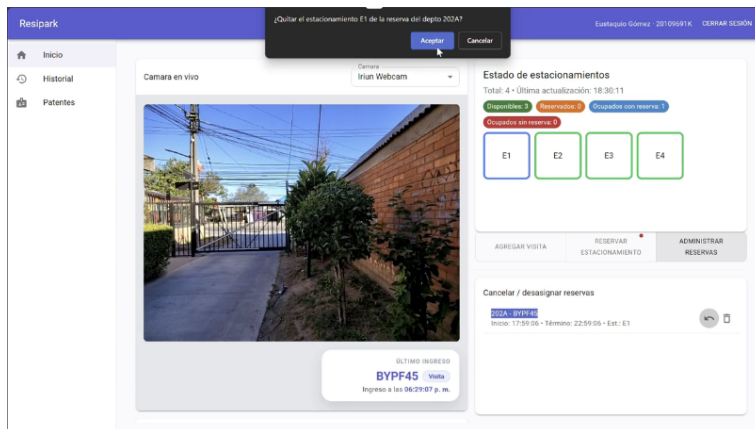


Figura 5.22: Desasignación de estacionamiento a una reserva, interfaz web.

Una vez desasignado el estacionamiento E1, su estado cambia a color rojo, indicando que el espacio se encuentra ocupado – el sensor sigue detectando la presencia física del vehículo – pero sin una reserva activa asociada en el sistema (véase en Figura 5.23). Esta distinción de estados mediante colores (verde, amarillo, azul y rojo) permite al personal de conserjería identificar rápidamente de un vistazo tanto la disponibilidad como la consistencia entre la ocupación física reportada por los sensores y las reservas registradas en el sistema.

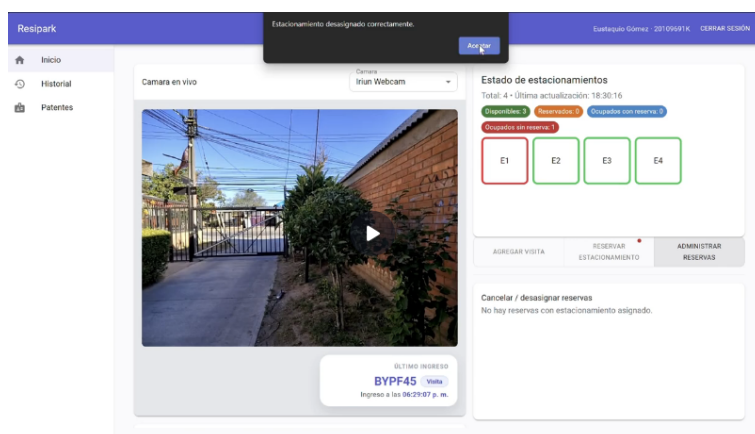


Figura 5.23: Estado de estacionamiento ocupado sin reserva, interfaz web.

5.4. Verificación y Validación del Sistema

Esta sección documenta el proceso de verificación y validación aplicado a *ResiPark*, diferenciando ambos conceptos según su alcance: la verificación evalúa si cada componente del sistema – en particular, los modelos de detección y la integración entre servicios – se construyó correctamente y cumple con las especificaciones técnicas definidas; la validación, en cambio, evalúa si el sistema en su conjunto responde adecuadamente a las necesidades reales de los usuarios en condiciones de operación cercanas a las de producción. Finalmente, se presenta el plan de testing que estructura las pruebas aplicadas a lo largo de ambas etapas.

5.4.1. Verificación

El proceso de verificación se compone de dos etapas complementarias: en primer lugar, se evalúa el desempeño individual de los modelos de detección mediante pruebas unitarias sobre sus respectivos conjuntos de datos; en segundo lugar, se verifica la correcta integración entre los distintos servicios que conforman el pipeline completo: (i) el modelo de detección, (ii) el módulo OCR, (iii) la API en la nube y (iv) la base de datos.

5.4.1.1. Pruebas unitarias de detección

Se entrenaron modelos de YOLOv8s y YOLOv11s sobre un conjunto de imágenes recolectadas de vehículos en el entorno real, con etiquetas manuales de patentes y tipos de vehículo. Se evaluó la capacidad del modelo para detectar regiones de interés (ROI) de patentes bajo distintas condiciones de luz y ángulo.

El modelo de detección vehicular (YOLOv8s) fue entrenado durante 50 épocas utilizando el conjunto de datos [109], compuesto por 9.689 imágenes distribuidas en entrenamiento, validación y prueba (véase en Figura 5.24).

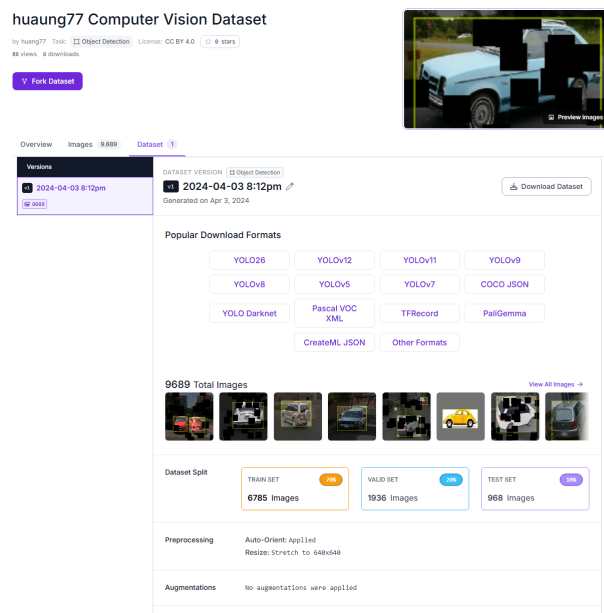


Figura 5.24: Dataset de vehículos *huanag77* (v1), por huang77, Roboflow Universe: 9.689 imágenes (70 % entrenamiento, 20 % validación, 10 % prueba)

El entrenamiento alcanzó una precisión de 0,997, un recall de 0,988, un mAP50 de 0,995 y un mAP50-95 de 0,973 sobre el conjunto de validación (véase en Figura 5.25). Estos resultados muestran un alto desempeño del modelo en la detección de vehículos dentro de la zona de interés definida, con una baja tasa de falsos negativos.

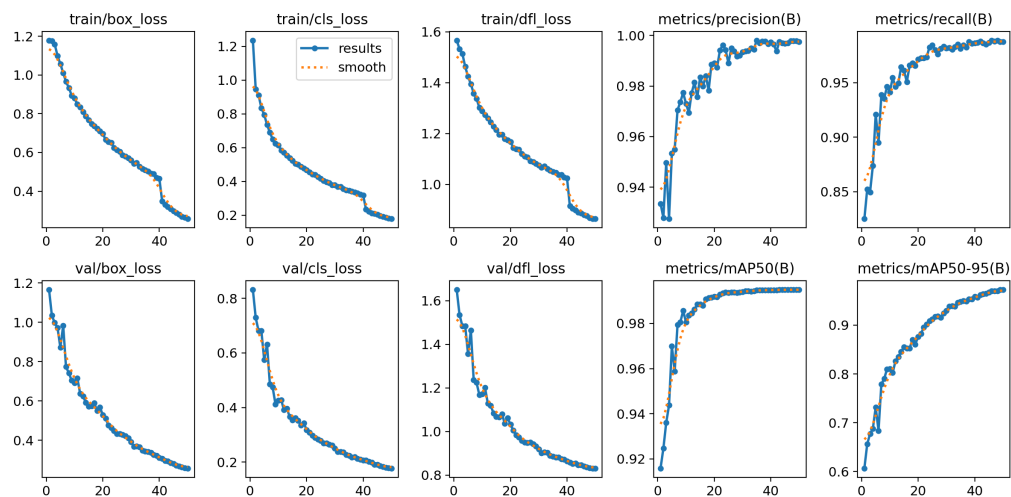


Figura 5.25: Resultados entrenamiento 50 épocas, modelo de vehículos (YOLOv8s)

Esta tendencia también se observa en la matriz de confusión del entrenamiento (véase en Figura 5.26), donde la gran mayoría de las instancias etiquetadas como vehículo son correctamente clasificadas por el modelo, con una cantidad marginal de confusiones respecto al fondo de la escena.

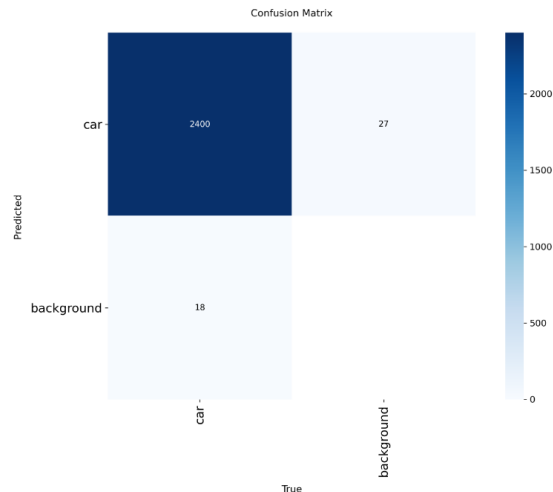


Figura 5.26: Matriz de confusión entrenamiento modelo de autos YOLOv8s

De manera análoga, el modelo de detección de patentes (YOLOv11s) fue entrenado durante 50 épocas sobre un conjunto [110] de 2.048 imágenes de validación con 2.195 instancias etiquetadas, también obtenido de Roboflow Universe (véase en Figura 5.27).

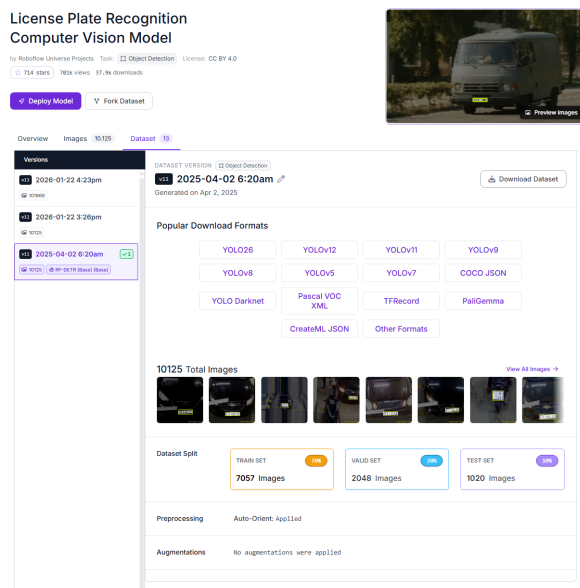


Figura 5.27: Dataset *License Plate Recognition* (v11), por Roboflow Universe Projects: 10.125 imágenes (70 % entrenamiento, 20 % validación, 10 % prueba)

Este entrenamiento alcanzó una precisión de 0,989, un recall de 0,948, un mAP50 de 0,977 y un mAP50-95 de 0,716 (véase en Figura 5.28). Si bien el mAP50 muestra un buen desempeño en la localización de la región de la patente en la gran mayoría de los casos, la caída del mAP50-95 respecto al modelo vehicular refleja una mayor sensibilidad a variaciones en la precisión geométrica del recuadro delimitador, esperable dado el menor tamaño relativo de la patente dentro del fotograma.

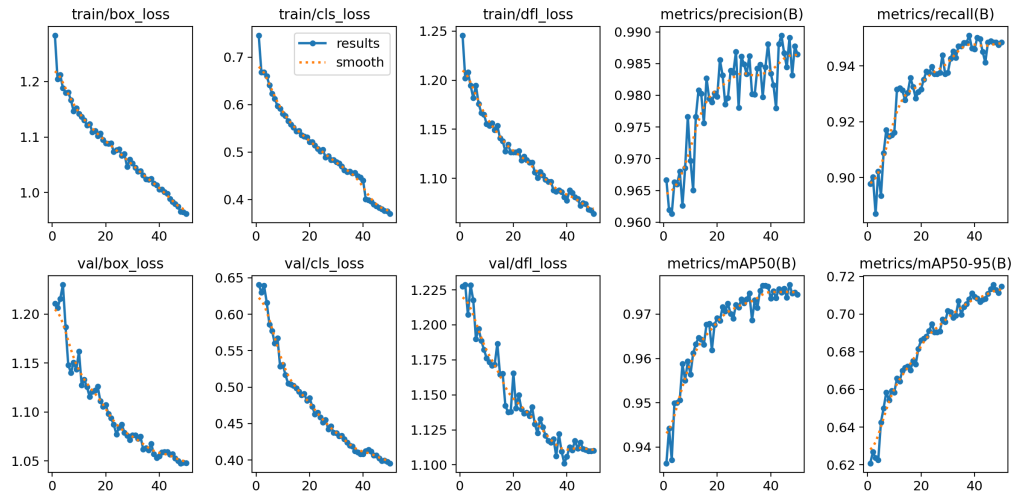


Figura 5.28: Resultados entrenamiento 50 épocas, modelo de patentes (YOLOv11s)

Esta mayor sensibilidad geométrica también se refleja en la matriz de confusión del modelo (véase en Figura 5.29), que muestra una proporción algo mayor de confusiones que la observada para el modelo vehicular, consistente con la naturaleza más exigente de localizar con precisión un objeto de menor tamaño relativo dentro del fotograma.

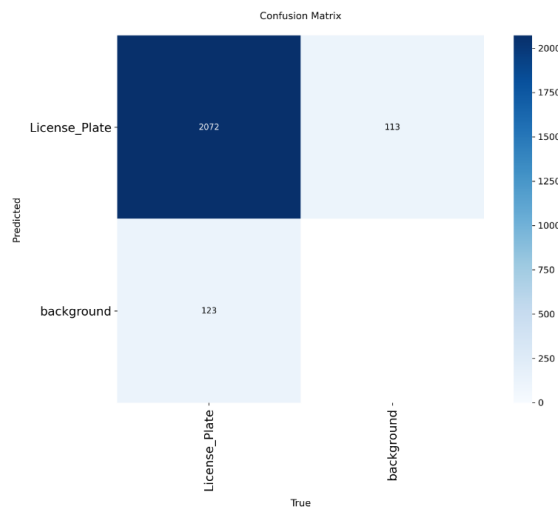


Figura 5.29: Matriz de confusión entrenamiento modelo de patentes YOLOv11s

5.4.1.2. Integración validada entre API, Colab y base de datos

Una vez verificado el desempeño individual de ambos modelos, se validó la correcta integración entre los distintos servicios que conforman el pipeline completo del sistema.

En primer lugar, se verificó la integración entre el modelo de detección de patentes y el módulo OCR: la conexión entre YOLOv11s (detección de la placa) y Microsoft Florence-2 (reconocimiento de caracteres) fue evaluada verificando que la salida del modelo de detección se utilizara correctamente como entrada al módulo OCR, validando que las coordenadas de las *bounding boxes* correspondieran efectivamente a las patentes localizadas (véase en Figura 5.30).

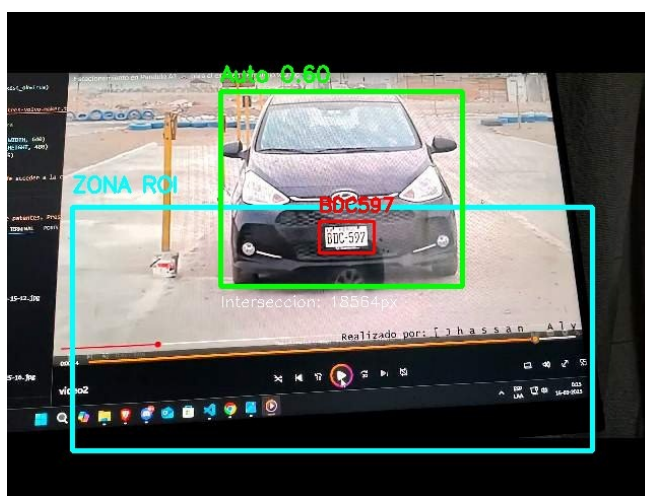


Figura 5.30: Salida modelo de detección y OCR

En segundo lugar, se probó la integración completa de extremo a extremo, siguiendo el flujo real de operación del sistema: captura de la imagen desde la cámara IP, envío al modelo YOLO alojado en Colab, generación de la salida del módulo OCR, y registro final del evento en la base de datos (Supabase). La ejecución exitosa de este flujo confirma que los distintos servicios – separados físicamente entre el dispositivo de borde y la infraestructura en la nube – operan de manera coordinada sin pérdida de información entre etapas.

5.4.2. Validación

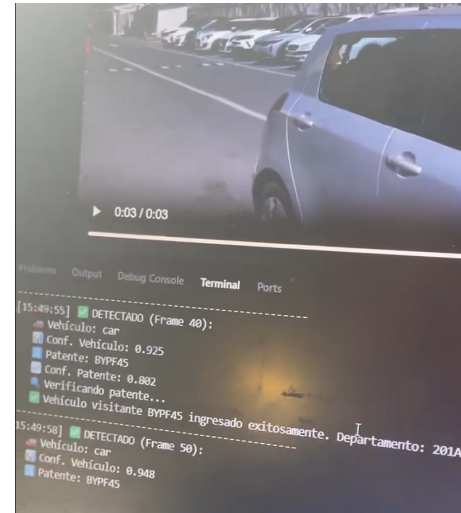
Mientras que la verificación confirmó el correcto funcionamiento técnico de los componentes de manera aislada, la validación evalúa el comportamiento del sistema completo bajo condiciones cercanas a un entorno de operación real, considerando factores como variaciones de iluminación, ángulo de cámara y retroalimentación directa de los usuarios finales del prototipo desplegado en el caso de estudio.

5.4.2.1. YOLO y Patentes

- **Pruebas en videos reales:** Se utilizaron grabaciones de accesos vehiculares en condiciones diurnas y nocturnas. Se validó que el sistema reconociera las patentes con precisión y detectara correctamente vehículos detenidos en una zona específica (véase en Figura 5.31).



(a) Video de vehículo a validar



(b) Client.py detectando patente registrada como visitante en la APP

Figura 5.31: Validación del sistema con video real de acceso vehicular: (a) fotograma de entrada procesado por el modelo de detección; (b) resultado de la verificación mostrando la patente detectada, correctamente identificada como visitante registrado en la aplicación.

- **Errores de transcripción observados:** Se analizaron los errores de reconocimiento registrados durante las pruebas en campo y se ajustó el umbral de confianza de YOLO para minimizar accesos erróneos. La Tabla 5.9 muestra un extracto del registro generado durante una sesión de pruebas en campo, donde se observa que la patente predominante (JIU770) fue transcrita erróneamente en dos de sus ocho detecciones (*JITU7770* y *JTU770*), evidenciando la sensibilidad del modelo OCR a variaciones de ángulo y distancia incluso sobre un mismo vehículo estacionario.

Timestamp	Vehicle Type	Vehicle Conf.	Plate Text	Plate Conf.
2025-09-18 14:45:26	car	0,9	JIU770	0,786
2025-09-18 14:45:32	car	0,748	—	0,809
2025-09-18 14:45:37	car	0,918	JITU7770	0,82
2025-09-18 14:45:40	car	0,909	—	0,807
2025-09-18 14:45:42	car	0,799	—	0,822
2025-09-18 14:45:44	car	0,93	—	0,835

Timestamp	Vehicle Type	Vehicle Conf.	Plate Text	Plate Conf.
2025-09-18 14:45:46	car	0,915	–	0,836
2025-09-18 14:45:50	car	0,866	–	0,834
2025-09-18 14:45:59	car	0,76	KWGS68	0,806
2025-09-18 14:46:01	car	0,81	KWGS68	0,809
2025-09-18 14:46:03	car	0,844	KWGS68	0,809
2025-09-18 14:46:05	car	0,777	KWGS68	0,801
2025-09-18 14:46:07	car	0,92	KWGS68	0,81
2025-09-18 14:46:09	car	0,893	JIU770	0,794
2025-09-18 14:46:11	car	0,86	JIU770	0,802
2025-09-18 14:46:12	car	0,897	JIU770	0,804
2025-09-18 14:46:14	car	0,878	JTU770	0,783
2025-09-18 14:46:16	car	0,845	JIU770	0,802
2025-09-18 14:46:18	car	0,846	JIU770	0,816
2025-09-18 14:46:30	car	0,778	HZPK49	0,817
2025-09-18 14:46:35	car	0,723	HZPK49	0,803
2025-09-18 14:46:42	car	0,701	HZPK49	0,821
2025-09-18 14:46:45	car	0,775	HZPK49	0,822
2025-09-18 14:46:51	car	0,744	HSDY28	0,824
2025-09-18 14:47:02	car	0,755	HSDW28	0,834
2025-09-18 14:47:09	car	0,741	HSDW28	0,819

Tabla 5.9: Registro de eventos de detección y transcripción de patentes durante pruebas en campo

- **Validación de robustez:** Se probó el modelo con diferentes cámaras (distintos ángulos y resoluciones) y se observó un desempeño estable, lo que sugiere una adecuada capacidad de adaptación ante variaciones en las condiciones de captura.
- **Retroalimentación de usuarios:** La retroalimentación obtenida de conserjes y administradores fue favorable respecto del registro automático de accesos, destacándose su potencial para reducir la carga operativa y mejorar la percepción de seguridad.

5.4.3. Plan de Testing

■ YOLO y Patentes

- **Funcional:** Validar que una patente autorizada sea detectada, reconocida y cruzada con la base de datos. Comprobar que accesos no registrados generen alertas.
- **Estrés:** Evaluar el desempeño al procesar flujos de video con múltiples vehículos simultáneos o en condiciones de alta concurrencia.
- **Precisión:** Métricas de *mAP* (Mean Average Precision), *Recall* y *F1-Score* sobre el conjunto de validación.
- **Seguridad:** Verificar el uso de HTTPS en la transmisión de imágenes y resultados, así como las medidas de protección implementadas para el almacenamiento de datos personales.

5.5. Gestión de Riesgos

Esta sección aborda la gestión de riesgos asociada al desarrollo e implementación del sistema, comenzando por los supuestos bajo los cuales se planifica el proyecto y las dependencias externas que condicionan su ejecución. A continuación, se detallan las restricciones técnicas y normativas dentro de las cuales debe operar el sistema. Finalmente, se identifican los principales riesgos del proyecto junto con sus respectivas medidas de mitigación.

5.5.1. Supuestos

Los siguientes son supuestos técnicos y operativos sobre los cuales se basa la planificación y ejecución del proyecto del sistema inteligente de seguridad residencial:

- Las cámaras de seguridad existentes o a instalar proporcionarán imágenes con la calidad (resolución, iluminación, ángulo) necesaria para el correcto funcionamiento de los modelos de visión por computador (detección de vehículos, patentes).
- Los residentes y el personal de conserjería contarán con los dispositivos necesarios (smartphones compatibles para la app móvil, computadores con acceso a internet para el panel web) y estarán dispuestos a adoptar y utilizar el nuevo sistema.
- Se dispondrá de una infraestructura de red local (cableada y/o Wi-Fi) estable y con el ancho de banda suficiente para la transmisión de video y datos entre los componentes del sistema.

- Los algoritmos de visión por computador podrán ser entrenados o ajustados para alcanzar niveles de precisión aptos para la seguridad en el entorno real de la comunidad de acceso cerrado.
- El entorno residencial piloto estará disponible y la comunidad del recinto colaborará para la realización de pruebas y recolección de feedback.
- Se contará con acceso a fuentes de datos (imágenes, videos) suficientes y representativas para el entrenamiento y prueba de los modelos de visión por computador.

5.5.2. Dependencias

El éxito del proyecto depende del cumplimiento de las siguientes condiciones o hechos que están, en parte o totalmente, fuera del control directo del equipo de proyecto:

- **Disponibilidad de Hardware Específico:** Adquisición e instalación oportuna de cámaras, microcontroladores y otros componentes de hardware que cumplan con las especificaciones requeridas.
- **Servicios Cloud:** Si se utilizan servicios en la nube para almacenamiento, procesamiento o base de datos, se depende de la disponibilidad, rendimiento y políticas de dichos proveedores.
- **Cooperación de la Administración de la Comunidad Residencial:** Permisos para instalación de equipos, acceso a infraestructura, colaboración durante pruebas piloto y facilitación de la comunicación con los residentes.
- **Marco Legal y Normativo Vigente:** Cambios en la Ley de Protección de Datos Personales o normativas locales sobre videovigilancia que puedan requerir ajustes en el diseño o funcionalidades del sistema.
- **Integración con Sistemas Existentes:** Si el sistema debe interactuar con sistemas de control de acceso (barreras o portones) o de gestión de la comunidad residencial preexistentes, se depende de la compatibilidad y APIs de dichos sistemas.

5.5.3. Restricciones

Las restricciones corresponden a condiciones o limitaciones impuestas al proyecto que no pueden ser modificadas por el equipo de desarrollo, ya sea por decisión del cliente, exigencias normativas o limitaciones de recursos, y que por tanto delimitan el espacio de diseño dentro del cual debe construirse el sistema. A continuación se detallan las restricciones identificadas para el desarrollo del proyecto:

- **Uso de Tecnologías Específicas:** El modelo de visión por computador para detección de vehículos debe basarse en YOLO. Se debe desarrollar una aplicación móvil (orientada a residentes) y una aplicación web (orientada a conserjes). La base de datos debe ser SQL.
- **Cumplimiento Normativo:** El sistema debe diseñarse y operarse en cumplimiento con la Ley de Protección de Datos Personales vigente en Chile (la normativa vigente sobre protección de datos personales en Chile) y cualquier otra normativa aplicable a la videovigilancia y tratamiento de datos personales.
- **Alcance del Sistema:** Las funcionalidades a desarrollar en la primera versión del sistema están limitadas a las definidas en la sección correspondiente del proyecto, excluyendo explícitamente componentes como ciberseguridad avanzada, autenticación multifactor completa, etcétera.
- **Recursos Humanos:** El desarrollo será llevado a cabo por el equipo de proyecto definido, con sus conocimientos y habilidades actuales.

5.5.4. Riesgos

Los riesgos corresponden a eventos inciertos que, de materializarse, podrían afectar negativamente el cumplimiento de los objetivos, el cronograma o la calidad del sistema desarrollado. Los riesgos fueron identificados considerando tanto los aspectos técnicos del proyecto (modelos de visión por computador, sensores, integración de módulos) como los aspectos operativos y normativos (adopción por parte de los usuarios, protección de datos personales). Para cada riesgo se define además una medida de mitigación orientada a reducir su probabilidad de ocurrencia o su impacto sobre el proyecto. La Tabla 5.10 presenta los principales riesgos identificados junto con sus correspondientes medidas de mitigación.

Tabla 5.10: Riesgos y Medidas de Mitigación

Riesgo	Medida de Mitigación
R01: Precisión insuficiente del modelo de detección de vehículos en condiciones reales, afectando la fiabilidad del control de acceso.	M01: Recopilación de un dataset de entrenamiento diverso y representativo del entorno real. Aplicación de técnicas de aumento de datos. Iteraciones de entrenamiento y ajuste fino del modelo. Pruebas exhaustivas en el entorno piloto y reentrenamiento si es necesario. Considerar un umbral de confianza ajustable para derivar casos dudosos a revisión manual.

Riesgo	Medida de Mitigación
R02: Fallos en el sistema de sensores para detectar ocupación de espacios, lo que podría llevar a información incorrecta en la app de residentes.	M02: Selección y prueba rigurosa de distintos tipos de sensores (por ejemplo, infrarrojos, ultrasonido, magnéticos) para la detección confiable de ocupación. Calibración cuidadosa de la ubicación y sensibilidad de los sensores en el entorno real. Implementación de mecanismos de validación temporal (como el promedio de lecturas en intervalos definidos) para reducir falsos positivos o negativos.
R03: Vulnerabilidades de ciberseguridad en la base de datos SQL, aplicación móvil o panel web, que comprometan la protección de datos personales (Ley 19.628) o permitan accesos no autorizados al sistema.	M03: Diseño seguro desde el inicio (security by design). Aplicación de buenas prácticas de codificación segura. Encriptación de contraseñas y datos sensibles en tránsito y reposo. Validación de todas las entradas de datos. Realización de pruebas de seguridad básicas (por ejemplo, OWASP Top 10 para web) y planificación de pruebas de ciberseguridad más exhaustivas (como objetivo específico). Limitar permisos de acceso a la base de datos.
R04: Baja adopción o resistencia al uso del sistema por parte de los residentes o personal de conserjería debido a interfaces poco intuitivas, complejidad percibida o falta de confianza.	M04: Diseño de interfaces centrado en el usuario (UI/UX). Realización de pruebas de usabilidad con usuarios finales durante el desarrollo del sistema. Elaboración de manuales de usuario claros y concisos. Sesiones de capacitación para conserjes. Comunicación clara de los beneficios del sistema.
R05: Fallos de componentes de hardware (cámaras, microcontroladores, servidor) o de la infraestructura de red, interrumpiendo el funcionamiento del sistema.	M05: Selección de hardware de calidad y adecuado para las condiciones de operación. Diseño con redundancia para componentes críticos si el presupuesto lo permite. Establecimiento de un plan de mantenimiento preventivo básico. Monitoreo del estado del sistema para detección temprana de fallos.

Riesgo	Medida de Mitigación
R06: Dificultad para integrar los diferentes módulos del sistema (visión por computador, microcontroladores, apps móviles y web, base de datos) de forma eficiente y funcional.	M06: Definición clara de APIs y protocolos de comunicación entre módulos. Desarrollo incremental y pruebas de integración continuas. Uso de herramientas de versionado y gestión de configuración.
R07: Retrasos en el cronograma del proyecto debido a la complejidad técnica inesperada, subestimación de tiempos o problemas en la adquisición de recursos.	M07: Planificación detallada de tareas con hitos claros. Seguimiento riguroso del avance. Identificación temprana de cuellos de botella. Flexibilidad para reasignar recursos o ajustar el alcance del sistema si es estrictamente necesario y se acuerda formalmente.

5.6. Análisis Crítico, Seguridad y Limitaciones

El desarrollo de un sistema IoT como ResiPark implica un análisis de riesgos que abarca desde el hardware de IoT hasta la infraestructura de nube y la seguridad de las aplicaciones. Esta sección detalla los riesgos técnicos identificados en el prototipo, los desafíos materializados durante el desarrollo, un análisis de seguridad extendido y las limitaciones del sistema.

5.6.1. Riesgos Técnicos del Prototipo Implementado

En esta categoría se agrupan los riesgos y puntos de fallo identificados en la arquitectura final del sistema, agrupados por módulo.

- **Gestión de Acceso Vehicular (M1)**

- **Dependencia de Google Colab:** El uso de Colab para la ejecución del modelo de IA introduce una dependencia crítica de un servicio externo y una latencia de red notable.
- **Latencia en Inferencia:** El tiempo de respuesta total (cámara → Colab → backend) puede aumentar significativamente con video de alta resolución o congestión de red, afectando la experiencia de usuario en el acceso.

- **Variabilidad Ambiental:** Condiciones extremas (lluvia intensa, reflejos, suciedad en la cámara, noche) pueden reducir la precisión tanto de YOLOv8/11 como de Florence-2.
- **Falsos Positivos/Negativos:** Una patente mal leída puede generar accesos indebidos o, peor aún, denegar el acceso a un residente registrado. El sistema mitiga esto parcialmente cruzando el tipo de vehículo, pero el riesgo persiste.

■ Interfaces de Usuario y Backend (M3 y M4)

- **Dependencia de Conectividad:** El funcionamiento integral del sistema depende de la disponibilidad de conectividad a Internet. Si la conexión de la comunidad residencial (para el receptor IoT y el cliente IA) o del usuario (móvil) falla, el sistema deja de operar.
- **Escalabilidad del Backend:** Las APIs de FastAPI se ejecutan como procesos individuales. Bajo alta concurrencia (cientos de residentes usando la app simultáneamente) podrían saturarse.
- **Riesgos de Seguridad:** Como se detalla en la sección [5.6.3](#), existen riesgos de seguridad en las APIs, la red local y los dispositivos de borde que deben ser gestionados.
- **Sincronización de Estados:** Un usuario podría ver un estacionamiento libre en la app (debido al refresco de 5s), justo cuando un vehículo se está estacionando y el sensor IoT aún no ha reportado el cambio. Esto puede generar conflictos en la asignación de reservas.

5.6.2. Riesgos Materializados y Mitigaciones Aplicadas

Durante la fase de desarrollo, se materializó un riesgo técnico significativo relacionado con la pila de software de IA. El plan original contemplaba el uso de **PaddleOCR** para el reconocimiento de caracteres. Sin embargo, su implementación presentó severos conflictos de dependencias y problemas de compatibilidad de versiones que bloquearon el progreso.

Como mitigación, el equipo investigó y validó una tecnología alternativa, migrando exitosamente el módulo de OCR al modelo **Microsoft Florence-2**. Este cambio no solo resolvió los problemas técnicos, sino que demostró una alta precisión en la lectura de patentes chilenas y una integración más estable. Esta adaptación demuestra la flexibilidad del diseño modular del sistema.

5.6.3. Análisis Extendido de Seguridad y Casos Borde

Un sistema de seguridad y control de acceso debe ser analizado rigurosamente ante vectores de ataque y fallos.

5.6.3.1. Seguridad ante Patentes Falsas

Uno de los principales vectores de ataque para sistemas ALPR es la **suplantación o falsificación de placas patentes**. Los escenarios de ataque incluyen: (i) instalar una placa falsa o clonada en otro vehículo, (ii) cubrir o alterar caracteres para confundir al OCR, o (iii) presentar una imagen de una patente frente a la cámara.

El sistema propuesto mitiga parcialmente estos riesgos:

1. **Detección de Vehículo:** El modelo YOLOv8 primero detecta un vehículo y solo entonces busca una patente dentro de ese vehículo. Esto dificulta los ataques basados en la presentación de una imagen aislada de una patente frente a la cámara.
2. **Cruce de Información:** El sistema valida no solo la patente, sino también el tipo de vehículo (por ejemplo, ‘auto’ vs ‘camioneta’) contra la base de datos (ver endpoint ‘/verificar-patente’). Un atacante necesitaría clonar tanto la patente como el tipo de vehículo del residente, incrementando la dificultad de ejecutar una suplantación exitosa.

A futuro, este módulo podría reforzarse integrando APIs de consulta externas (como las del Registro Civil) para validar la autenticidad y estado legal de una patente en tiempo real.

5.6.3.2. Seguridad de APIs y Control de Acceso

La arquitectura de microservicios (3 APIs) introduce una superficie de ataque que debe ser gestionada:

- **Autenticación y Autorización (JWT):** El sistema utiliza JWTs para la autenticación de las APIs web y móvil. La API Móvil (‘reservas.py’, ‘vehiculos.py’) implementa una mitigación de seguridad crucial: valida que el ‘id departamento’ (obtenido del ‘sub’ del token JWT) coincida con el ‘id departamento’ del recurso que se intenta modificar. Esto previene que un residente pueda ver, modificar o borrar las reservas o vehículos de otro residente (escalada de privilegios horizontal).

- **Seguridad del Sistema de Reservas:** Un atacante con un token de residente válido podría intentar un ataque de denegación de servicio creando miles de reservas falsas. Como mitigación a futuro, se debe implementar *Rate Limiting* (limitación de peticiones) en los endpoints de creación.
- **Protección del Cliente IA:** El ‘client.py’ ataca un endpoint (‘/verificar-patente/’) que actualmente se encuentra accesible dentro de la red local. A futuro, este endpoint debe ser protegido con una clave de API (API Key) o un token de servicio estático para evitar que cualquier dispositivo en la red local pueda consultarlo.

5.6.4. Limitaciones del Sistema

Finalmente, es crucial definir el alcance del prototipo finalizado, reconociendo sus limitaciones como base para el trabajo futuro:

- **Limitación 1 - Actuador Físico:** La integración con el actuador físico del portón se realizó mediante un relé, cuya señal de apertura fue validada en el portón correspondiente al caso de estudio, ubicado en un pasaje cerrado de la comuna de Cerrillos. Sin embargo, esta implementación se encuentra condicionada a las características específicas de dicho portón y de su mecanismo de accionamiento, por lo que su adopción en otras comunidades residenciales requerirá evaluar la compatibilidad con la infraestructura existente y la necesidad de adaptar el mecanismo de integración.
- **Limitación 2 - Infraestructura de IA:** El modelo de detección se despliega en Google Colab, un entorno de desarrollo que no garantiza disponibilidad (uptime) y presenta una latencia de red significativa. Para un despliegue real, este modelo debe ser migrado a un servidor dedicado con GPU (por ejemplo, AWS EC2 G4/G5, un servidor local con una GPU NVIDIA, o un dispositivo de borde como NVIDIA Jetson).
- **Limitación 3 - Sincronización de Estados:** La actualización del estado de los estacionamientos de visita depende de la comunicación entre los sensores IoT, el nodo receptor, el backend y las aplicaciones web y móvil, por lo que pueden producirse desfases temporales entre la ocupación física de un espacio y el estado visualizado en la plataforma. Esta condición puede generar inconsistencias durante la asignación o reserva de estacionamientos, especialmente cuando ocurren cambios de estado en intervalos cercanos. Para un despliegue a mayor escala, será necesario implementar mecanismos de actualización en tiempo real, como WebSockets o notificaciones push, que permitan reducir estos desfases y mantener una representación más consistente del estado de los estacionamientos.

En síntesis, este capítulo documentó el proceso de solución del problema planteado: desde la selección tecnológica mediante matrices de decisión ponderada, hasta la implementación de los cuatro módulos de ResiPark y su evaluación en un caso de estudio real. Los resultados obtenidos permiten verificar el funcionamiento de los principales componentes y validar el flujo integrado del sistema: el modelo de detección vehicular (YOLOv8s) alcanzó un mAP50 de 0,995 y un mAP50-95 de 0,973, mientras que el modelo de patentes (YOLOv11s) alcanzó un mAP50 de 0,977 y un mAP50-95 de 0,716, diferencia que refleja una mayor sensibilidad geométrica asociada al menor tamaño de la patente dentro del fotograma; las pruebas en campo, además, mostraron cierta sensibilidad del OCR a variaciones de ángulo y distancia. Estos resultados constituyen una alternativa viable y de bajo costo para el segmento de pasajes cerrados sin soluciones de gestión inteligente, aunque persisten riesgos y limitaciones – especialmente la dependencia de infraestructura en la nube – que deben abordarse antes de un despliegue en producción a mayor escala. A continuación, se presentan las conclusiones generales y las proyecciones de trabajo futuro.

Capítulo 6

Conclusiones

6.1. Cumplimiento de los Objetivos Planteados

El presente trabajo tuvo como objetivo general diseñar e implementar un sistema inteligente de seguridad para comunidades residenciales de acceso cerrado, orientado a optimizar la protección de los residentes y reducir vulnerabilidades ante accesos no autorizados. A partir del despliegue piloto de *ResiPark* en un pasaje cerrado de la comuna de Cerrillos – un recinto que históricamente ha carecido de conserjería, infraestructura digital y presupuesto para adoptar las soluciones comerciales – se considera que el objetivo general fue alcanzado de acuerdo con los resultados obtenidos durante el despliegue piloto, validando la viabilidad técnica de una solución modular y de bajo costo para este segmento específico de comunidades residenciales. A continuación, se detallan las conclusiones asociadas a cada uno de los objetivos específicos planteados.

6.1.1. OE1 - Análisis de mercado de sistemas inteligentes de seguridad

El análisis de mercado desarrollado en el Capítulo 3 permitió identificar y caracterizar al menos cinco tecnologías, metodologías y soluciones comerciales aplicables al desarrollo del sistema, cumpliendo así lo establecido en OE1. Este análisis evidenció una brecha concreta en el mercado chileno: las soluciones administrativas (Edifito, Comunidad Feliz, Kastor, ComunidadActiva) tratan el control de acceso como una funcionalidad accesorio, mientras que las soluciones de control de acceso físico (SafeCard, Federal Access, Smartki, Citofon, VisitPass) y los referentes internacionales (Proptia, placa.ai), si bien resuelven de forma nativa la identificación vehicular, constituyen en todos los casos tecnología propietaria cerrada, sin documentación técnica pública, y ninguna ofrece gestión de reservas de estacionamiento de visita.

Esta brecha, observada también en el modelo de integración por componentes representado por ZKTeco, fundamentó la propuesta de una arquitectura modular, documentada y adaptada a comunidades de menor escala como la abordada en el caso de estudio.

6.1.2. OE2 - Aplicación y entrenamiento de un modelo YOLO para detección de vehículos y patentes

Se cumplió cabalmente este objetivo mediante la implementación de dos modelos de la familia YOLO diferenciados por subtask: YOLOv8s para la detección vehicular y YOLOv11s para la localización de patentes, ambos reentrenados sobre conjuntos de datos propios obtenidos de Roboflow (9.689 y 10.125 imágenes respectivamente), alcanzando un mAP50 de 0,995 para vehículos y de 0,977 para patentes. Cabe destacar que la elección de estos dos modelos específicos no fue arbitraria, sino que se sustentó en la Matriz de Decisión Ponderada desarrollada en el Capítulo 5 donde YOLOv11s obtuvo la mayor puntuación ponderada (8,60) por su mejor equilibrio entre precisión y eficiencia computacional. La decisión de mantener YOLOv8s para la detección vehicular – pese a no ser la opción mejor puntuada – respondió a un criterio pragmático de reutilización de un modelo ya entrenado, evidenciando que las decisiones de ingeniería en un proyecto real no siempre se alinean de forma estricta con el resultado teórico de una matriz de decisión, sino que también deben ponderar restricciones de tiempo y recursos disponibles.

Asimismo, la integración de Florence-2 para la transcripción OCR – en reemplazo de PaddleOCR, descartado por conflictos de dependencias durante el desarrollo, tal como se documentó en la Sección 5.6 – demuestra la capacidad de adaptación del diseño modular del sistema ante contingencias técnicas imprevistas, sin comprometer el cumplimiento del objetivo planteado.

6.1.3. OE3 - Sistema de sensores *open-source* para detección de estacionamientos disponibles

Se implementó un sistema de sensado basado en un microcontrolador ESP32 con sensor ultrasónico de proximidad (rango de detección de 60 cm), comunicado mediante un módulo de radio LoRa SX1276/78 hacia un nodo receptor que reenvía la información al backend vía WiFi, cumpliendo lo establecido en OE3. Es relevante señalar que, si bien la Matriz de Decisión Ponderada del Capítulo 5 había identificado al sensor magnético como la alternativa de mayor puntuación por su desempeño en entornos subterráneos, la implementación final optó por el sensor ultrasónico, decisión consistente con el criterio ya aplicado en la selección de modelos de detección: la disponibilidad de componentes, el costo y la

facilidad de integración con la arquitectura LoRa existente primaron sobre el resultado teóricamente óptimo de la matriz. Esta discrepancia no invalida el proceso de evaluación desarrollado, sino que evidencia su naturaleza de herramienta de apoyo a la decisión, no de regla determinística.

6.1.4. OE4 - Aplicación móvil y plataforma web para residentes, conserjes y administradores

Se diseñaron e implementaron una aplicación móvil orientada a residentes (desarrollada en React Native, con funcionalidades de registro, reserva y consulta de historial de estacionamientos de visita) y una plataforma web orientada a conserjes y administradores (con monitoreo en tiempo real de accesos vehiculares, historial de ingresos y administración de patentes autorizadas), cumpliendo lo establecido en OE4. Ambas interfaces fueron validadas funcionalmente durante el despliegue piloto, permitiendo un flujo completo de reserva, asignación, ocupación y liberación de estacionamientos de visita, tal como se documentó en la Sección de Funcionamiento del Sistema del Capítulo 5. Cabe destacar que las interfaces finales conservan la estructura y lógica de navegación validadas tempranamente mediante los mockups presentados en el Capítulo 4 (login, panel de estacionamientos, registro de patentes y módulo de portón), lo que evidencia la utilidad del prototipado temprano como guía consistente para el desarrollo del frontend.

6.1.5. Vínculo con la problemática y el estado del arte

Los resultados obtenidos permiten sostener que *ResiPark* responde de manera concreta a las dos problemáticas centrales identificadas en el Capítulo 2: el control manual e ineficiente de los accesos vehiculares, y la gestión ineficiente de los estacionamientos de visita. Frente al primer problema, el pipeline de cinco etapas implementado (detección de zona de interés, detección vehicular, detección de patente, transcripción OCR y validación heurística) reemplaza la verificación subjetiva del personal de conserjería por un mecanismo automatizado y trazable, mitigando – aunque no eliminando por completo, según se discutió en el análisis de seguridad ante patentes falsas – las vulnerabilidades de ingeniería social documentadas en el Capítulo 3. Frente al segundo problema, la red de sensores IoT y el sistema de reservas digitalizan por completo un proceso que, en el modelo tradicional, dependía de rondas presenciales o directamente no se realizaba.

En relación con el estado del arte revisado, la evolución observada en la literatura de sistemas ALPR – desde enfoques clásicos de procesamiento de imágenes hacia arquitecturas de aprendizaje profundo, pasando por casos híbridos como en [55] hasta sistemas end-to-end como VehicleTracker

[69] – encuentra continuidad directa en la arquitectura de *ResiPark*, que además incorpora un elemento no explorado en los antecedentes revisados: la diferenciación entre residentes y visitantes con reserva activa, resuelta mediante dos rutas de verificación dentro de un mismo servicio, y la integración nativa de la gestión de reservas de estacionamiento – carencia identificada de forma transversal en las siete soluciones comerciales analizadas en la Sección 3.5.

Asimismo, la arquitectura de cuatro capas adoptada para el sistema – Obtención de Datos, Comunicación, Soporte (Middleware) y Aplicación – constituye una aplicación directa del modelo funcional propuesto en [6] y [71], revisado en el Capítulo 3, evidenciando que el estado del arte no solo orientó la selección de componentes individuales, sino también el diseño estructural completo del sistema.

6.2. Trabajos futuros

A partir de las limitaciones del producto mínimo viable documentadas en la Sección 5.6 se identifican las siguientes líneas de trabajo futuro:

- **Migración de la infraestructura de inferencia:** sustituir Google Colab – entorno que no garantiza disponibilidad ni latencia consistente – por un servidor dedicado con GPU (instancia en la nube tipo AWS EC2 G4/G5, servidor local con GPU NVIDIA, o un dispositivo de borde como NVIDIA Jetson, ya evaluado en el análisis comparativo de hardware del Capítulo 5), reduciendo la dependencia crítica de un servicio externo.
- **Robustecimiento ante patentes falsificadas:** incorporar validación en tiempo real contra fuentes externas (por ejemplo, APIs del Registro Civil) para verificar la autenticidad y estado legal de una patente detectada, complementando las mitigaciones actuales basadas en la detección previa del vehículo y el cruce con el tipo de vehículo registrado.
- **Endurecimiento de seguridad en las APIs:** implementar *rate limiting* en los endpoints de creación de reservas para prevenir ataques de denegación de servicio, y proteger el endpoint de verificación de patentes – actualmente abierto en la red local – mediante una clave de API o token de servicio.
- **Mejora de la resiliencia ante baja iluminación:** integrar algoritmos complementarios de detección nocturna, en línea con lo identificado en la literatura revisada para sistemas híbridos CV-IoT, con el fin de reducir la degradación de precisión observada en condiciones de iluminación adversa.

- **Escalabilidad del backend:** evaluar arquitecturas de despliegue con múltiples instancias o balanceo de carga para las APIs de FastAPI, de cara a un escenario de adopción en comunidades de mayor tamaño con uso concurrente elevado.
- **Reducción de la latencia de sincronización de estados:** explorar mecanismos de actualización en tiempo real (por ejemplo, WebSockets o *push notifications*) que reemplacen el refresco periódico actual, mitigando los conflictos de asignación identificados entre el estado mostrado en la aplicación y el estado real reportado por el sensor IoT.
- **Extensión a otras tipologías residenciales:** validar el sistema en comunidades de mayor escala (condominios con conserjería permanente) para evaluar su desempeño como complemento – y no solo como sustituto – del personal de seguridad existente, y explorar su adaptación a otras tipologías habitacionales mencionadas en el Capítulo 2 (villas, blocks, pasajes de mayor tamaño).

6.3. Comentarios

El desarrollo de este proyecto representó un proceso de aprendizaje que fue mucho más allá de la implementación de un sistema tecnológico. A lo largo del trabajo, fue necesario enfrentarse a situaciones que inicialmente no habían sido consideradas y que obligaron a replantear decisiones, reducir el alcance de la propuesta y aprender a trabajar con los recursos realmente disponibles.

En las primeras etapas, la idea original contemplaba un sistema mucho más amplio, que incluía funciones de administración de la comunidad residencial, pago de gastos comunes, comunicación entre residentes, reserva de espacios comunes e incluso mecanismos de vigilancia orientados a detectar situaciones anómalas. Con el avance del desarrollo, fue necesario reconocer que intentar abarcar todas estas funcionalidades habría significado perder profundidad en los problemas que realmente se buscaba abordar. Aunque reducir el alcance no fue una decisión sencilla, con el tiempo se comprendió que concentrar los esfuerzos en el control de accesos vehiculares y la gestión de estacionamientos permitió construir una solución más completa y coherente.

Otro aspecto que marcó el desarrollo del proyecto fue la disponibilidad de recursos. Desde el comienzo se tuvo claro que no era posible contar con infraestructura de alto costo ni con equipamiento especializado, por lo que muchas decisiones estuvieron condicionadas por el presupuesto disponible. Esto llevó a trabajar con componentes accesibles, herramientas de código abierto y servicios que ofrecieran alternativas gratuitas o de bajo costo. En algunos momentos estas restricciones representaron dificultades importantes, pero también permitieron comprender que una solución tecnológica no necesariamente debe construirse a partir de los recursos más costosos para resultar útil.

Personalmente, uno de los aspectos que más llamó mi atención durante la investigación fue la diferencia entre la percepción de inseguridad y los indicadores objetivos de criminalidad. El estudio de la problemática permitió observar cómo la inseguridad se encuentra presente de manera constante en la vida cotidiana y cómo la forma en que se percibe el delito puede llegar a ser tan relevante como su ocurrencia efectiva. Esta reflexión fue particularmente significativa, considerando que la preocupación por la seguridad fue precisamente uno de los puntos de partida de este proyecto.

La ejecución también implicó aprender a convivir con errores y cambios de decisión. Algunas tecnologías consideradas inicialmente tuvieron que ser reemplazadas, ciertos componentes no funcionaron como se esperaba y varias ideas debieron modificarse a medida que aumentaba el conocimiento sobre el problema y sobre las posibilidades reales de implementación. En retrospectiva, estas dificultades fueron parte importante del aprendizaje, ya que permitieron comprender que el desarrollo de un proyecto de ingeniería no consiste únicamente en ejecutar un diseño previamente definido, sino también en adaptarse cuando las condiciones cambian.

Finalmente, uno de los aspectos más significativos de esta experiencia fue haber podido desarrollar el proyecto junto a tres compañeros con quienes se ha compartido gran parte de la trayectoria universitaria. Más allá de los resultados técnicos alcanzados, haber cerrado esta etapa trabajando como equipo convirtió todo el proceso en una experiencia especialmente significativa a nivel personal. El trabajo implicó momentos de dificultad, diferencias de opinión, largas jornadas y también instancias de satisfacción al ver que una idea que inicialmente parecía difícil de concretar terminaba transformándose en un sistema funcional. Concluir esta etapa de la carrera junto a las mismas personas con las que se compartió gran parte de ella constituye, en lo personal, uno de los aspectos más valiosos de todo este proyecto.

Bibliografía

- [1] Emol, “Portonazos y encerronas: Fiscalía Oriente registró 6.769 casos entre 2022 y 2025.” Emol.com, <https://www.emol.com/noticias/Nacional/2026/02/24/1192384/portonazos-encerronas-fiscalia-orient.html>, Feb. 2026. Consultado en julio de 2026.
- [2] Instituto de Políticas Públicas, Universidad Andrés Bello, “Seguridad en los hogares en Chile: Evidencia desde la encuesta CASEN 2015–2024,” tech. rep., Universidad Andrés Bello, Santiago, Chile, Jan. 2026. https://ipp.unab.cl/wp-content/uploads/2026/01/20260109-IPP-UNAB_Informe-Encuesta-Casen-2024_Seguridad.pdf.
- [3] Mercado Libre Chile, “Control remoto universal 4 canales RF 433MHz portón.” <https://www.mercadolibre.cl/control-remoto-universal-4-canales-rf-433mhz-porton/p/MLC50798468>, 2024. Consultado en julio de 2026.
- [4] Casa Royal, “Control remoto Rossi Rolling Code 433MHz.” <https://www.casaroyal.cl/r0ssi-control-remoto-rossi-rolling-code-433m-/p>, 2024. Consultado en julio de 2026.
- [5] Controlhouse SpA, “Receptor portón automático 350 metros.” <https://controlhouse.cl/inicio/124-receptor-porton-automatico-350-metros.html>, 2024. Consultado en julio de 2026.
- [6] H. K. Alaoui, M. Coulibaly, and A. Errami, “Analyzing smart parking system: Wsn technologies, power saving and guidance,” in *2024 International Conference on Circuit, Systems and Communication (ICCSC)*, pp. 1–6, 2024.
- [7] K. J. Alfaro, “El impacto del internet de las cosas (IoT) en la sociedad moderna, con relación al crecimiento y evolución industrial, y sus retos. una revisión sistemática de literatura,” *ilia*, pp. 16–25, oct 2024.
- [8] L. S. Seminario, “Inseguridad ciudadana: Problema social, latente que afecta a nivel de latinoamérica,” *Revista de Climatología Edición Especial Ciencias Sociales*, vol. 23, p. 3040, 2023.

- [9] L. Kanashiro, “Entre el miedo y la ira. prácticas de seguridad en los sectores de menores recursos en Lima, Perú,” *Revista mexicana de ciencias políticas y sociales*, vol. 66, no. 241, pp. 317–345, 2021.
- [10] C. E. Torres Reyes, “Impacto del plan de vigilancia comunitaria por cuadrantes en la seguridad ciudadana del barrio chapinero central en Bogotá 2015-2018,” *Revista Jurídica Piélagus*, vol. 19, no. 2, pp. 9–38, 2020.
- [11] F. Sánchez and C. Juárez, “Política de seguridad en México: Combate al narcotráfico - entre la seguridad nacional y la seguridad pública,” *Revista IUS*, vol. 13, no. 44, pp. 229–250, 2019.
- [12] La Tercera, “Enusc y percepción de inseguridad.” Editorial La Tercera, <https://www.latercera.com/editorial/noticia/enusc-y-percepcion-de-inseguridad/>, July 2025. Publicado: 08-jul-2025. Accedido: 29-abr-2026.
- [13] Escuela de Arquitectura y Diseño, Pontificia Universidad Católica de Valparaíso, “Conjuntos habitacionales: Estudio de tipologías y casos.” Casiopea, https://wiki.ead.pucv.cl/Conjuntos_Habitacionales:_Estudio_de_tipolog%C3%ADas_y_casos, 2010.
- [14] Ministerio de Vivienda y Urbanismo and Biblioteca del Congreso Nacional, “Soluciones comerciales y marco normativo para la administración de condominios en Chile: Plataformas SaaS (ComunidadFeliz, Edipro, Kastor, Swapi), servicios integrales (ACRPRO, Adec Chile, Gestión Condominios) y cumplimiento de la ley n° 21.442.” Santiago, Chile, <https://bcn.cl>, 2024. Accedido: 22-abr-2024.
- [15] Ministerio Secretaría General de la Presidencia, “Ley 21719: Regula la protección y el tratamiento de los datos personales y crea la agencia de protección de datos personales.” Biblioteca del Congreso Nacional de Chile (BCN) - Ley Chile, <https://www.bcn.cl/leychile/navegar?idNorma=1209272>, Dec. 2024.
- [16] Ministerio de Vivienda y Urbanismo, “Ley 21442: Aprueba nueva ley de copropiedad inmobiliaria.” Biblioteca del Congreso Nacional de Chile (BCN) - Ley Chile, <https://www.bcn.cl/leychile/navegar?idNorma=1174663>, Apr. 2022. Publicada: 13-abr-2022.
- [17] J. L. Pacheco Moya, “Inseguridad ciudadana: Problema social latente que afecta a nivel de Latinoamérica (revisión sistemática 2023-2025),” *Revista Tribunal*, vol. 6, pp. 45–62, Apr. 2026. <https://revistatribunal.org/index.php/tribunal/article/view/1135>.
- [18] G. Fernández L., “Estadística delictual a nivel Latinoamericano,” Informe Estadístico SUP: 138916, Biblioteca del Congreso Nacional de Chile, Asesoría Técnica Parlamentaria, Valparaíso, Chile, July

2023. https://obtienearchivo.bcn.cl/obtienearchivo?id=repositorio/10221/34471/1/BCN_GF_Estadistica_delictual_a_nivel_Latinoamericano.pdf.
- [19] W. J. Téllez Acosta, “Explorando la percepción de la delincuencia como principal problema en América Latina,” *Revista Internacional de Investigación en Ciencias Sociales*, vol. 11, pp. 195–208, Dec. 2015.
- [20] Instituto Nacional de Estadísticas (INE) and Subsecretaría de Prevención del Delito (SPD), “Encuesta nacional urbana de seguridad ciudadana 2025: Presentación de resultados nacional,” tech. rep., INE Chile, July 2026. <https://www.ine.gob.cl/docs/default-source/seguridad-ciudadana/publicaciones-y-anuarios/2025/nacional/sintesis-nacional---enusc-2025.pdf>.
- [21] Equipo Multimedia Emol, “El consumo audiovisual en chile se transforma: El 77 % prefiere plataformas de streaming y un 64 % opta por la tv abierta.” Emol.com, <https://www.emol.com/noticias/Tecnologia/2023/10/19/1110518/consumo-audiovisual-en-chile.html>, Oct. 2023.
- [22] Instituto Nacional de Estadística y Geografía (INEGI), “Reporte de resultados 33/25: Encuesta nacional de victimización y percepción sobre seguridad pública (envepe),” tech. rep., INEGI, Sept. 2025. https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2025/ENVIPE/ENVIPE_25_RR.pdf.
- [23] Instituto Nacional de Estadística y Geografía (INEGI), “Comunicado de prensa 40/26: Encuesta nacional de seguridad pública urbana (ensu).” INEGI, https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2026/ensu/ENSU2026_07_CP.pdf, July 2026.
- [24] Instituto Nacional de Estadística y Geografía (INEGI), “Reporte de resultados 19/26: Encuesta nacional sobre disponibilidad y uso de tecnologías de la información en los hogares (endutih),” tech. rep., INEGI, June 2026. https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2026/endutih/ENDUTIH_25_RR.pdf.
- [25] Instituto Nacional de Estadística e Informática (INEI), “Informe técnico: Estadísticas de seguridad ciudadana. semestre móvil: Setiembre 2025 – febrero 2026,” tech. rep., INEI, 2026. Encuesta Nacional de Programas Presupuestales (ENAPRES). <https://cdn.www.gob.pe/uploads/document/file/9895443/8077235-informe-tecnico-estadisticas-de-seguridad-ciudadana-del-semestre-movil-setiembre-2025-febrero-2026.pdf>.
- [26] PRODU, “La tv lineal concentra el mayor consumo de video en hogares de Perú y se complementa

- con el de plataformas digitales.” Produ.com, <https://www.produ.com/television/noticias/la-tv-lineal-concentra-el-mayor-consumo-de-video-en-hogares-de-peru-y-se-complementa-con-el-de> Sept. 2025. Estudio Cross Platform View, Kantar IBOPE Media.
- [27] Departamento Administrativo Nacional de Estadística (DANE), “Boletín técnico: Encuesta de convivencia y seguridad ciudadana (ecsc) 2024,” tech. rep., DANE, 2024. <https://www.dane.gov.co/files/operaciones/ECSC/bol-ECSC-2024.pdf>.
- [28] Infobae Tecno, “Tv o streaming: qué contenido consumen más los colombianos.” Infobae.com, <https://www.infobae.com/tecno/2025/09/12/tv-o-streaming-que-contenido-consumen-mas-los-colombianos/>, Sept. 2025. Datos basados en estudio de Kantar IBOPE Media Colombia.
- [29] Observatorio Ecuatoriano de Crimen Organizado, “Boletín anual de homicidios 2023,” tech. rep., OEEO, Ecuador, Apr. 2024. <https://oeco.padf.org/wp-content/uploads/2024/04/OEEO-BOLETIN-ANUAL-DE-HOMICIDIOS-2023.pdf>.
- [30] L. Dammert, G. Croci, and A. Frey, “¿por qué tanta violencia homicida en América Latina? Caracterizando el fenómeno y expandiendo su marco de interpretación,” Documento de Trabajo 94, Fundación Carolina, Feb. 2024. https://www.fundacioncarolina.es/wp-content/uploads/2024/02/DT_FC_94-1.pdf.
- [31] E. Anzola *et al.*, “Criminalidad en localidades de la primera línea del metro de Bogotá del 2018 al 2023,” *Revista Criminalidad*, vol. 66, no. 1, 2024. http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1794-31082024000300045.
- [32] F. Quezada, “Informe PDI revela las comunas con más encerronas en la región Metropolitana.” BioBioChile, <https://www.biobiochile.cl/noticias/nacional/region-metropolitana/2025/02/03/informe-pdi-revela-las-comunas-con-mas-encerronas-en-la-region-metropolitana.shtml>, Feb. 2025. Consultado en julio de 2026.
- [33] A. Luneke Reyes, “Inseguridad urbana, participación ciudadana y cuidado vecinal: la búsqueda por protección en los barrios,” *Revista INVI*, vol. 36, no. 102, pp. 302–327, 2021.
- [34] C. Fuenzalida, M. E. Donoso, S. Arancibia, D. Olazo, F. Ulloa, P. Carrillo, J. Sandoval, P. Pinto, M. P. Poblete, M. J. Domínguez, V. Vera, C. Corces, and D. Olguin, “Conjuntos habitacionales: Estudio de tipologías y casos.” Wiki Casiopea, Escuela de Arquitectura y Diseño, Pontificia Universidad Católica de Valparaíso, https://wiki.ead.pucv.cl/Conjuntos_Habitacionales.

- [_Estudio_de_tipolog%C3%ADas_y_casos](#), 2010. Última edición: 18 de mayo de 2010. Consultado en julio de 2026.
- [35] E. Suryady, S. Tan, and R. Hidayat, “Design and implementation of gate control and visitor management system for residential areas,” *Journal of Systems Engineering and Information Technology*, vol. 5, no. 2, pp. 45–52, 2021.
- [36] K. Foster and L. Davis, “Physical access control and human vulnerabilities in gated residential environments,” *Journal of Applied Security Research*, vol. 15, no. 3, pp. 310–328, 2020.
- [37] M. Hasan, M. Islam, and M. Rahman, “Automated smart gate and visitor log management system for secured communities,” *International Journal of Computer Applications*, vol. 175, no. 12, pp. 18–24, 2020.
- [38] A. Kumar, P. Sharma, and R. Gupta, “Limitations of manual entry systems and transition to automated gate security,” in *International Conference on Smart Electronics and Communication (ICOSEC)*, pp. 112–118, IEEE, 2021.
- [39] A. Khanna and R. Anand, “Iot based smart parking system,” pp. 266–270, 2016.
- [40] T. Lin, H. Rivano, and F. Le Mouél, “A survey of smart parking solutions,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 12, pp. 3229–3253, 2017.
- [41] B. C. Welsh and D. P. Farrington, “Situational crime prevention,” in *The Oxford Handbook of Crime Prevention*, pp. 100–125, Oxford University Press, 2012.
- [42] S. Du, M. Ibrahim, M. Shehata, and W. Badawy, “Automatic license plate recognition (alpr): A state-of-the-art review,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 2, pp. 311–325, 2012.
- [43] Fundación Endesa, “Smart city: ¿qué es una ciudad inteligente?.” Endesa Educa, <https://www.fundacionendesa.org/es/educacion/endesa-educa/recursos/smart-city>, 2026. Accedido el 1 de agosto de 2026.
- [44] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, “Internet of things for smart cities,” *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.
- [45] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: A survey on enabling technologies, protocols, and applications,” *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.
- [46] A. Sharma, P. Singh, and R. Kumar, “Smart building automation and security management using ai and iot: A review,” *Journal of Building Engineering*, vol. 43, p. 102821, 2021.

- [47] C. Nikhil, K. Shalini, M. Abrar, M. S. Velpuru, G. B. Regulwar, and G. G. Regulwar, “Dynamic parking system: A blend of automation and intelligence,” in *IEEE Nexa-Tech - International Conference on Advanced Technologies and Innovations (ICATI)*, pp. 1–6, IEEE, 2026.
- [48] D. L. H. Anh, A. C. Quang, A. N. Hong, A. K. Duc, and M. N. Truong, “Design and implementation of visitor management system using mobile application for apartment,” in *Proceedings of the 9th International Conference on Cloud Computing and Internet of Things (CCIoT 2024)*, (Hanoi, Vietnam), pp. 73–82, ACM, Nov. 2024.
- [49] D. Kennedy and R. Hupp, “Apartment security and litigation: Key issues,” *Security Journal*, vol. 11, pp. 21–28, 1998.
- [50] N. Manzolillo, “Apartment reception: Should you hire a front desk attendee?,” ButterflyMX Blog, <https://butterflymx.com/blog/apartment-reception/>, 2024. Consultado en julio de 2026.
- [51] D. Henry, A. Dillon, E. Paul, S. Nick, D. William, W. Elisabeth, and M. Michael, “ESP32-CAM as a programmable camera research platform,” *Electronic Imaging*, vol. 34, no. 7, pp. 232–1–232–6, 2022.
- [52] V. Rusyn, A. Sambas, and C. Skiadas, “Security access using simple RFID reader and Arduino UNO: A study case,” in *Lecture Notes in Networks and Systems*, pp. 193–202, 2022.
- [53] Arduino, “UNO R3 — arduino documentation.” <https://docs.arduino.cc/hardware/uno-rev3/>, 2024. Consultado en julio de 2026.
- [54] G. di Pasco, “SG90 servo characterization.” ResearchGate, <https://www.researchgate.net/publication/353754375>, 2021. Consultado en julio de 2026.
- [55] D. Sarma, A. Bora, and A. Bhagat, “Automatic license plate detection and recognition system for security purposes,” in *2023 IEEE Guwahati Subsection Conference (GCON)*, pp. 1–6, IEEE, 2023.
- [56] B. Hongliang and L. Changping, “A hybrid license plate extraction method based on edge statistics and morphology,” in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR 2004)*, vol. 2, pp. 831–834, IEEE, 2004.
- [57] Y. Qiu, M. Sun, and W. Zhou, “License plate extraction based on vertical edge detection and mathematical morphology,” in *2009 International Conference on Computational Intelligence and Software Engineering*, pp. 1–5, IEEE, 2009.
- [58] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and E. Kayafas, “A license plate-

- recognition algorithm for intelligent transportation system applications,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 3, pp. 377–392, 2006.
- [59] O. Bulan, V. Kozitsky, P. Ramesh, and M. Shreve, “Segmentation-and annotation-free license plate recognition with deep localization and failure identification,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 9, pp. 2351–2363, 2017.
- [60] Z. Selmi, M. Ben Halima, and A. M. Alimi, “Deep learning system for automatic license plate detection and recognition,” in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 1, pp. 1132–1137, IEEE, 2017.
- [61] T. E. de Campos, B. R. Babu, and M. Varma, “Character recognition in natural images,” in *Proceedings of the International Conference on Computer Vision Theory and Applications*, (Lisbon, Portugal), Feb. 2009.
- [62] C.-T. Hsieh, Y.-S. Juan, and K.-M. Hung, “Multiple license plate detection for complex background,” in *19th International Conference on Advanced Information Networking and Applications (AINA ’05)*, vol. 2, pp. 389–392, IEEE, 2005.
- [63] M. Chaman, A. El Maliki, N. Jariri, H. Dahou, H. Laâmari, and A. Hadjoudja, “Enhanced deep neural network-based vehicle detection system using yolov11 for autonomous vehicles,” in *2025 5th International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, pp. 1–6, IEEE, 2025.
- [64] S. Rasheed, A. Naeem, and O. Ishaq, “Automated number plate recognition using hough lines and template matching,” in *Proceedings of the World Congress on Engineering and Computer Science 2012*, vol. 1, (San Francisco, USA), pp. 199–203, 2012.
- [65] N. Kharina and S. Chernyadyev, “Software for car license plates recognition with minimal computing resources,” in *2022 24th International Conference on Digital Signal Processing and its Applications (DSPA)*, IEEE, 2022.
- [66] M. Dixit, S. C, R. Doss, S. Loke, and M. Saleemdurai, “Smart parking with computer vision and iot technology,” in *2020 International Conference on Telecommunications and Signal Processing (TSP)*, pp. 170–174, IEEE, 2020.
- [67] H. Shi and D. Zhao, “License plate recognition system based on improved yolov5 and gru,” *IEEE Access*, vol. 11, pp. 10429–10439, 2023.
- [68] R. Laroca, L. A. Zanlorensi, G. R. Gonçalves, E. Todt, W. R. Schwartz, and D. Menotti, “An

- efficient and layout-independent automatic license plate recognition system based on the YOLO detector,” *IET Intelligent Transport Systems*, vol. 15, no. 4, pp. 483–503, 2021.
- [69] I. Tilkari, A. Sarda, K. Deokar, N. Mukhopadhyay, and K. Kundu, “VehicleTracker: A real-time ALPR system for multi-point vehicle tracking,” in *2026 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, pp. 1–6, IEEE, 2026.
- [70] S. V. Srikanth, P. J. Pramod, K. P. Dileep, S. Tapas, M. U. Patil, and S. C. Babu N, “Design and implementation of a prototype smart parking (spark) system using wireless sensor networks,” in *2009 International Conference on Advanced Information Networking and Applications Workshops*, pp. 568–573, 2009.
- [71] Y.-Y. Chu and K.-H. Liu, “Iot in vehicle presence detection of smart parking system,” in *2nd IEEE Eurasia Conference on IOT, Communication and Engineering 2020*, pp. 56–61, 2020.
- [72] Edifito, “Software online de administración de condominios,” 2026.
- [73] Comunidad Feliz, “Plataforma de administración de edificios y condominios,” 2026.
- [74] Kastor Software, “Plataforma de administración de comunidades y gastos comunes,” 2026.
- [75] ComunidadActiva, “Software de condominios en chile: Gastos comunes, accesos, pagos,” 2026.
- [76] SafeCard Access Technologies, “Control de accesos inteligente y seguro,” 2026.
- [77] Federal Access, “Control de acceso para condominios y empresas,” 2026.
- [78] Smartki, “Control de acceso de condominios,” 2026.
- [79] Citofon, “Detección de patentes: Control de acceso,” 2026.
- [80] VisitPass, “Control de acceso inteligente para condominios,” 2026.
- [81] ZKTeco Latinoamérica, “Zkteco latinoamérica: Control de acceso, biometría y automatización,” 2026.
- [82] Proptia, “License plate recognition system: Auto lpr & anpr solutions,” 2026.
- [83] PLACA.AI, “License plate recognition for HOA and gated communities,” 2026.
- [84] NVIDIA Corporation, “NVIDIA Jetson Nano Developer Kit.” <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>, 2024.
- [85] Raspberry Pi Foundation, “Raspberry Pi 4.” <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>, 2024.

- [86] Google, “Coral USB Accelerator.” <https://coral.ai/products/accelerator/>, 2024.
- [87] Ultralytics, “Yolov5 by ultralytics.” <https://github.com/ultralytics/yolov5>, 2020. Accedido: junio de 2025.
- [88] Ultralytics, “Yolov8 - documentación oficial.” <https://docs.ultralytics.com/es/models/yolov8/#overview>, 2023. Accedido: junio de 2025.
- [89] Roboflow, “Yolov4-tiny model.” <https://roboflow.com/model/yolov4-tiny>, 2021. Accedido: junio de 2025.
- [90] S. Sen, “Mobilenetv2-ssd: Implementation of mobilenetv2-ssd object detection model.” <https://github.com/saunack/MobileNetv2-SSD>, 2019. Accedido: junio de 2025.
- [91] O. Team, “Opencv library.” <https://github.com/opencv/opencv/tree/master>, 2000. Accedido: junio de 2025.
- [92] Ultralytics, “Yolo11 - documentación oficial.” <https://docs.ultralytics.com/models/yolo11>, 2024. Accedido: agosto de 2026.
- [93] ArduinoModules.info, “KY-032 Infrared Obstacle Avoidance Sensor Module.” <https://arduinomodules.info/ky-032-infrared-obstacle-avoidance-sensor-module/>, 2025. Consultado en junio de 2025.
- [94] Interlink Electronics, “FSR 402 Force Sensing Resistor.” <https://www.interlinkelectronics.com/fsr-402>, 2025. Consultado en junio de 2025.
- [95] Allegro Microsystems, “A3144 Hall Effect Sensor Datasheet.” <https://www.alldatasheet.es/datasheet-pdf/pdf/55092/ALLEGRO/A3144.html>, 2025. Consultado en junio de 2025.
- [96] SparkFun Electronics, “HC-SR04 Ultrasonic Sensor Datasheet.” <https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>, 2025. Consultado en junio de 2025.
- [97] Wi-Fi Alliance, “Discover Wi-Fi.” <https://www.wi-fi.org/>, 2025. Consultado en junio de 2025.
- [98] LoRa Alliance, “LoRaWAN Specification and Technical Resources.” <https://lora-alliance.org/resource-hub/>, 2025. Consultado en junio de 2025.
- [99] Connectivity Standards Alliance, “Zigbee Technology Overview.” <https://csa-iot.org/all-solutions/zigbee/>, 2025. Consultado en junio de 2025.
- [100] Thread Group, “What is Thread?.” <https://www.threadgroup.org/What-is-Thread>, 2025. Consultado en junio de 2025.

- [101] Bluetooth Special Interest Group, “Bluetooth Technology Overview.” <https://www.bluetooth.com/>, 2025. Consultado en junio de 2025.
- [102] Oracle Corporation, “Página Oficial de MySQL.” <https://www.mysql.com/>, 2024.
- [103] The PostgreSQL Global Development Group, “Página Oficial de PostgreSQL.” <https://www.postgresql.org/>, 2024.
- [104] MariaDB Corporation, “Página Oficial de MariaDB.” <https://mariadb.org/>, 2024.
- [105] D. Richard Hipp, “Página Oficial de SQLite.” <https://www.sqlite.org/>, 2024.
- [106] Oracle Corporation, “Página Oficial de Oracle Database.” <https://www.oracle.com/database/>, 2024.
- [107] MongoDB, Inc., “Página Oficial de MongoDB.” <https://www.mongodb.com/>, 2024.
- [108] Google, “Página Oficial de Firebase.” <https://firebase.google.com/>, 2024.
- [109] huang77, “huaung77 object detection dataset.” Roboflow Universe, <https://universe.roboflow.com/huang77-8ldff/huaung77/dataset/1>, 2024.
- [110] Roboflow Universe Projects, “License plate recognition dataset.” Roboflow Universe, <https://universe.roboflow.com/roboflow-universe-projects/license-plate-recognition-rxg4e/dataset/11>, 2025.