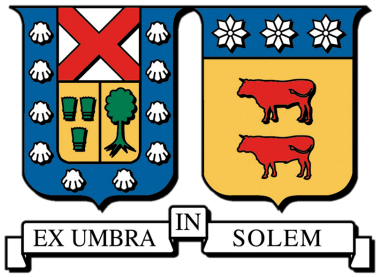
UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA

DEPARTAMENTO DE ELECTRÓNICA

SANTIAGO – CHILE



PLATAFORMA IOT PARA EL CONTROL DE ACCESOS VEHICULARES Y GESTIÓN DE ESTACIONAMIENTOS EN EDIFICIOS RESIDENCIALES

NILSSON IGNACIO ACEVEDO PEÑALOZA

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL TELEMÁTICO

PROFESOR GUÍA
DRA. PATRICIA FRANCO TROYA

PROFESOR CORREFERENTE
DR. NICOLÁS SEBASTIÁN GÁLVEZ RAMÍREZ

JUNIO 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

Título del trabajo: PLATAFORMA IOT PARA EL CONTROL DE ACCESOS VEHICULARES Y GESTIÓN DE ESTACIONAMIENTOS EN EDIFICIOS RESIDENCIALES

Nombre del candidato(a): Nilsson Ignacio Acevedo Peñaloza

Carrera / Grado: Ingeniera Civil Telemática

Campus: San Joaquín **Departamento:** Departamento de Electrónica

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Patricia Franco Troya, 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

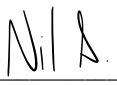
Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

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

Fecha: 25 de Junio del 2026 **Firma:** _____

Estudiante o Candidato(a):

Fecha: 25 de Junio del 2026 **Firma:** 

Agradecimientos

En primer lugar, quiero agradecer profundamente a mi familia: Marisol, Luis y Matías, por apoyarme desde el primer minuto y por ser quienes me motivaron a estudiar esta carrera. Ustedes saben que, tanto al inicio como durante el camino, muchas veces pensé en rendirme, sin embargo, siempre creyeron en mí, me acompañaron en cada momento difícil y me dieron la fuerza necesaria para seguir adelante. Gracias a su amor, apoyo y confianza, hoy puedo decir que soy un profesional. Espero que estén tan orgullosos de mí como yo lo estoy de ustedes. Agradezco a Dios por haberme dado una familia tan maravillosa.

También quiero agradecer a mi perrito Bobby, quien llegó a mi vida en uno de los años más difíciles de mi carrera, justo cuando dejábamos el plan común para comenzar a aprender las especialidades propias de la carrera. Desde entonces, ha sido y sigue siendo un gran apoyo en mi vida, acompañándome de una forma única e incondicional.

Sin dejar de lado a mis profesores, quiero agradecer especialmente a la profesora Patricia Franco, por creer en nuestro proyecto y por su apoyo durante todo este proceso. Asimismo, agradezco a mis profesores de carrera: Berioska Contreras, Nicolás Torres, Nicolás Gálvez, Rodrigo Muñoz y Sebastián Contreras, quienes siempre creyeron en mí y me enseñaron, durante estos seis años, no solo los conocimientos de la profesión, sino también los valores necesarios para convertirme en un buen profesional.

También quiero agradecer a mis amigos de carrera Benjita, Diego, Benja Mena, Rodri, Roder, Nacho, Tomy, Carrasco, Drigo, Edu y a mis compañeros de memoria Diego Prokes y Diego Maldonado, por estar conmigo en todo momento, por compartir alegrías, desafíos, buenos y malos momentos, y por ser parte importante de este camino. Su amistad y compañía hicieron que este proceso fuera mucho más llevadero y significativo.

Finalmente, quiero dedicar estas palabras a todos quienes están persiguiendo un sueño. Este camino me enseñó que las metas no siempre se alcanzan de manera fácil ni rápida, pero con constancia, perseverancia y fe en uno mismo, los sueños sí se cumplen. Cada esfuerzo, cada noche difícil y cada obstáculo superado valen la pena cuando se llega al final y se mira hacia atrás con orgullo. Hoy cierro una etapa importante de mi vida con gratitud, emoción y la certeza de que todo sacrificio tuvo sentido.

Dedicatoria

En memoria de quien me acompaña eternamente desde el cielo,
Rosa Agustina Plaza Abarca, mi lela.

Resumen

En muchas comunidades residenciales, el control de acceso vehicular y la administración de estacionamientos de visita continúan dependiendo de procedimientos manuales (controles remotos, citófono y registros en papel), lo que introduce demoras, baja trazabilidad y vulnerabilidades ante accesos no autorizados. Esta memoria presenta ResiPark, un sistema ciberfísico orientado a mejorar la seguridad y la gestión operativa en condominios y pasajes con portón eléctrico mediante la integración de visión por computador, sensorización IoT y servicios digitales. La solución se estructura en un modelo por capas: adquisición de datos (cámara IP en el acceso y sensores de ocupación por plaza), comunicación (LoRa y red IP), soporte (API y persistencia) y aplicaciones (panel web para conserjería y app móvil para residentes). En el núcleo del sistema se implementa un pipeline de percepción que detecta la presencia de vehículos, lee patentes y valida autorización (residente o visita con reserva vigente) antes de habilitar la apertura, manteniendo registro de eventos para auditoría. El prototipo fue desplegado y probado en un entorno real de baja escala, demostrando viabilidad con infraestructura limitada y sin intervención del motor del portón, además de evidenciar restricciones relevantes (dependencia de conectividad, sensibilidad a condiciones de captura y cobertura parcial de los casos borde). Finalmente, se discuten mecanismos de mitigación para escenarios adversos y se proponen líneas de trabajo para robustez, escalabilidad e integración con entornos operacionales de mayor demanda.

Palabras clave: comunidades inteligentes; control de acceso vehicular; IoT; LoRa; reconocimiento de patentes; sistemas ciberfísicos; trazabilidad; visión por computador.

Abstract

In many residential communities, vehicle access control and visitor parking management still rely on manual procedures (handheld remotes, intercom calls, and paper logs), which increase delays, reduce traceability, and expose vulnerabilities to unauthorized entries. This thesis presents ResiPark, a cyber-physical system designed to strengthen security and improve operational management in condominiums and gated streets by integrating computer vision, IoT sensing, and digital services. The solution follows a layered architecture: data acquisition (IP cameras at the gate and per-space occupancy sensors), communication (LoRa and IP networking), support services (API and persistence), and user applications (a web dashboard for guards and a mobile app for residents). At its core, the system implements a perception-and-decision pipeline that detects vehicle presence, reads license plates, verifies authorization (registered residents or valid visitor reservations), and only then triggers gate opening while logging events for auditability. A functional prototype was deployed and evaluated in a real low-scale setting, confirming feasibility under limited infrastructure and without modifying the gate motor, while also revealing practical constraints (network dependence, capture-condition sensitivity, and incomplete coverage of edge cases). The work concludes by outlining mitigation strategies and future directions to improve robustness, scalability, and integration into higher-demand operational contexts.

Keywords: cyber-physical systems; intelligent communities; IoT; license plate recognition; LoRa; traceability; vehicle access control; computer vision.

Índice general

1. Introducción y contextualización del problema **1**
1.1. Problemática . . . 1
1.1.1. Escenario actual de accesos vehiculares en edificios residenciales . . . 2
1.1.2. ¿Cuánto tiempo toma entrar a un edificio residencial? . . . 3
1.1.3. ¿Son costosos los sistemas de parking con detección de patentes en Chile? 4
1.1.4. ¿Cuál es la infraestructura en común en recintos con comunidades? . 5
1.2. Propósito . . . 6
1.3. Objetivos . . . 7
1.3.1. Objetivo General . . . 7
1.3.2. Objetivos Específicos . . . 7
1.3.3. Contribuciones . . . 8
1.4. Estructura de la memoria . . . 8

2. Estado del Arte **9**
2.1. Seguridad vehicular residencial en comunidades inteligentes: contexto, tecnologías y brechas . . . 10
2.1.1. Comunidades inteligentes, edificios inteligentes y seguridad vehicular residencial . . . 10
2.1.2. Sistemas ciberfísicos y redes de sensores inalámbricas como marco conceptual . . . 11
2.1.3. Arquitecturas de control de acceso vehicular automatizado . . . 12
2.1.4. Tecnologías de comunicación para WSN residenciales: criterios de diseño y comparación entre LoRa, Zigbee y WiFi . . . 13
2.1.5. Detección de ocupación y gestión de plazas: smart parking basado en WSN/IoT . . . 16
2.1.6. Reconocimiento de patentes y percepción en accesos: de ALPR modular a modelos fundacionales . . . 18
2.1.7. Interfaces móviles, paneles operativos y servicios centrales . . . 20
2.1.8. Ecosistema chileno actual y oportunidad de integración sobre plataformas existentes . . . 22
2.1.9. Síntesis de brechas y conexión con la propuesta de la memoria . . . 23

3. Diseño de la solución **25**
3.1. Modelo de Capas . . . 25
3.1.1. Obtención de datos . . . 25
3.1.2. Comunicación . . . 26
3.1.3. Soporte . . . 26
3.1.4. Aplicación . . . 27
3.2. Requisitos del Sistema . . . 27
3.2.1. Requerimientos Funcionales . . . 27

3.2.2. Requerimientos de Interfaces . . . 28
3.2.3. Requerimientos de Ambiente . . . 28
3.3. Arquitectura del Sistema . . . 29
3.3.1. Diagrama de Contexto . . . 29
3.3.2. Diagrama de Arquitectura . . . 30
3.4. Diseño de interfaces . . . 30
3.4.1. Modelo de navegación . . . 31
3.4.2. Prototipo de Interfaces Usuarias . . . 31
3.5. Diseño de la infraestructura electrónica para adquisición de datos . . . 36
3.5.1. Diseño de comunicación: topología, gateway y flujo extremo a extremo 38
3.5.2. Flujo operativo de datos y control . . . 38
3.5.3. Diagrama de la arquitectura y diseño físico . . . 39
3.6. Simulación de la tecnología de comunicación para M2 . . . 41
3.6.1. Diseño experimental . . . 42
3.6.2. Robustez del enlace en función de la distancia . . . 42
3.6.3. Compromiso entre robustez y costo energético . . . 43
3.6.4. Escalamiento con la cantidad de nodos . . . 44
3.6.5. Latencia del enlace y comportamiento bajo sobrecarga . . . 45
3.6.6. Pérdidas observadas bajo incremento de carga . . . 47
3.6.7. Síntesis de resultados de simulación . . . 48
3.6.8. Estimación de autonomía energética del nodo LoRa . . . 49
3.6.9. Estimación conceptual de latencia en enlace punto a punto para un nodo único . . . 51
3.6.10. Discusión y justificación de la decisión tecnológica . . . 52
3.6.11. Evaluación de la capa de comunicación . . . 54
3.6.12. Discusión y justificación de la decisión tecnológica . . . 56

4. Implementación y Despliegue de la Solución 58
4.1. Implementación de la Arquitectura . . . 58
4.1.1. Contexto General del Prototipo . . . 58
4.1.2. Descripción de Componentes . . . 58
4.2. Procesamiento de datos . . . 59
4.2.1. Infraestructura de adquisición de datos y trazabilidad operativa . . . 59
4.3. Infraestructura electrónica y montaje en terreno . . . 61
4.3.1. Nodo de sensado de ocupación (TX): sensor ultrasónico y transmisor LoRa . . . 61
4.3.2. Nodo receptor/gateway (RX): recepción LoRa y publicación vía WiFi 62
4.3.3. Nodo de borde para actuación: Raspberry Pi 4B y relé de apertura . 62
4.3.4. Nodo de borde para captura: Raspberry Pi y cámara para entrada de IA 63
4.3.5. Justificación de decisiones de diseño . . . 64
4.4. Despliegue de la solución . . . 65
4.4.1. Validación del sistema en un entorno real . . . 65
4.4.2. Testing . . . 67
4.5. Gestión de riesgos, validación en terreno y casos borde . . . 69
4.5.1. Controles a nivel de arquitectura . . . 69

4.5.2. Mitigación principal en operación: lógica escalonada de autorización . 69
4.5.3. Pruebas controladas del comportamiento esperado 70
4.5.4. Casos borde observados y degradación a human-in-the-loop 71

5. Conclusiones y Perspectivas **75**
5.1. Conclusiones Generales 75
5.2. Caso de estudio: Inmobiliaria Norte Verde 76
5.3. Caso de estudio: Pasaje con portón eléctrico 77
5.4. Perspectivas y trabajo a futuro 77

Índice de figuras

1.1. Ingreso Vehicular del edificio de la inmobiliaria Norte Verde. 2
1.2. Diagrama de flujo del ingreso vehicular de un residente en un edificio no automatizado. 3
1.3. Diagrama de flujo del ingreso vehicular de un visitante en un edificio no automatizado. 5

2.1. Arquitectura por capas para sistemas integrados de *smart parking*. 15
2.2. Arquitectura general de un sistema de *smart parking* basado en WSN. 17
2.3. Ejemplo de *pipeline* ALPR modular en tiempo real. 20

3.1. Diagrama de contexto del sistema ResiPark. 29
3.2. Arquitectura general del sistema en cuatro capas. 30
3.3. Prototipo de inicio de sesión de la plataforma web para conserjería. 32
3.4. Prototipo del panel principal de monitoreo para conserjería. 33
3.5. Prototipo del módulo de historial de accesos vehiculares. 34
3.6. Prototipo del módulo de administración de patentes autorizadas. 34
3.7. Prototipos principales de la aplicación móvil para residentes: (a) inicio de sesión, (b) menú principal y (c) reserva de estacionamiento de visita. 35
3.8. Zonificación conceptual del despliegue físico del sistema. 37
3.9. Flujo de los sensores en el estacionamiento 38
3.10. Flujo de datos. 39
3.11. Diseño de la Arquitectura. 40
3.12. DER del enlace LoRa en función de la distancia entre nodo y *gateway* para distintas configuraciones de *spreading factor*. 43
3.13. Energía promedio consumida por byte transmitido según *spreading factor*. . . 44
3.14. DER del enlace LoRa al incrementar la cantidad de nodos sensores en la topología de estacionamientos. 45
3.15. Latencia media del enlace LoRa bajo distinta carga, expresada mediante el intervalo entre transmisiones. 46
3.16. DER del enlace LoRa bajo distinta carga, expresada mediante el intervalo entre transmisiones. 47
3.17. Colisiones y paquetes débiles observados bajo distinta carga para configuraciones representativas del enlace LoRa. 48
3.18. Autonomía estimada de un nodo LoRa alimentado con batería de 2500 mAh según la frecuencia con que despierta para sensar y transmitir el estado de ocupación. 50
3.19. Latencia media del enlace LoRa P2P para un único nodo sensor al variar la distancia respecto del *gateway*. 51

4.1. Nodo de sensado de ocupación (TX) utilizado para medir presencia vehicular y transmitir el estado por LoRa. 61
4.2. Nodo receptor/gateway (RX) encargado de recibir estados por LoRa y enviarlos al backend mediante WiFi. 62
4.3. Nodo de borde para actuación: ejecución del pulso de apertura condicionado por una instrucción validada desde el backend. 63
4.4. Subsistema de captura de video para IA: configuración usada en terreno y diseño compatible con cámara IP de exterior. 64
4.5. Entorno real de despliegue del sistema en un pasaje residencial. En rojo se destaca la caja de componentes y la cámara IP. 66
4.6. Secuencia de prueba con vehículo en movimiento utilizada para validar el flujo de detección y registro de ingresos. 67
4.7. Vista de la caja impresa en 3D que alberga el sensor ultrasónico HC-SR04 para detectar ocupación de estacionamiento. 68
4.8. Escenario de prueba: una persona sostiene una patente frente a la cámara. El sistema no abre el portón al no detectar un vehículo. 70
4.9. Salida del cliente en Raspberry Pi: detecciones de vehículo y fallos controlados durante la verificación de patente, sin accionar la apertura. 71
4.10. Caso borde ilustrativo: patente dibujada/adulterada. La detección puede ocurrir, pero la lectura OCR suele degradarse, por lo que no se considera evidencia suficiente para apertura automática. 72
4.11. Caso borde ilustrativo: misma patente, color distinto al registrado. Este patrón es compatible con suplantación, por lo que se recomienda incorporar atributos adicionales y degradar a revisión manual. 73

Índice de tablas

2.1. Comparación de tecnologías de comunicación relevantes para una WSN de estacionamientos residenciales. 14
2.2. Matriz de decisión para el tramo sensor–gateway en la propuesta de esta memoria. 15
2.3. Comparación de enfoques para detección de ocupación y gestión de disponibilidad en estacionamientos residenciales. 18
2.4. Comparación de enfoques ALPR relevantes para control de acceso vehicular en entornos residenciales. 19
2.5. Distribución de roles, interfaces y requerimientos operativos en sistemas residenciales de control vehicular y gestión de estacionamientos. 21
2.6. Posicionamiento de la propuesta frente al ecosistema chileno actual. 22
2.7. Relación entre las brechas identificadas en el estado del arte y las decisiones de diseño adoptadas en la propuesta. 24

3.1. Criterio de selección del *spreading factor* para el módulo M2. 52
3.2. Comparación funcional de tecnologías de comunicación consideradas en la arquitectura. 53
3.3. Síntesis comparativa de las decisiones de comunicación en la arquitectura propuesta. 54
3.4. Comparación de tecnologías para el tramo sensor–gateway del módulo M2. . 56
3.5. Síntesis comparativa de las decisiones de comunicación en la arquitectura propuesta. 57

Capítulo 1

Introducción y contextualización del problema

En los últimos años, el desarrollo de smart cities, smart communities y edificios inteligentes ha impulsado la incorporación de tecnologías digitales en entornos urbanos y residenciales. Estas tecnologías buscan optimizar procesos, mejorar la calidad de vida y fortalecer la operación de infraestructuras cotidianas. En este contexto, herramientas basadas en Inteligencia Artificial (IA), Visión por Computador, Internet de las Cosas (IoT) y plataformas de servicios han comenzado a emplearse en problemas de vigilancia, trazabilidad, control de accesos y gestión eficiente de recursos compartidos [1, 2, 3, 4].

Estas tendencias adquieren especial relevancia en ciudades densamente pobladas como Santiago de Chile, donde la seguridad en entornos residenciales se ha transformado en una preocupación operativa permanente. En este escenario, los accesos vehiculares y el uso de estacionamientos de visita dejan de ser tareas meramente administrativas y pasan a constituir un problema de seguridad, coordinación y trazabilidad, especialmente en comunidades que aún operan mediante procesos manuales [5, 6, 7].

1.1. Problemática

La problemática central radica en que la gestión de accesos y estacionamientos en edificios residenciales sigue siendo predominantemente manual, dependiendo del criterio humano y de herramientas poco integradas. Esto genera tres consecuencias negativas principales identificadas en el levantamiento del problema:

- **Vulnerabilidad en la seguridad:** La validación de ingreso suele depender de la observación del conserje, del uso de controles remotos o de la autorización verbal del residente. La ausencia de un mecanismo automatizado de verificación favorece errores, accesos indebidos y situaciones de ingeniería social [8, 9, 4].
- **Gestión ineficiente de recursos comunes:** Los estacionamientos de visita son un recurso escaso y conflictivo. Cuando no existe monitoreo de ocupación ni correlación entre reservas, tiempo de uso y presencia efectiva del vehículo, se facilita la ocupación prolongada o indebida de estos espacios, generando fricción entre residentes y administración [10, 11, 12].
- **Falta de trazabilidad:** Los registros manuales o semimanuales dificultan reconstruir con precisión quién ingresó, cuándo lo hizo y bajo qué autorización. Ante un incidente,

la recuperación de evidencia se vuelve lenta, ambigua y dependiente de la memoria o criterio del personal [8, 13, 9].

1.1.1. Escenario actual de accesos vehiculares en edificios residenciales

Para contextualizar el problema en un entorno real, se consideró como referencia un edificio residencial asociado a la inmobiliaria Norte Verde, ubicado en Av. Rondizzoni, Santiago Centro. Este caso permite observar con mayor claridad tres dimensiones del problema: los cuellos de botella temporales en el ingreso vehicular, las barreras económicas de las soluciones comerciales disponibles y la infraestructura base ya presente en edificios residenciales de similares características [14].



Figura 1.1: Ingreso Vehicular del edificio de la inmobiliaria Norte Verde.

Más que describir un caso excepcional, este escenario resulta representativo de una parte importante del parque residencial urbano: comunidades con portón motorizado, conserjería, cámaras de vigilancia y conectividad básica, pero donde la lógica de autorización y control sigue dependiendo principalmente de procedimientos manuales o escasamente integrados, como se muestra en la Figura 1.1.

1.1.2. ¿Cuánto tiempo toma entrar a un edificio residencial?

El proceso de ingreso a un recinto residencial del caso de estudio varía significativamente dependiendo de si el usuario es un residente o un visitante. En edificios que operan con esquemas tradicionales, los tiempos de espera y la seguridad del proceso quedan condicionados por la disponibilidad del personal, la posesión de dispositivos físicos de apertura y la comunicación sincrónica entre conserjería y residentes [14, 8].

A partir de observaciones exploratorias realizadas en el edificio utilizado como caso de estudio, se identificó que el proceso de ingreso vehicular presenta tiempos variables según el tipo de usuario y el mecanismo de autorización disponible. En el caso de residentes que cuentan con control remoto, la apertura del portón suele completarse en un intervalo aproximado de 5 a 15 segundos. No obstante, cuando el residente no dispone del dispositivo de apertura y debe recurrir a la validación manual por parte de conserjería, el proceso se extiende debido a la necesidad de identificar al usuario y autorizar la apertura, alcanzando tiempos cercanos a 20–40 segundos. Para visitantes, el flujo resulta más lento, ya que requiere la intervención del conserje, el contacto con el residente mediante citófono y la confirmación verbal de autorización. En estos casos, se observó que el ingreso puede tomar entre 60-100 segundos, dependiendo principalmente de la disponibilidad del personal de conserjería y del tiempo de respuesta del residente contactado.

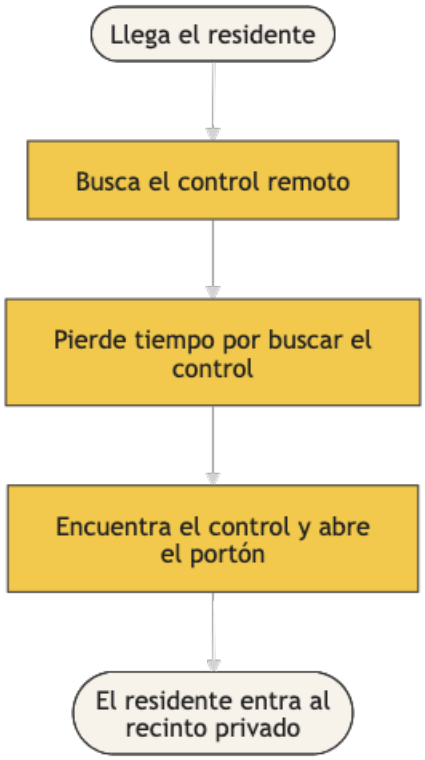


Figura 1.2: Diagrama de flujo del ingreso vehicular de un residente en un edificio no automatizado.

En el caso de los **residentes**, el flujo actual depende en gran medida de la posesión física de un control remoto, de una llamada telefónica o de la identificación visual por parte del

conserje. Como se observa en la Figura 1.2, el proceso incluye tiempos muertos asociados a la búsqueda del dispositivo de apertura. Si el residente no cuenta con el control, debe esperar a que el personal de conserjería valide su identidad, lo cual está sujeto a la disponibilidad y atención del personal en ese momento.

Por otro lado, el ingreso de **visitantes** presenta un flujo mucho más crítico y propenso a la congestión. Tal como se sintetiza en la Figura 1.3, la dependencia del citófono y de la intermediación humana crea un cuello de botella operativo:

- El visitante debe esperar a que el conserje esté disponible para atender.
- El conserje debe contactar al departamento, esperando que el residente conteste.
- Se requiere una confirmación verbal para autorizar la apertura.

Este procedimiento manual no solo incrementa los tiempos de espera, sino que además introduce ambigüedad en la validación del acceso. En la práctica, la decisión final puede quedar supeditada al criterio del conserje y a la calidad de la comunicación con el residente, debilitando la trazabilidad y aumentando la exposición a ingresos no verificados [8, 9].

1.1.3. ¿Son costosos los sistemas de parking con detección de patentes en Chile?

El mercado chileno ofrece soluciones para control de acceso vehicular con detección de patentes (LPR/ANPR), pero en edificios residenciales de tamaño medio el costo total suele percibirse como elevado debido a dos factores recurrentes: (i) muchos proveedores comercializan soluciones orientadas a control de acceso e infraestructura de uso intensivo, que requieren instalación e integración profesional, y (ii) existen modelos de contratación que transfieren el costo desde una inversión inicial (CAPEX) hacia pagos mensuales permanentes (OPEX).

En términos de precios publicados, Federal Access reporta rangos referenciales (2025) para barreras de estacionamiento considerando la tecnología e integración incluidas: una barrera con lector de patente integrado desde $1.400.000 + IVA, y una barrera de alta velocidad con integración completa desde $1.900.000 + IVA. En el mismo documento se indica que el precio final depende de variables como el modelo, la tecnología incluida, la integración con software y la complejidad de la instalación, por lo que estos valores deben interpretarse como pisos de referencia y no como costo total del proyecto (por ejemplo, sin desagregar obras civiles, canalizaciones, energía, red o puesta en marcha) [15].

En un modelo basado en suscripción, Citofon publica valores mensuales (UF + IVA) para integrar detección de patentes y, opcionalmente, apertura automática. En particular, el plan con detección de patentes y apertura, con arriendo de equipo, se lista en 6,68 UF + IVA mensuales [13]. A modo referencial, el Banco Central de Chile reporta que al 25 de febrero de 2026 el valor de la UF fue $39.773,62, por lo que 6,68 UF equivale aproximadamente a $265.688 mensuales (sin IVA) [16].

Finalmente, en un esquema basado en componentes e integración, proveedores como Access Chile se enfocan en suministro e instalación de barreras asociadas a sistemas electrónicos de control de acceso, incluyendo lectura de patente dentro del ecosistema de accionamiento.

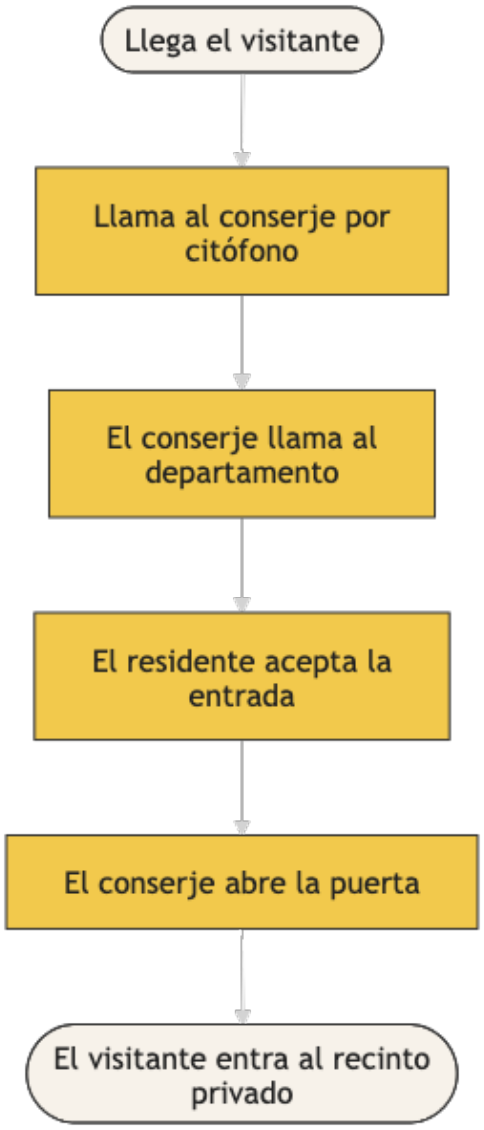


Figura 1.3: Diagrama de flujo del ingreso vehicular de un visitante en un edificio no automatizado.

Este enfoque es flexible, pero típicamente requiere diseño de ingeniería, integración y servicios técnicos especializados, lo que aumenta la variabilidad del costo total y suele obligar a cotizar caso a caso [17].

1.1.4. ¿Cuál es la infraestructura en común en recintos con comunidades?

A partir del levantamiento realizado sobre el caso de estudio y de la revisión de soluciones ya desplegadas en comunidades residenciales, se observa que una parte importante de los edificios en altura comparte una base tecnológica mínima sobre la cual es posible construir una solución incremental, sin reemplazar completamente la infraestructura existente [14, 8, 9, 17].

En términos generales, esa infraestructura común puede resumirse en los siguientes elementos:

1. **Acceso vehicular motorizado:** Portones eléctricos o barreras que pueden ser accionados mediante pulsos o controladores, lo que habilita automatización sin intervenir el motor principal.

2. **Sistema de videovigilancia (CCTV):** Presencia de cámaras de seguridad, muchas veces IP o analógicas convertibles, ubicadas en accesos y zonas comunes. Aunque en numerosos casos se usan solo para monitoreo pasivo, su ubicación es adecuada para tareas de percepción visual.

3. **Conectividad básica:** Disponibilidad de alimentación eléctrica continua y, en muchos casos, conexión a internet o red local en conserjería para labores administrativas.

4. **Personal humano de operación:** Presencia de conserjes o personal de acceso, lo que permite implementar una transición gradual hacia esquemas híbridos de operación *human-in-the-loop*.

Esta observación es relevante porque redefine el problema de implementación: la principal barrera no es la inexistencia total de infraestructura, sino la falta de una arquitectura que logre articular los componentes ya disponibles con nuevas capacidades de sensado, validación y trazabilidad.

1.2. Propósito

En la actualidad, aunque algunos edificios residenciales cuentan con sistemas automáticos para el control vehicular, la mayoría aún gestiona este proceso de forma manual, delegando la autorización de ingreso al criterio del personal de conserjería. Esta dependencia presenta diversas limitaciones, entre ellas, una validación poco rigurosa de los accesos, lo que puede derivar en la autorización de ingresos no justificados o incluso malintencionados.

Un problema recurrente es la posibilidad de que personas externas consigan persuadir al conserje para ingresar al recinto, sin contar con un mecanismo efectivo de verificación. Esta falta de validación sistemática abre la puerta a accesos indebidos que pueden poner en riesgo la seguridad de los residentes y vulnerar la confianza en la administración del edificio.

A su vez, el sistema actual dificulta una gestión eficiente del uso de los estacionamientos de visitas. Es frecuente que estos espacios se utilicen sin respetar las políticas internas establecidas, lo que genera conflictos entre los residentes y una ocupación inadecuada de los recursos comunes.

En conjunto, estas falencias evidencian la necesidad de una solución tecnológica que permita mejorar el control de accesos vehiculares y optimizar la administración de los espacios compartidos, contribuyendo así a una mayor seguridad, eficiencia y calidad de vida dentro de los complejos residenciales sin intervenir la infraestructura existente.

1.3. Objetivos

El proyecto busca construir un sistema inteligente de seguridad para edificios residenciales que reduzca vulnerabilidades ante accesos no autorizados y, a la vez, haga más eficiente y trazable el acceso vehicular de residentes y visitas. Para ello, la solución integra percepción automática en el acceso, sensado distribuido en estacionamientos y un backend que centraliza la validación, el registro de eventos y la generación de evidencias, de modo que las decisiones de apertura y control no dependan únicamente de validaciones manuales, sino de datos verificables y auditables.

En el marco de ese proyecto más amplio, el foco específico se concentra en el diseño de la arquitectura ciberfísica, la infraestructura de comunicación y la integración entre dispositivos de terreno, servicios centrales e interfaces de operación. En otras palabras, el aporte principal del trabajo no está en desarrollar aisladamente cada modelo o interfaz como productos autónomos, sino en articularlos dentro de una solución coherente, desplegable y trazable sobre infraestructura residencial existente.

1.3.1. Objetivo General

Diseñar e implementar una arquitectura modular para un sistema inteligente de seguridad vehicular residencial, orientada a integrar percepción automática, sensado IoT, comunicación de datos y servicios digitales sobre infraestructura preexistente.

1.3.2. Objetivos Específicos

- **Caracterizar el problema operativo y técnico** del control de accesos vehiculares y la gestión de estacionamientos de visita en edificios residenciales, identificando restricciones de integración, trazabilidad y escalabilidad.
- **Diseñar una arquitectura ciberfísica del sistema**, definiendo la interacción entre adquisición de datos, comunicación, soporte y aplicación para coordinar acceso vehicular, ocupación de plazas y evidencia operativa.
- **Desarrollar la red de comunicación del sistema**, evaluando alternativas tecnológicas para el tramo sensor–gateway y su integración con el backhaul IP del edificio.
- **Implementar un prototipo funcional** que integre dispositivos de borde, backend, persistencia e interfaces de usuario, permitiendo validar el flujo de detección, autorización, registro y monitoreo.
- **Validar el comportamiento del sistema en un entorno real**, analizando riesgos, controles de seguridad, casos borde y limitaciones del despliegue sobre infraestructura residencial existente.

1.3.3. Contribuciones

La contribución principal de esta memoria consiste en proponer y materializar una arquitectura integradora para seguridad vehicular residencial, donde convergen visión por computador, IoT, comunicación inalámbrica y servicios digitales dentro de un mismo flujo operativo auditable. Más que limitarse a un componente aislado, el trabajo articula el ingreso vehicular, la verificación de autorizaciones, el registro de eventos y la ocupación de estacionamientos de visita en una solución coherente y modular.

Desde la perspectiva del aporte técnico del autor, la memoria contribuye en tres niveles. Primero, formaliza el problema como un sistema ciberfísico, separando percepción, comunicación, soporte y aplicación. Segundo, diseña una infraestructura de comunicación adecuada para un entorno residencial con restricciones reales de cobertura e integración. Tercero, implementa y valida un prototipo funcional sobre infraestructura existente, demostrando que es posible avanzar hacia automatización y trazabilidad sin requerir reemplazos totales de la instalación base del recinto.

1.4. Estructura de la memoria

La estructura del presente documento se compone de la siguiente forma:

- En el Capítulo 1 se contextualiza el problema, se define la motivación del trabajo y se establecen el propósito, los objetivos y las contribuciones de la memoria.
- En el Capítulo 2 se revisa el estado del arte en control de acceso vehicular, smart parking, visión por computador y arquitecturas IoT/ciberfísicas aplicadas a entornos residenciales.
- En el Capítulo 3 se presenta el diseño de la solución propuesta, incluyendo el modelo por capas, la arquitectura del sistema y el diseño de la infraestructura electrónica y de comunicaciones.
- En el Capítulo 4 se aborda la implementación y el despliegue del prototipo en un entorno real, junto con su validación, análisis de riesgos y discusión de casos borde.

Capítulo 2

Estado del Arte

El control de acceso vehicular y la gestión de estacionamientos en entornos residenciales constituyen procesos relevantes para la seguridad y la operación cotidiana de comunidades como condominios, edificios y pasajes cerrados. Sin embargo, en numerosos casos estos procesos continúan basándose en mecanismos tradicionales, tales como controles remotos, sistemas de citofonía, validaciones presenciales y registros manuales de visitas. Aunque estas soluciones han sido funcionales durante décadas, presentan limitaciones importantes en términos de trazabilidad, eficiencia operativa, escalabilidad y capacidad de adaptación a nuevas exigencias de seguridad y digitalización [8, 9].

En los últimos años, el avance de tecnologías asociadas a la digitalización de infraestructuras, particularmente en el ámbito de IoT y los sistemas ciberfísicos, ha impulsado el desarrollo de soluciones inteligentes orientadas a automatizar y optimizar el control de accesos y la gestión de recursos compartidos. Estas soluciones integran sensores, redes de comunicación, plataformas de procesamiento de datos y aplicaciones digitales para habilitar funciones como la identificación automática de vehículos, la monitorización en tiempo real de la ocupación de estacionamientos y la gestión remota de autorizaciones de ingreso [18, 2, 3].

Dentro de este contexto, la visión por computador ha adquirido un rol protagónico mediante técnicas de detección de vehículos y reconocimiento automático de matrículas, permitiendo reducir la intervención humana en los procesos de validación de accesos. Paralelamente, el uso de tecnologías de comunicación de bajo consumo y largo alcance, así como arquitecturas basadas en servicios, ha favorecido la implementación de sistemas distribuidos capaces de operar con infraestructura limitada y adaptarse a distintos escenarios de despliegue [19, 20, 21].

No obstante, a pesar del progreso alcanzado, la literatura y las soluciones comerciales existentes evidencian diversas restricciones relacionadas con la robustez frente a condiciones ambientales adversas, la dependencia de conectividad, los costos de implementación y la integración con infraestructuras preexistentes, particularmente en comunidades residenciales de pequeña y mediana escala. Estas limitaciones justifican la necesidad de explorar enfoques que combinen automatización inteligente, modularidad y viabilidad de despliegue en contextos reales.

Dado el carácter aplicado e integrador del problema abordado, la revisión considera tanto literatura académica como soluciones comercialmente desplegadas en el contexto chileno. Esta combinación permite analizar no solo avances tecnológicos reportados en investigación, sino también restricciones, enfoques de implementación y dinámicas reales presentes en sistemas actualmente utilizados por comunidades residenciales.

En este capítulo se revisan tecnologías y enfoques relacionados con el control de acceso vehicular, *smart parking*, visión por computador e IoT residencial, con el fin de identificar brechas que justifican la propuesta ResiPark.

2.1. Seguridad vehicular residencial en comunidades inteligentes: contexto, tecnologías y brechas

El control de accesos vehiculares y la administración de estacionamientos en edificios residenciales han pasado de ser una tarea meramente operativa a convertirse en un problema de seguridad, trazabilidad y coordinación entre múltiples actores. En ciudades densas como Santiago de Chile, donde la movilidad cotidiana convive con presión inmobiliaria, alta ocupación de espacios comunes y riesgos de seguridad asociados al acceso vehicular, la gestión manual del ingreso de residentes y visitas evidencia limitaciones importantes: baja trazabilidad, dependencia de criterios *ad hoc* y escasa capacidad para verificar, en tiempo real, si una autorización, una reserva o una ocupación efectiva corresponden entre sí [7, 6, 5].

La literatura sobre comunidades inteligentes y edificios inteligentes plantea que este tipo de problemas no se resuelve únicamente con barreras físicas o dispositivos aislados, sino mediante la integración entre percepción del entorno, comunicación de datos, procesamiento de eventos y servicios digitales capaces de coordinar decisiones y mantener evidencia persistente [1, 3, 2]. Desde esta perspectiva, la seguridad vehicular residencial deja de ser un subsistema periférico y pasa a entenderse como parte de una infraestructura digital más amplia, donde accesos, estacionamientos, reglas comunitarias y aplicaciones de usuario deben operar de manera coherente.

En paralelo, el mercado chileno presenta un escenario fragmentado. Por una parte, existen plataformas de administración comunitaria como ComunidadFeliz, EdiPro y Kastor, que digitalizan pagos, comunicaciones, reservas y parte de la operación cotidiana [22, 23, 24]. Por otra, operan proveedores especializados en control de acceso como Citofon, Federal Access y Access Chile, que ofrecen barreras, lectura de patentes, automatización de portones y control vehicular [9, 13, 25, 26]. Aunque ambos grupos de soluciones cubren partes relevantes del problema, persiste una brecha de integración: el acceso vehicular y la gestión efectiva de estacionamientos de visita tienden a resolverse como dominios separados, lo que dificulta contar con un flujo único que asocie autorización, evidencia visual, ocupación real de plaza y reglas temporales de uso.

En consecuencia, el desafío no consiste solo en automatizar una apertura de portón o en detectar si una plaza está libre, sino en articular ambas capacidades dentro de una arquitectura que permita auditar, verificar y actuar sobre eventos del mundo físico. Bajo esta lectura, es preciso profundizar en el marco general de comunidades inteligentes y en las decisiones específicas de percepción, comunicación e integración que sustentan la propuesta desarrollada en esta investigación.

2.1.1. Comunidades inteligentes, edificios inteligentes y seguridad vehicular residencial

El concepto de comunidad inteligente extiende la noción de ciudad inteligente a escalas más acotadas y operativas, como barrios, condominios o edificios, donde la digitalización se orienta a mejorar la gestión de recursos, la coordinación entre usuarios y la calidad de los servicios cotidianos [3, 2]. En este contexto, la infraestructura física ya no se concibe como un soporte pasivo, sino como un entorno instrumentado capaz de generar datos sobre su

propio funcionamiento. Sensores, cámaras, aplicaciones móviles y plataformas web permiten observar estados del entorno, registrar eventos y reaccionar frente a condiciones previamente definidas.

En los edificios inteligentes, esta lógica se traduce en la integración entre múltiples subsistemas que históricamente habían operado de manera aislada: seguridad, control de accesos, monitoreo ambiental, eficiencia energética y gestión operacional [1]. Esta integración es relevante para el problema de esta memoria porque muestra que la verdadera complejidad no está en cada tecnología por separado, sino en la coordinación entre ellas. Un sistema de acceso vehicular residencial no solo debe identificar vehículos o abrir barreras, sino también relacionar esa decisión con el contexto comunitario: quién autorizó el ingreso, a qué visita o reserva corresponde, cuánto tiempo se utilizará una plaza y qué evidencia queda registrada.

En el caso chileno, esta discusión adquiere especial pertinencia. La disminución reciente en algunos indicadores de robos violentos no elimina la necesidad de reforzar el control situacional en accesos residenciales, particularmente en la Región Metropolitana, donde se concentra una proporción importante de los casos de robo violento de vehículos y donde persisten modalidades como encerronas y portonazos [27, 28]. Por ello, la contribución de un sistema de seguridad vehicular residencial no debe evaluarse únicamente por su capacidad de automatizar el ingreso, sino por su capacidad de reducir ambigüedad operativa, dejar trazabilidad y sostener decisiones consistentes basadas en datos.

2.1.2. Sistemas ciberfísicos y redes de sensores inalámbricas como marco conceptual

Desde un punto de vista conceptual, el problema puede modelarse como un sistema ciberfísico (*Cyber-Physical System*, CPS), es decir, un sistema donde componentes físicos y digitales interactúan continuamente mediante ciclos de percepción, comunicación, procesamiento y actuación. En un acceso vehicular residencial, este ciclo se activa cuando un sensor o una cámara detectan un evento del entorno, dicha información es transmitida a un servicio central, se contrasta con reglas o estados previos, y finalmente se desencadena una acción, como registrar un ingreso, actualizar el estado de una plaza o autorizar la apertura de un portón [1, 18, 2].

Esta lectura es importante porque permite ubicar la contribución de la memoria en el nivel correcto. El aporte del trabajo no reside únicamente en un algoritmo de reconocimiento de patentes ni en un sensor individual de ocupación, sino en el diseño e implementación de una arquitectura donde esos elementos se coordinan para producir decisiones trazables y operativamente útiles. En otras palabras, la solución propuesta se entiende mejor como una cadena de evidencia y actuación que como una suma de dispositivos.

Las redes de sensores inalámbricas (*Wireless Sensor Networks*, WSN) aportan el marco técnico para digitalizar recursos físicos distribuidos, como las plazas de estacionamiento de visita. En este tipo de redes, cada nodo sensor observa una condición local, la comunica al sistema y contribuye a construir una representación global del estado del entorno. En aplicaciones de *smart parking*, la literatura muestra que este enfoque permite transformar cada plaza en una entidad digital con estado verificable, habilitando reservas, monitoreo, alertas y análisis posterior [10, 11, 12]. Esta capacidad resulta especialmente pertinente en edificios residenciales, donde la ocupación de plazas no puede depender exclusivamente de

observación manual o de inferencias indirectas a partir del flujo de vehículos.

2.1.3. Arquitecturas de control de acceso vehicular automatizado

Antes de abordar específicamente el *smart parking*, conviene revisar las principales arquitecturas utilizadas en control automatizado de accesos vehiculares. En términos generales, estas arquitecturas combinan un mecanismo de identificación, una lógica de validación y un actuador físico encargado de ejecutar la apertura o denegación del ingreso. La diferencia entre unas y otras radica en qué tipo de credencial utilizan para identificar al vehículo y en qué nivel de integración alcanzan con los servicios de la comunidad.

Una primera familia corresponde a sistemas basados en barreras automáticas y controladores locales. En este esquema, el acceso se apoya en un actuador motorizado conectado a una controladora que recibe una señal de apertura desde un dispositivo de autorización. Esta arquitectura es ampliamente utilizada en recintos residenciales y comerciales porque permite reutilizar la infraestructura existente del portón o barrera, concentrando el problema en la validación del evento y no en la mecánica del acceso [25, 17].

Una segunda familia agrupa soluciones de identificación por credenciales físicas o de proximidad, como TAG, RFID, controles remotos y credenciales móviles. Estas arquitecturas favorecen rapidez operativa y baja latencia, pero dependen de la posesión previa del identificador y no necesariamente dejan evidencia rica sobre el contexto del acceso. En el mercado chileno, este enfoque sigue vigente tanto en sistemas de acceso vehicular basados en TAG como en esquemas apoyados por aplicaciones móviles [8, 29]. En escenarios de mayor rotación, como estacionamientos comerciales o corporativos, también aparecen esquemas de *ticketing*; sin embargo, su lógica suele estar más orientada al cobro o control de permanencia que a la validación comunitaria de residentes y visitas.

Una tercera familia corresponde a soluciones de reconocimiento automático de matrículas (ANPR/LPR). En estas arquitecturas, una cámara captura la patente del vehículo, un *pipeline* de percepción genera una lectura candidata y una capa de negocio valida si dicha patente está autorizada para ingresar. Su principal ventaja es que eliminan la necesidad de un *token* físico adicional y entregan mayor trazabilidad, ya que la evidencia visual puede asociarse al evento registrado [9, 13, 25, 26]. No obstante, también introducen exigencias de captura, iluminación, conectividad y procesamiento que deben ser consideradas en el diseño del sistema.

Desde la perspectiva de esta memoria, el desafío de integrar cualquiera de estas arquitecturas sobre un edificio residencial existente no reside solo en "abrir el portón", sino en acoplar de forma confiable el actuador físico con una lógica verificable de autorización. En portones eléctricos residenciales, esto implica compatibilizar dispositivos heterogéneos, mantener mecanismos de operación manual o de respaldo y asegurar que la decisión de actuación pueda trazarse hasta un evento concreto del entorno. Esa necesidad de integración justifica que la propuesta desarrollada aquí combine percepción, backend y actuación como partes de una misma cadena operativa.

2.1.4. Tecnologías de comunicación para WSN residenciales: criterios de diseño y comparación entre LoRa, Zigbee y WiFi

Una vez establecido que la ocupación de plazas puede abordarse mediante una red de sensores inalámbricos (WSN), la siguiente decisión crítica corresponde a la selección de la tecnología de comunicación para el tramo entre los nodos de sensado y el sistema central. En este punto, la literatura y la práctica muestran que no existe una tecnología universalmente superior, sino alternativas con ventajas y limitaciones según el patrón de tráfico, el entorno físico, el consumo energético y la infraestructura disponible.

En una red de estacionamientos residenciales, los sensores por plaza transmiten mensajes breves y esporádicos, normalmente asociados a cambios de estado como *libre* u *ocupado*. Por lo tanto, los criterios dominantes no son el ancho de banda ni el *throughput* máximo, sino la cobertura, la penetración en entornos con obstáculos, el bajo consumo energético y la facilidad de despliegue [30, 31, 32, 33]. Bajo estos criterios, las tecnologías más relevantes para este trabajo son LoRa, Zigbee y WiFi.

WiFi ofrece conectividad IP nativa, un ecosistema ampliamente disponible y una integración directa con redes ya presentes en edificios. Por esta razón, resulta especialmente útil para cámaras IP, *gateways* y enlaces de retorno hacia servidores o aplicaciones. Sin embargo, cuando se utiliza en nodos de sensado distribuidos y de tráfico liviano, su consumo energético y su dependencia de cobertura continua en cada punto pueden volverlo menos conveniente para el tramo sensor–gateway [33].

Zigbee, por su parte, ha sido ampliamente adoptado en aplicaciones IoT debido a su bajo consumo y a su capacidad de formar redes malladas. Esta propiedad puede ser ventajosa en instalaciones densas, ya que permite extender la cobertura mediante nodos intermedios. No obstante, su desempeño depende de la estabilidad del mallado y de la correcta ubicación de repetidores, lo que puede complejizar el despliegue en subterráneos, estacionamientos o estructuras con alta atenuación [32, 30].

LoRa y LoRaWAN destacan por su combinación de largo alcance, bajo consumo energético y adecuación a mensajes discretos. Estas características las vuelven especialmente apropiadas para escenarios donde cada nodo solo debe reportar eventos puntuales y donde la cobertura del tramo más adverso resulta prioritaria [31, 30]. Su principal limitación es el bajo caudal disponible, por lo que no resulta adecuada para transportar video o datos de alta frecuencia; sin embargo, esta restricción no afecta de forma significativa al caso de uso de sensores de ocupación.

Desde la lógica de diseño de esta memoria, la comparación conduce a una decisión híbrida: utilizar LoRa para el tramo sensor–gateway, donde predominan los requisitos de cobertura, bajo consumo y transmisión de eventos discretos, y utilizar WiFi o Ethernet como *backhaul* IP entre el *gateway* y los servicios centrales. Esta separación funcional permite aprovechar las fortalezas de cada tecnología sin exigir que una sola red resuelva todos los requerimientos del sistema.

La Tabla 2.1 resume las principales características de estas tecnologías para una WSN residencial. La comparación evidencia que LoRa resulta más adecuada para el sensado distribuido de baja tasa, Zigbee constituye una alternativa válida en escenarios donde el mallado pueda mantenerse estable, y WiFi queda mejor posicionado para componentes con mayores requerimientos de integración IP o ancho de banda, como cámaras o enlaces hacia el backend.

Tabla 2.1: Comparación de tecnologías de comunicación relevantes para una WSN de estacionamientos residenciales.

Tecnología	**Cobertura relativa**	**Consumo energético**	**Topología típica**	**Ventajas y límites para una WSN residencial**
LoRa / LoRaWAN	Alta	Bajo	Estrella / estrella-de-estrellas	Adecuada para mensajes breves, eventos discretos y tramos físicamente adversos. Favorece cobertura y penetración con bajo consumo. Como límite, ofrece bajo caudal y requiere un *gateway* para integrarse al dominio IP [31, 30].
Zigbee	Media	Bajo	Malla	Adecuada para IoT de baja potencia y alta densidad de nodos. Su desempeño depende de un mallado estable y de la ubicación de repetidores, lo que puede complejizar su despliegue en subterráneos o estructuras densas [32, 30].
WiFi	Media	Medio / alto	Infraestructura IP	Presenta excelente integración con redes existentes y alto ancho de banda, por lo que resulta útil para cámaras IP, *gateways* y *backhaul*. Para nodos por plaza puede resultar sobredimensionada por consumo y dependencia de cobertura continua [33].

Para complementar la comparación cualitativa, se construyó una matriz de decisión ponderada orientada específicamente al tramo sensor–gateway del módulo M2. Los criterios considerados responden al contexto de estacionamientos residenciales: cobertura en interiores complejos, consumo energético del nodo, adecuación a eventos discretos, facilidad de despliegue e integración con la red IP existente. Como se observa en la Tabla 2.2, LoRa obtiene el mayor puntaje ponderado, principalmente por su desempeño esperado en cobertura, consumo y transmisión de mensajes de baja tasa.

Tabla 2.2: Matriz de decisión para el tramo sensor–gateway en la propuesta de esta memoria.

Criterio	**Peso**	**LoRa**	**Zigbee**	**WiFi**
Cobertura en interiores complejos	0.30	5	3	2
Consumo energético del nodo	0.25	5	4	2
Adecuación a eventos discretos	0.20	5	4	3
Facilidad de despliegue	0.15	4	3	4
Integración con red IP	0.10	2	3	5
Puntaje ponderado	**1.00**	**4.45**	**3.45**	**2.85**

La Tabla 2.2 no busca establecer una jerarquía universal entre tecnologías, sino justificar la elección en función de los requerimientos concretos del sistema propuesto. En este caso, la decisión favorece LoRa para el tramo de sensado distribuido, mientras que WiFi y Ethernet se mantienen como tecnologías complementarias para el transporte IP hacia los servicios centrales. De esta forma, la arquitectura evita sobredimensionar los nodos de ocupación y, al mismo tiempo, conserva compatibilidad con la infraestructura de red disponible en el edificio.

Como marco general, Biyik et al. revisan arquitecturas por capas para sistemas integrados de *smart parking*, donde se distinguen funciones de sensado, red, middleware y aplicación [34]. Esta lógica de estratificación se presenta en la Figura 2.1, adaptada de [34] que sirve como antecedente conceptual para el modelo por capas adoptado en esta memoria.

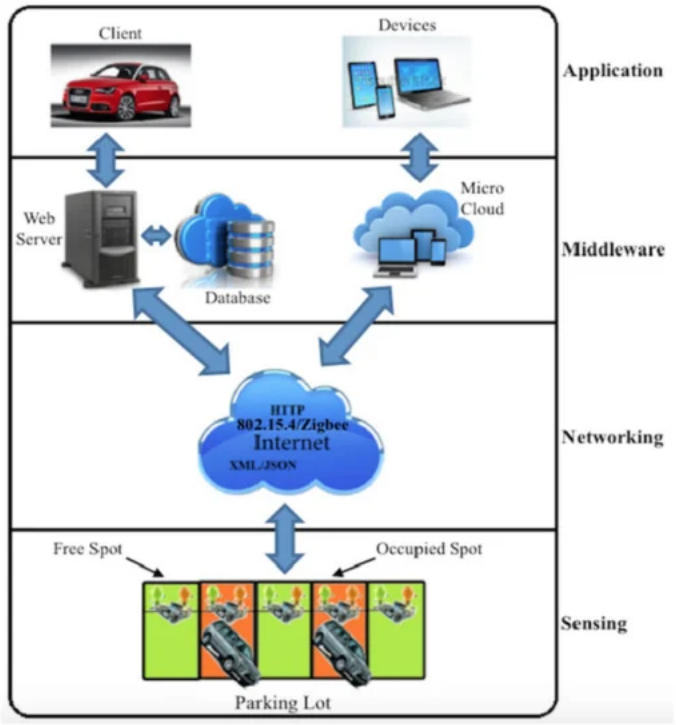


Figura 2.1: Arquitectura por capas para sistemas integrados de *smart parking*.

La Figura 2.1 refuerza que un sistema de *smart parking* no debe entenderse como un dispositivo aislado, sino como una arquitectura compuesta por capas con responsabilidades diferenciadas. En este esquema, la capa de sensado captura el estado físico del entorno, la capa de red transporta los datos, la capa intermedia procesa y organiza la información, y la capa de aplicación la expone a usuarios o servicios. Esta separación resulta coherente con la propuesta ResiPark, donde la ocupación de plazas, la comunicación LoRa, el backend y las interfaces de usuario se integran dentro de una arquitectura modular y trazable.

2.1.5. Detección de ocupación y gestión de plazas: smart parking basado en WSN/IoT

La ocupación de plazas de estacionamiento constituye un problema distinto del control de acceso vehicular, aunque ambos se relacionan operativamente. Mientras el acceso responde a la pregunta de qué vehículo ingresa, la gestión de estacionamientos de visita requiere conocer qué plaza está siendo utilizada, cuándo comenzó su uso, cuánto tiempo lleva ocupada y si dicho uso corresponde a una reserva o autorización válida. Por ello, la literatura de *smart parking* ha desarrollado dos líneas principales: inferencia de ocupación mediante visión por computador y medición directa mediante sensores distribuidos [35, 11, 12].

Las revisiones sistemáticas indican que la visión por computador puede reducir hardware por plaza, pero presenta sensibilidad a oclusiones, cambios de iluminación, ángulos de cámara y demanda de procesamiento continuo [35]. En estacionamientos interiores o subterráneos, estas limitaciones pueden amplificarse debido a columnas, interferencias visuales y variación de condiciones lumínicas. Las soluciones basadas en sensores por plaza, en cambio, transforman cada espacio en una fuente de estado discreto y verificable, lo que mejora la confiabilidad operacional y reduce ambigüedad en la lectura de disponibilidad.

Un referente clásico es SPARK, que plantea una arquitectura de *smart parking* basada en WSN con sensado por plaza, destacando la viabilidad de desplegar el sistema sobre estacionamientos ya construidos y minimizando la necesidad de cableado dedicado [10]. Trabajos más recientes mantienen esta lógica por capas: percepción, red, middleware y aplicación [11, 12]. Esta organización es particularmente valiosa para entornos residenciales, porque desacopla la captura del estado físico de la lógica de reservas, alertas y visualización, permitiendo que cada capa evolucione con menor dependencia de las demás.

La contribución práctica del sistema propuesto no está solo en saber cuántos vehículos ingresaron, sino en poder correlacionar el ingreso con el uso efectivo de un recurso físico compartido. Esta distinción es clave para administrar plazas de visita, aplicar reglas temporales y reducir conflictos operativos en la comunidad.

Srikanth et al. proponen una arquitectura WSN donde cada plaza opera como una fuente autónoma de estado, conectada a una estación de gestión y a clientes de consulta [10]. La Figura 2.2, tomada de [10], ilustra esa organización y funciona como antecedente directo del subsistema de ocupación considerado en esta memoria.

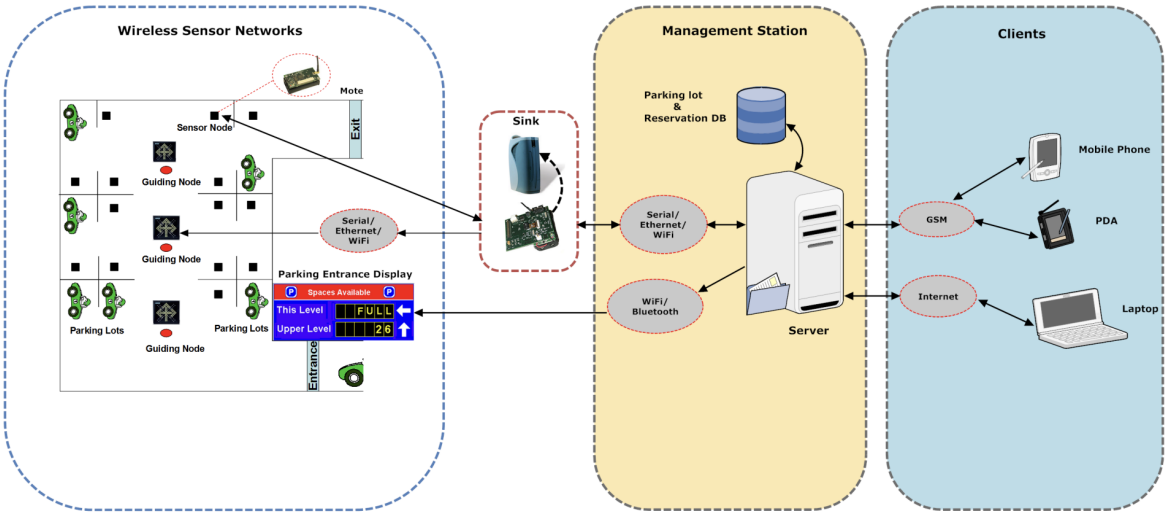


Figura 2.2: Arquitectura general de un sistema de *smart parking* basado en WSN.

La Figura 2.2 permite observar la separación entre la red de sensores, la estación de gestión y los clientes de consulta. Esta organización es relevante para ResiPark porque muestra que la disponibilidad de estacionamientos no debe tratarse como un dato aislado, sino como un evento capturado en terreno, comunicado hacia una capa de gestión y finalmente expuesto a usuarios o servicios.

La Tabla 2.3 resume las principales estrategias utilizadas para detectar ocupación y gestionar disponibilidad en estacionamientos residenciales. La comparación muestra que los enfoques basados exclusivamente en visión por computador presentan ventajas en reducción de hardware, pero pueden verse afectados por condiciones del entorno y requerimientos de procesamiento continuo. En contraste, las soluciones basadas en WSN/IoT permiten representar cada plaza como un recurso digital verificable y trazable, lo que resulta especialmente pertinente para escenarios residenciales donde la confiabilidad operacional y la consistencia entre ocupación física y estado lógico son factores prioritarios.

Tabla 2.3: Comparación de enfoques para detección de ocupación y gestión de disponibilidad en estacionamientos residenciales.

Enfoque	**Señal principal**	**Ventajas y límites en edificios residenciales**	**Referencias**
Visión por computador	Cámara + detección de vehículo por plaza	Reduce hardware por plaza, pero depende de visibilidad continua, buena iluminación y procesamiento sostenido. Puede verse afectada por oclusiones, columnas y variaciones del entorno.	[35]
WSN/IoT con sensores por plaza	Sensor discreto + red inalámbrica	Entrega ocupación verificable, facilita trazabilidad y reduce ambigüedad operacional. Requiere robustecer encapsulado, calibración y mantenimiento del sensor.	[10, 11, 12]
Arquitectura por capas	Eventos de sensores + middleware + aplicación	Permite integrar reservas, alertas, historial y reglas temporales. Exige diseño de consistencia entre estado físico, eventos de red y representación en la aplicación.	[11, 35]

En síntesis, la literatura de *smart parking* respalda el uso de sensores distribuidos y arquitecturas por capas para escenarios donde la ocupación debe ser verificable y auditable. Para la propuesta de esta memoria, esto justifica abordar cada plaza de visita como una fuente de estado propia, conectada a una capa de comunicación y a servicios capaces de relacionar ocupación, reserva y evento de acceso.

2.1.6. Reconocimiento de patentes y percepción en accesos: de ALPR modular a modelos fundacionales

El control de acceso vehicular requiere identificar o verificar vehículos de manera confiable y con latencia acotada. En este ámbito, la literatura ALPR (*Automatic License Plate Recognition*) ha evolucionado desde *pipelines* tradicionales basados en segmentación y reglas heurísticas hacia enfoques de aprendizaje profundo capaces de operar en escenarios más variables. Un punto importante es que, incluso en arquitecturas modernas, persiste una separación funcional entre las etapas de detección del vehículo o la placa, reconocimiento del contenido alfanumérico y aplicación de reglas o validaciones posteriores.

En [20] los autores desarrollan un sistema ALPR basado en aprendizaje profundo que muestra la utilidad de combinar detección y reconocimiento en un *pipeline* robusto ante variaciones de escena. Laroca et al. profundizan esta línea con una propuesta en tiempo real basada en YOLO, mostrando que una arquitectura por etapas bien optimizada puede alcanzar buen desempeño incluso en condiciones más cercanas al mundo real [21]. Este tipo de evidencia resulta especialmente pertinente para accesos residenciales, donde la cámara

debe operar con ángulos, iluminación y velocidades variables, y donde la consecuencia de un falso positivo puede traducirse en una apertura no deseada o en un registro incorrecto.

Desde el punto de vista de diseño, esta literatura respalda una decisión clave de la memoria: separar percepción de lógica de negocio. La cámara y el *pipeline* ALPR deben encargarse de producir evidencia y una lectura candidata de patente, mientras que la validación final, el registro del evento y la decisión de actuación deben permanecer en servicios de backend. Esta separación mejora mantenibilidad, facilita reemplazar modelos o ajustar umbrales, y reduce el acoplamiento entre el componente visual y las reglas comunitarias.

Una línea más reciente es la de modelos fundacionales de visión, como Florence-2, que permiten abordar tareas múltiples mediante *prompting*, incluyendo detección, *grounding* y descripción contextual de escenas [36]. En un sistema residencial, esta capacidad no reemplaza necesariamente al ALPR clásico en tiempo real, pero sí abre una línea complementaria para enriquecer la auditoría posterior, por ejemplo, para describir obstrucciones, colas, múltiples vehículos o condiciones anómalas alrededor del acceso. Así, el valor de estos modelos no está solo en leer una placa, sino en aportar semántica adicional a la evidencia visual.

La Tabla 2.4 sintetiza distintos enfoques utilizados en reconocimiento automático de matrículas y sus implicancias para entornos residenciales. La comparación evidencia que las arquitecturas modulares ofrecen ventajas importantes en mantenibilidad, trazabilidad y separación de responsabilidades entre percepción y lógica de negocio.

Tabla 2.4: Comparación de enfoques ALPR relevantes para control de acceso vehicular en entornos residenciales.

Enfoque	**Componentes típicos**	**Ventajas y límites en acceso residencial**	**Referencias**
Pipeline modular multietapa	Detección vehículo/placa + OCR + postprocesamiento	Alta trazabilidad por etapas; permite calibración fina, umbrales y diagnósticos. Requiere control de calidad de captura e iluminación.	[20, 37, 21]
Detección en tiempo real basada en YOLO	Detector en tiempo real + reconocimiento + reglas	Buena relación velocidad/precisión para escenarios operativos. Puede degradarse en condiciones nocturnas, oclusión o movimiento si la captura no está bien controlada.	[21, 38]
Modelos fundacionales / multimodales	VLM + *prompting* + *grounding* / descripción	Añaden contexto semántico valioso para auditoría y análisis posterior. Su costo computacional y complejidad operativa dificultan su uso como mecanismo primario de apertura en tiempo real.	[36]

Los autores de [21] muestran un *pipeline* modular en tiempo real donde se distinguen las etapas de detección de vehículo, detección de placa, segmentación de caracteres y reconocimiento [21]. Esa separación por etapas, ilustrada en la Figura 2.3, tomada de [21], resulta especialmente pertinente para esta memoria porque refuerza la conveniencia de desacoplar percepción y decisión.

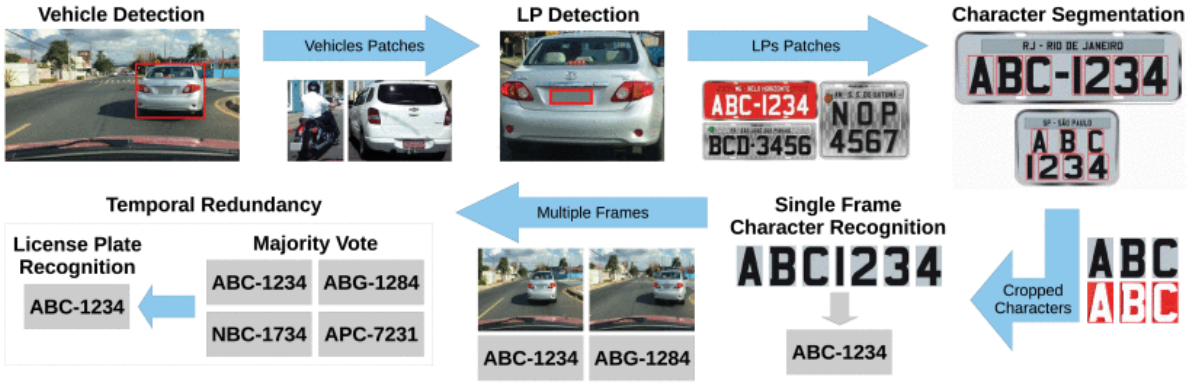


Figura 2.3: Ejemplo de *pipeline* ALPR modular en tiempo real.

La Figura 2.3 permite observar que la lectura de una patente no ocurre como una única operación, sino como una secuencia de etapas con responsabilidades diferenciadas. Esta estructura es relevante para ResiPark porque permite registrar evidencia visual, diagnosticar errores de captura o lectura y mantener separada la inferencia visual de la validación final de autorización.

No obstante, una parte importante de la literatura ALPR reporta resultados principalmente sobre *datasets* controlados o escenarios acotados, concentrándose en métricas de precisión y reconocimiento, pero discutiendo en menor profundidad aspectos asociados a integración operacional, mantenimiento del *pipeline*, manejo de lecturas ambiguas o estrategias de respaldo ante fallos de percepción. Estas dimensiones resultan especialmente relevantes en entornos residenciales reales, donde las decisiones del sistema tienen consecuencias operativas directas sobre el control de acceso.

2.1.7. Interfaces móviles, paneles operativos y servicios centrales

La utilidad de un sistema de seguridad vehicular residencial no depende solo de la calidad del sensado o de la percepción, sino también de cómo se distribuye el control entre residentes, conserjería y administración. En edificios inteligentes, la literatura reporta una tendencia consistente hacia esquemas donde los usuarios finales interactúan mediante aplicaciones móviles, mientras que la operación centralizada y la supervisión recaen sobre paneles administrativos o consolas web [4, 39, 3]. Este patrón es especialmente adecuado en comunidades residenciales, donde distintos actores requieren distintos niveles de visibilidad y control.

En un plano funcional, residentes necesitan reservar plazas, autorizar visitas y recibir notificaciones; conserjería requiere monitorear eventos en curso, validar incidencias y actuar ante situaciones ambiguas; administración necesita revisar historiales, evidencias y métricas

de uso. Esta diferenciación de roles implica que la interfaz no puede diseñarse como un componente accesorio, sino como parte de la arquitectura misma. La manera en que se presentan estados, alertas y evidencias afecta directamente la capacidad del sistema para ser adoptado y operar de forma segura.

Por debajo de estas interfaces, los servicios centrales cumplen el rol de integrar eventos provenientes de cámaras, sensores y reglas comunitarias. La literatura de *smart buildings* y *smart parking* enfatiza la importancia de un *middleware* o capa de servicios capaz de agregar datos, correlacionar estados, exponer APIs y mantener trazabilidad [1, 35, 39]. Para este trabajo, ello implica que el backend no debe ser un simple repositorio de registros, sino el espacio donde se materializa la lógica del sistema: validación de autorizaciones, reconciliación entre acceso y ocupación, generación de alertas y conservación de evidencia.

Esta capa de servicios es también la que habilita la interoperabilidad con otras plataformas. En el contexto residencial chileno, donde muchas comunidades ya utilizan software de administración, el valor de una nueva solución no depende únicamente de sus funciones internas, sino de su capacidad para coexistir con sistemas previos y ampliar sus capacidades sin obligar a reemplazos totales.

La Tabla 2.5 sintetiza cómo distintos actores dentro de una comunidad residencial requieren interfaces y capacidades diferenciadas para interactuar de manera efectiva con el sistema.

Tabla 2.5: Distribución de roles, interfaces y requerimientos operativos en sistemas residenciales de control vehicular y gestión de estacionamientos.

Actor	**Interfaz principal**	**Funciones esperadas**	**Requerimientos clave**
Residente	Aplicación móvil	Reservar plaza, autorizar visitas, recibir notificaciones y consultar estado de accesos.	Baja fricción, confirmación clara, disponibilidad.
Conserjería	Panel operativo web	Monitorear accesos, validar incidencias, revisar evidencia y responder ante eventos ambiguos.	Visibilidad en tiempo real, trazabilidad, rapidez de operación.
Administración	Panel administrativo / *dashboard*	Auditar eventos, revisar uso de plazas, configurar reglas y analizar historial.	Persistencia, reportabilidad, control de políticas.

La Tabla 2.5 evidencia que la arquitectura del sistema no puede diseñarse desde una única perspectiva de usuario, ya que residentes, conserjería y administración poseen necesidades operativas distintas y niveles diferentes de interacción con la plataforma. Esto refuerza la necesidad de una separación clara entre interfaces, lógica de negocio y mecanismos de supervisión dentro de la propuesta desarrollada en esta memoria.

2.1.8. Ecosistema chileno actual y oportunidad de integración sobre plataformas existentes

En Chile, la digitalización de comunidades residenciales ya cuenta con actores consolidados, pero la evidencia del mercado muestra que las capacidades relevantes para este problema están repartidas entre distintos tipos de proveedores. Plataformas como ComunidadFeliz, EdiPro y Kastor se orientan a administración, comunicaciones, reservas y operación general de la comunidad [22, 23, 24]. En cambio, soluciones como Citofon, Federal Access y Access Chile concentran su propuesta en control de acceso, barreras, lectura de patentes y automatización de ingresos [9, 13, 25, 26].

La Tabla 2.6 sintetiza cómo la propuesta de esta memoria se posiciona frente a las principales familias de soluciones actualmente presentes en el contexto chileno. Esto permite observar que las soluciones disponibles tienden a especializarse en dominios específicos, ya sea administración comunitaria o control de acceso vehicular. Sin embargo, persiste una brecha de integración entre autorización de ingreso, evidencia visual, ocupación efectiva de plazas y trazabilidad operacional.

Tabla 2.6: Posicionamiento de la propuesta frente al ecosistema chileno actual.

Familia de solución	**Capacidades visibles**	**Espacio de integración para la propuesta**
Plataformas de administración comunitaria	Reservas, comunicaciones, pagos, conserjería y operación general [22, 23, 24]	Pueden beneficiarse de una capa adicional de percepción vehicular y ocupación verificable por plaza, conectada con reservas y eventos de acceso.
Control de acceso vehicular especializado	Lectura de patentes, barreras, automatización de portones, control vehicular [9, 13, 25, 26]	Pueden complementarse con reglas comunitarias, historial correlacionado y estados de estacionamientos de visita dentro de un flujo unificado.
Solución modular propuesta en esta memoria	Sensado por plaza + ALPR + backend + reglas + evidencia + posibilidad de integración	Busca operar como capa articuladora entre acceso vehicular, uso efectivo de plazas y ecosistema digital de la comunidad.

Esta división es relevante porque permite situar mejor la propuesta de la memoria. El objetivo no es competir frontalmente con plataformas administrativas ni replicar un *stack* comercial cerrado de control de acceso, sino proponer una arquitectura modular capaz de extender, complementar o integrarse con ese ecosistema existente. La oportunidad técnica aparece precisamente en el espacio intermedio al poder conectar el dominio vehicular con el dominio comunitario, relacionando reservas con ocupación efectiva, eventos de acceso con evidencia visual y reglas temporales con estados medidos en terreno.

Desde esta perspectiva, la originalidad del sistema propuesto no radica en afirmar que no existan soluciones previas, sino en mostrar que las existentes suelen cubrir fragmentos del flujo. Algunas se enfocan en la administración general de la comunidad; otras, en el acceso

vehicular; otras, en el monitoreo puntual. La contribución de esta memoria consiste en cerrar ese circuito dentro de una misma arquitectura, manteniendo desacoplamiento entre capas y permitiendo interoperabilidad sobre infraestructura y plataformas ya presentes en el contexto chileno.

En este contexto, la propuesta ResiPark busca actuar como una capa articuladora capaz de conectar dichos componentes dentro de un flujo unificado y auditable, sin exigir el reemplazo completo de las plataformas o dispositivos ya presentes en la comunidad.

2.1.9. Síntesis de brechas y conexión con la propuesta de la memoria

La revisión realizada permite identificar una brecha precisa. La literatura ofrece soluciones maduras para percepción vehicular mediante ALPR, para sensado distribuido mediante WSN/IoT y para integración de servicios en comunidades inteligentes [20, 21, 10, 11, 12, 1, 3]. Sin embargo, el principal desafío no está resuelto por la simple disponibilidad de estas tecnologías, sino por su articulación operativa en un flujo único que vincule ingreso vehicular, ocupación efectiva de plaza, reglas temporales, evidencia persistente y capacidad de actuación.

En términos prácticos, el vacío no es la ausencia de sensores, cámaras o aplicaciones, sino la falta de una arquitectura que los integre sin convertir el sistema en un conjunto frágil de componentes acoplados. Esta observación es especialmente importante para el contexto chileno, donde ya existen plataformas y proveedores, pero donde el control vehicular y la gestión de estacionamientos de visita suelen quedar separados o resueltos de forma parcialmente manual [22, 23, 24, 9, 25, 26]. De este modo, la propuesta desarrollada en esta memoria se justifica como una solución de arquitectura e integración antes que como la invención de un componente aislado.

Bajo esta lectura, la contribución del autor dentro de la memoria se vuelve explícita al definir una arquitectura modular, seleccionar las tecnologías de comunicación adecuadas para el patrón de tráfico del problema, separar percepción de lógica de negocio, proponer un mecanismo verificable de ocupación por plaza y diseñar la integración entre backend, interfaces y entorno físico.

La Tabla 2.7 resume las principales brechas identificadas en el estado del arte y cómo estas se traducen en decisiones concretas de diseño dentro de la propuesta desarrollada en esta memoria. Dicha Tabla 2.7 evidencia que las decisiones de diseño adoptadas en esta memoria no surgen de manera arbitraria, sino como respuesta directa a limitaciones observadas tanto en la literatura como en las soluciones existentes. En particular, se observa que la principal brecha identificada no corresponde a la ausencia de tecnologías individuales, sino a la dificultad de integrarlas dentro de un flujo coherente, verificable y desplegable sobre infraestructura residencial real.

Tabla 2.7: Relación entre las brechas identificadas en el estado del arte y las decisiones de diseño adoptadas en la propuesta.

Brecha o decisión	**Fundamento desde el estado del arte**	**Traducción en la propuesta de la memoria**
Disponibilidad real de plazas	La literatura de *smart parking* recomienda sensado verificable por plaza y arquitectura por capas [10, 11, 35].	Se adopta sensado por plaza para transformar cada estacionamiento de visita en un recurso digital con estado auditable.
Comunicación en entornos complejos	LoRa destaca por cobertura y bajo consumo en mensajes discretos; WiFi resulta útil como *backhaul* IP [30, 31, 33].	Se propone una red híbrida: LoRa en el tramo sensor–gateway y WiFi/Ethernet hacia servicios centrales.
Identificación vehicular confiable	ALPR modular y en tiempo real ofrece trazabilidad y mantenibilidad superiores al acoplamiento total entre percepción y decisión [20, 21].	Se separa la lectura de patentes de la lógica de validación, registro y autorización de acceso.
Interoperabilidad con la operación comunitaria	La literatura y el mercado muestran fragmentación entre percepción, middleware y aplicaciones [1, 35, 22].	La propuesta se concibe como una arquitectura modular capaz de extender o integrarse con plataformas comunitarias existentes.
Contribución de la memoria	La brecha principal es de integración, no de inexistencia tecnológica.	El aporte central es de arquitectura, diseño e implementación de un flujo auditable de acceso y uso de estacionamientos.

Otro aspecto relevante identificado durante la revisión es que muchas propuestas reportan resultados satisfactorios bajo condiciones controladas o centradas exclusivamente en el desempeño de un componente específico, como precisión de detección, cobertura de red o latencia de comunicación. Sin embargo, con menor frecuencia se discuten aspectos asociados a la integración operacional del sistema completo, tales como mantenibilidad del *pipeline*, mecanismos de respaldo ante fallos de conectividad, manejo de lecturas ambiguas o coexistencia con infraestructura residencial ya instalada. Esta diferencia entre desempeño experimental y viabilidad operacional refuerza la necesidad de abordar el problema desde una perspectiva arquitectónica e integradora.

Capítulo 3

Diseño de la solución

En este capítulo se presenta el diseño de la solución propuesta para el sistema inteligente de control de accesos vehiculares y gestión de estacionamientos de visitas en edificios residenciales. El objetivo es alinear la arquitectura técnica con los requisitos levantados en los capítulos previos, manteniendo la modularidad en torno a los cuatro componentes principales definidos para el proyecto: gestión de accesos vehiculares (M1), gestión de estacionamientos de visitas (M2), interfaces de usuario (M3) y servicios centrales (M4).

3.1. Modelo de Capas

El sistema se diseña siguiendo un modelo por capas que organiza las responsabilidades de forma jerárquica. Esto facilita la escalabilidad, la mantenibilidad y la integración futura con otros servicios de la comunidad. A alto nivel, se distinguen cuatro capas:

- **Capa de obtención de datos**: engloba cámaras, sensores IoT y dispositivos de borde responsables de capturar información del entorno físico.
- **Capa de comunicación**: se encarga del transporte confiable de los datos, tanto dentro de la red local del edificio como hacia servicios en la nube.
- **Capa de soporte**: implementa los servicios centrales del sistema, como la API REST, la base de datos y los modelos de IA.
- **Capa de aplicación**: expone la funcionalidad a los usuarios finales mediante aplicaciones web y móviles.

En las secciones siguientes se detalla el rol de cada capa y su relación con los módulos M1–M4.

3.1.1. Obtención de datos

La capa de obtención de datos está compuesta por los dispositivos físicos que interactúan directamente con el entorno:

- **Cámaras IP en el acceso vehicular** (M1), encargadas de capturar el flujo de video de los vehículos que ingresan o salen del recinto. Estas cámaras entregan imágenes en tiempo real que son procesadas por el modelo de visión por computador para la detección de patentes y tipo de vehículo.

- **Sensores de ocupación en estacionamientos de visitas** (M2), implementados sobre microcontroladores ESP32 con módulos de comunicación LoRa. Los sensores detectan la presencia de un vehículo en cada plaza y reportan un estado binario (ocupado / libre) hacia el backend.
- **Dispositivos de borde y gateways**, como Raspberry Pi u otros equipos similares, que permiten agrupar señales de cámaras y sensores, realizar cierto preprocesamiento local y garantizar la comunicación con la nube incluso en entornos de conectividad limitada.

Esta capa abstrae la complejidad del hardware y entrega al resto del sistema información estructurada (patentes detectadas, tipo de vehículo, estado de cada estacionamiento) en formato adecuado para su transporte y procesamiento.

3.1.2. Comunicación

La capa de comunicación conecta los dispositivos de campo con los servicios centrales y las aplicaciones:

- A nivel de sensores de estacionamiento (M2), se utiliza un enlace *LoRa P2P* entre los nodos ESP32 y un nodo receptor, permitiendo cubrir distancias medias con bajo consumo energético.
- La comunicación entre el receptor LoRa, el servidor backend y las bases de datos se realiza sobre redes Wi-Fi o Ethernet, utilizando HTTP/HTTPS como protocolo de transporte para la API REST (M4).
- Las cámaras IP (M1) se integran mediante protocolos estándar (por ejemplo, RTSP) para el *streaming* de video, el cual es consumido por el componente de visión por computador.

Esta capa debe garantizar una latencia razonable para las operaciones en tiempo casi real (accesos vehiculares y actualización del estado de estacionamientos), así como mecanismos básicos de seguridad (segmentación de red, uso de HTTPS y credenciales protegidas).

3.1.3. Soporte

La capa de soporte concentra la lógica de negocio y los servicios centrales del sistema(M4). Sus componentes principales son:

- **API REST backend**, implementada en FastAPI, que expone los endpoints necesarios para registrar accesos vehiculares, gestionar usuarios, administrar reservas de estacionamientos y consultar el estado de los sensores.
- **Base de datos SQL**, desplegada en un servicio gestionado (Supabase), donde se almacenan usuarios, vehículos, reservas, eventos de acceso y estados de estacionamientos. La base de datos aplica mecanismos de seguridad básicos como cifrado de contraseñas y control de accesos por rol.

- **Módulos de IA**, que incluyen el modelo de detección de vehículos y patentes basado en YOLO y el módulo OCR (Florence-2) para la lectura de caracteres. Actualmente se ejecutan en una infraestructura de cómputo remoto (Colab), pero su diseño permite migrar a un servidor dedicado o a un dispositivo de borde con GPU en etapas posteriores.

Esta capa actúa como núcleo integrador: recibe datos desde la capa de obtención, los procesa según la lógica de negocio y entrega respuestas a las aplicaciones cliente.

3.1.4. Aplicación

Finalmente, la capa de aplicación (M3) entrega la funcionalidad al usuario final mediante interfaces específicas para cada perfil:

- **Aplicación web para conserjería y administración**: permite visualizar en tiempo real la cámara de acceso, revisar el estado de los estacionamientos de visitas, gestionar patentes autorizadas, consultar el historial de accesos y administrar usuarios.
- **Aplicación móvil para residentes**: permite a los residentes iniciar sesión, visualizar la disponibilidad de estacionamientos de visita, crear y gestionar reservas para sus visitas y revisar su propio historial de uso.

Ambas aplicaciones consumen la API REST y utilizan los mismos servicios centrales, lo que favorece la consistencia de la información y simplifica el mantenimiento del sistema.

3.2. Requisitos del Sistema

Los requisitos del sistema se derivan directamente de la problemática levantada en el Capítulo 1 y de la revisión del estado del arte en Capítulo 2. Para efectos de esta memoria, se organizan en tres categorías principales: requisitos funcionales, de interfaces y de ambiente, con el fin de separar claramente las capacidades del sistema, los puntos de interacción con los usuarios y las condiciones mínimas del entorno de operación.

3.2.1. Requerimientos Funcionales

Los requerimientos funcionales describen qué debe hacer el sistema para cumplir su propósito de seguridad y operación. En primer lugar, el sistema debe automatizar la identificación de vehículos autorizados mediante reconocimiento de patentes y su validación contra una base de datos que consolida residentes y visitantes. Todo intento de acceso debe quedar registrado de forma trazable, incluyendo fecha, hora, patente y resultado de la validación (autorizado o rechazado). Dicho historial debe estar disponible para el personal de conserjería a través de la aplicación web, permitiendo consulta y revisión de eventos de manera simple y oportuna. Asimismo, ante accesos no reconocidos o no autorizados, el sistema debe generar alertas visibles para la operación, de modo de apoyar la respuesta inmediata y la auditoría posterior.

En segundo lugar, el sistema debe abordar la gestión de estacionamientos de visita como un recurso compartido que requiere visibilidad y control. Para ello, debe detectar y reportar

en tiempo real la disponibilidad de plazas (libres u ocupadas) mediante sensores IoT, y permitir que la administración configure políticas de uso tales como duración máxima, horarios habilitados y tolerancias operativas. Los residentes, a su vez, deben poder visualizar desde la aplicación móvil el estado actual de dichos estacionamientos y disponer de mecanismos para reservar plazas para visitas, especificando fecha y horario, ya sea directamente o mediante roles habilitados como propietarios o administradores. Todo uso de cada plaza debe quedar registrado, asociando vehículo, hora de entrada, hora de salida y duración, para habilitar trazabilidad y análisis de uso. En coherencia con las políticas definidas, el sistema debe generar alertas cuando se detecte un uso que exceda el tiempo permitido, permitiendo la gestión activa del recurso.

Finalmente, el sistema debe incorporar funciones transversales de gestión de usuarios y control de acceso a funcionalidades. Los usuarios deben contar con un mecanismo de recuperación de credenciales para asegurar continuidad de acceso a la plataforma. Además, el sistema debe distinguir roles de residentes, conserjes y administradores controlando permisos y visibilidad de acciones según responsabilidades, con el objetivo de resguardar la operación y evitar accesos indebidos a información o funciones críticas.

3.2.2. Requerimientos de Interfaces

Los requerimientos de interfaces describen cómo el sistema interactúa con los usuarios y con otros componentes técnicos:

- **Interfaces de usuario (M3)**: la aplicación web debe ser responsiva y adaptarse a distintos tamaños de pantalla en conserjería; la aplicación móvil debe ser intuitiva y orientada a tareas frecuentes (ver disponibilidad, reservar, consultar historial).
- **Interfaces técnicas y de comunicación (M4)**: el backend debe ofrecer una API RESTful segura sobre HTTPS, integrarse con cámaras IP mediante RTSP y recibir datos de sensores IoT a través del receptor LoRa conectado a la red del edificio.
- **Interfaces con servicios externos**: la arquitectura considera el uso de servicios en la nube para base de datos y despliegue de modelos de IA, así como plataformas de notificaciones *push* y correo electrónico para alertas a usuarios.

3.2.3. Requerimientos de Ambiente

Finalmente, se especifica el entorno de hardware y software en el que se espera que el sistema opere:

- **Hardware**: Cámaras IP con resolución mínima 1080p, microcontroladores ESP32 con módulos LoRa para cada plaza de estacionamiento de visitas, un nodo receptor con conectividad Wi-Fi o Ethernet, y un servidor (local o en la nube) para el backend y la base de datos.
- **Software**: Sistema operativo tipo Linux para el backend, Python y FastAPI para la API, PostgreSQL como base de datos relacional, y frameworks como React y React Native para las interfaces web y móvil.

- **Conectividad**: Red local estable en el edificio y conexión a Internet para comunicarse con los servicios en la nube y entregar notificaciones hacia los usuarios.

3.3. Arquitectura del Sistema

La arquitectura del sistema concreta el modelo por capas y los requisitos definidos, describiendo cómo se relacionan los módulos M1–M4 y qué información fluye entre ellos.

3.3.1. Diagrama de Contexto

La Figura 3.1 presenta el diagrama de contexto del sistema ResiPark, donde se representa la interacción entre los actores externos (residentes, visitantes, conserjes y administradores) y los dispositivos físicos asociados al sistema (cámaras, sensores y portón). En el diagrama de contexto se representa la interacción entre el sistema ResiPark, los actores externos (residentes, visitantes, conserjes, administradores) y los dispositivos físicos (cámaras, sensores, portón). A nivel lógico, el flujo principal es el siguiente:

1. Las cámaras capturan las patentes en el acceso vehicular.
2. El sistema envía las imágenes al módulo de IA, que identifica la patente y el tipo de vehículo.
3. El backend consulta la base de datos para verificar si el vehículo está autorizado.
4. Si el acceso es válido, se envía la señal de apertura al portón y se registra el evento en la base de datos; en caso contrario, se genera una alerta para conserjería.
5. En paralelo, el sistema monitorea el uso de los estacionamientos de visita de acuerdo con las reservas activas y los datos de los sensores IoT.

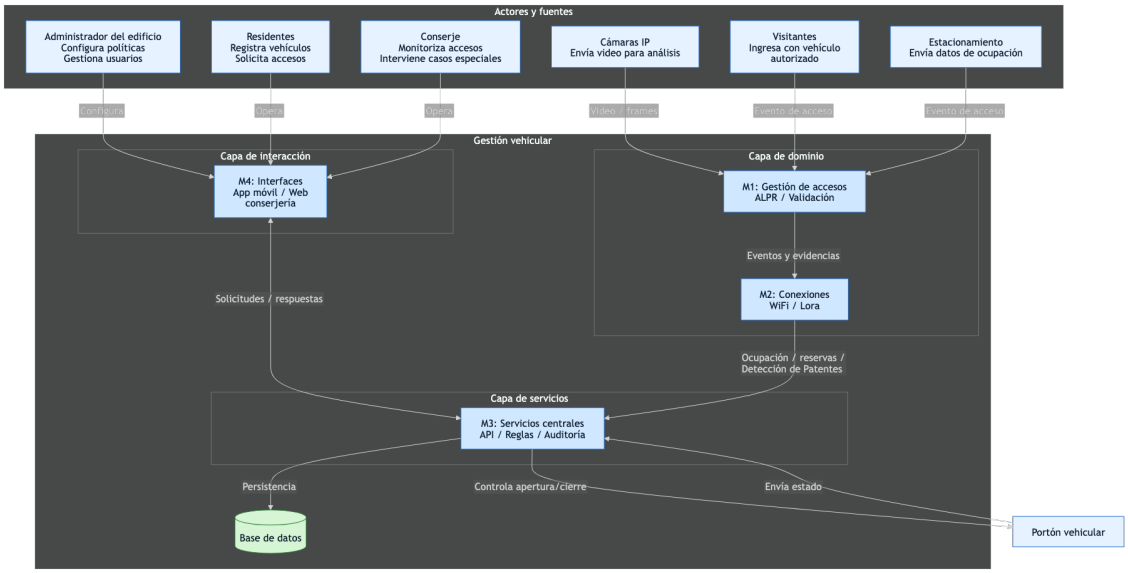


Figura 3.1: Diagrama de contexto del sistema ResiPark.

3.3.2. Diagrama de Arquitectura

La Figura 3.2 resume la arquitectura general de ResiPark, organizada en cuatro capas inspiradas en modelos de sistemas IoT. Esta estructura permite separar las responsabilidades del sistema entre la adquisición de datos, la comunicación, los servicios de soporte y las aplicaciones de usuario.

- **Capa de obtención de datos**: Cámaras IP, sensores de ocupación y modelos de IA para capturan datos del entorno.
- **Capa de comunicación**: Enlaces LoRa, Wi-Fi y Ethernet transportan la información hacia el backend.
- **Capa de soporte**: Servicios de backend (endpoints) y base de datos que realizan el procesamiento y la toma de decisiones.
- **Capa de aplicación**: Aplicaciones web y móviles exponen la información y permiten la interacción de los usuarios con el sistema.

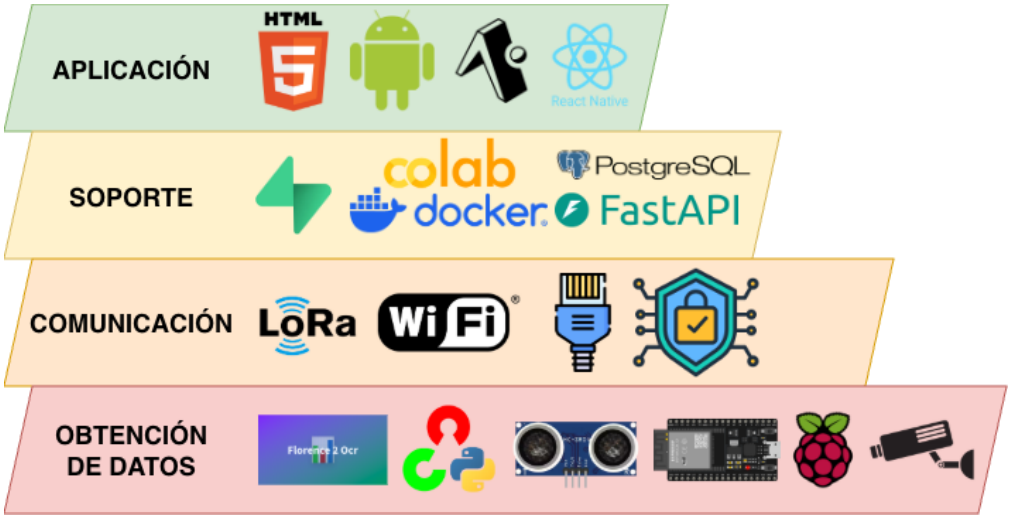


Figura 3.2: Arquitectura general del sistema en cuatro capas.

3.4. Diseño de interfaces

El diseño de interfaces del sistema se plantea como el punto de interacción entre los distintos actores de la solución y los servicios centrales descritos previamente. En particular, se consideran dos interfaces principales: una plataforma web orientada al personal de conserjería y una aplicación móvil destinada a residentes. Esta separación responde a los distintos contextos de uso del sistema. Mientras la interfaz web prioriza supervisión, consulta y control operacional en tiempo real, la interfaz móvil privilegia rapidez, simplicidad y acceso directo a funciones de reserva y consulta.

Desde una perspectiva de diseño, ambas interfaces comparten criterios comunes: reducir la fricción en tareas frecuentes, entregar información relevante de manera inmediata y mantener una navegación coherente con los roles de usuario definidos para el sistema. De este modo, la propuesta de interfaz no solo busca cumplir con los requerimientos funcionales asociados a accesos, reservas e historial, sino también facilitar la adopción del sistema en un entorno residencial real.

3.4.1. Modelo de navegación

El modelo de navegación propuesto distingue dos recorridos principales según el tipo de usuario y su contexto de operación. Por una parte, el flujo de conserjería se implementa sobre una interfaz web centrada en el monitoreo del acceso vehicular y la revisión del estado operativo del sistema. Por otra, el flujo de residentes se implementa mediante una aplicación móvil enfocada en la autogestión de reservas y la consulta del estado de los estacionamientos de visita.

En ambos casos, el punto de entrada corresponde a una pantalla de autenticación. A partir de dicha validación, el usuario accede solamente a las vistas asociadas a su rol. Esta separación permite mantener control de acceso a las funcionalidades críticas del sistema y simplifica la experiencia de uso, evitando exponer opciones que no corresponden al perfil autenticado.

En el caso de la plataforma web, el flujo considera como vista inicial una pantalla de ingreso para conserjería. Tras autenticarse, el usuario accede a un panel principal desde el cual puede observar la cámara de entrada, el estado de los estacionamientos y los eventos relevantes del sistema. Desde este panel, la navegación continúa hacia módulos específicos de historial de accesos y administración de patentes. Este esquema concentra las operaciones críticas en pocas vistas, con el objetivo de reducir tiempos de respuesta frente a situaciones de acceso o incidencias operativas.

En el caso de la aplicación móvil, el flujo inicia igualmente con una pantalla de ingreso. Una vez autenticado, el residente accede a un menú principal desde el cual puede consultar la disponibilidad de estacionamientos y gestionar reservas. La navegación móvil se diseñó bajo una lógica de baja profundidad, privilegiando accesos directos a las funcionalidades más frecuentes y minimizando la cantidad de pasos necesarios para completar una acción.

3.4.2. Prototipo de Interfaces Usuarias

Con el propósito de representar visualmente la propuesta de interacción del sistema, se elaboraron prototipos de baja y media fidelidad para ambas plataformas. Estas vistas permiten visualizar de forma temprana la organización de la información, la jerarquía visual de los componentes y la distribución de las funciones principales antes de consolidar la implementación definitiva.

Diseño de interfaz web para conserjería

La interfaz web fue concebida como una herramienta de operación continua para el personal de conserjería. Su diseño prioriza la visualización simultánea de información crítica, de

modo que el usuario pueda supervisar el acceso vehicular, identificar el estado de ocupación de los estacionamientos y consultar registros sin depender de múltiples cambios de pantalla.

La primera vista considerada corresponde a la pantalla de inicio de sesión para conserjería, mostrada en la Figura 3.3. En esta se propone un acceso simple mediante RUT y contraseña, incorporando además un enlace para recuperación de credenciales. La decisión de mantener una interfaz de autenticación minimalista responde a la necesidad de privilegiar rapidez de acceso y claridad operativa en un entorno donde el sistema debe ser utilizado de manera reiterada durante la jornada.

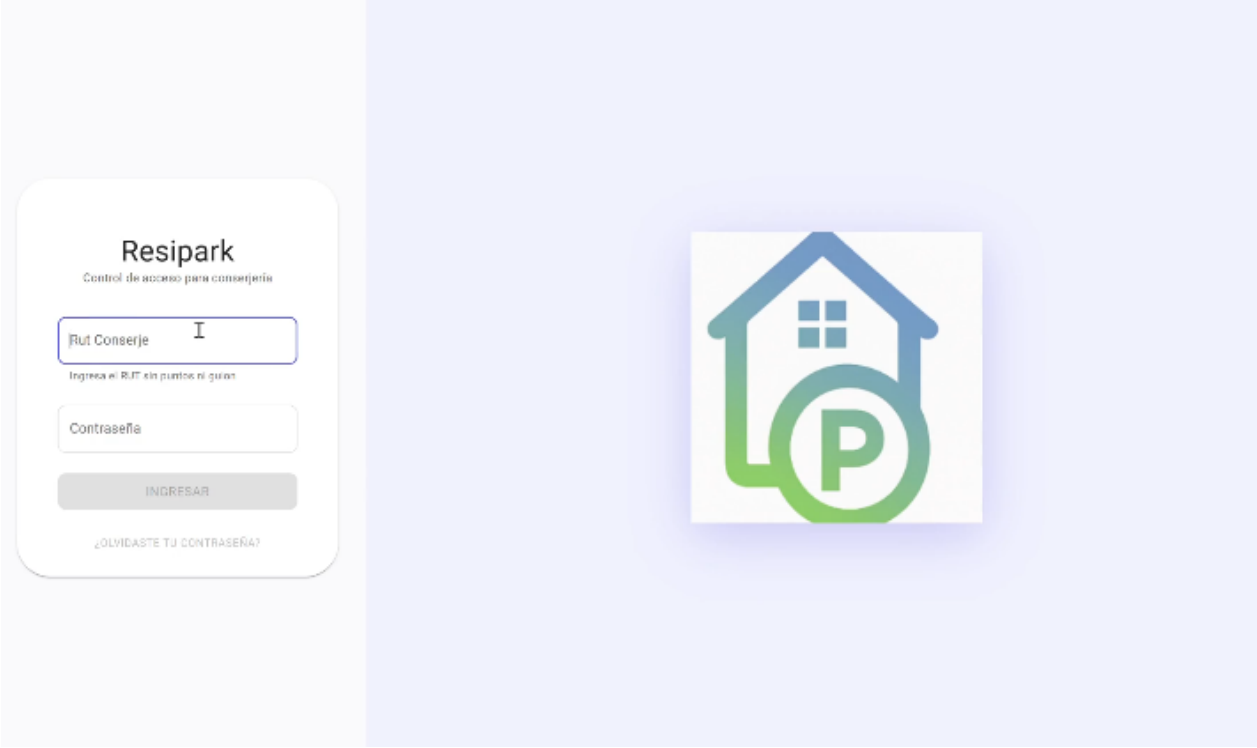


Figura 3.3: Prototipo de inicio de sesión de la plataforma web para conserjería.

Una vez autenticado, el usuario accede al panel principal presentado en la Figura 3.4. Esta vista concentra la operación general del sistema. En ella se propone integrar la visualización de la cámara de acceso, la grilla de disponibilidad de estacionamientos y un módulo lateral de gestión operativa. El objetivo de esta composición es ofrecer una visión centralizada del estado del sistema, permitiendo al conserje tomar decisiones rápidas frente a ingresos, salidas o conflictos asociados al uso de plazas de visita.

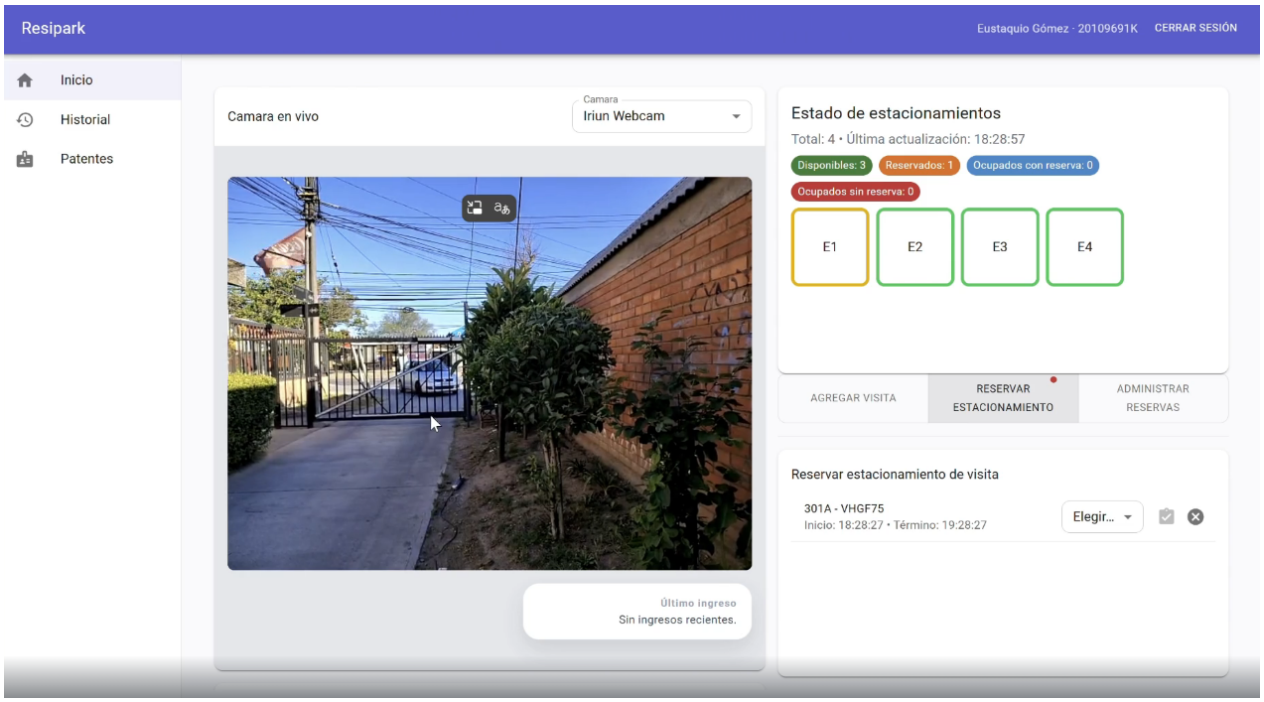


Figura 3.4: Prototipo del panel principal de monitoreo para conserjería.

Complementariamente, la plataforma contempla una vista de historial de ingresos, presentada en la Figura 3.5. Esta pantalla fue diseñada para apoyar la trazabilidad de los eventos registrados por el sistema. La incorporación de una tabla con campos como patente, tipo de usuario, hora de entrada y hora de salida permite reconstruir el comportamiento del flujo vehicular y facilita tanto labores de consulta operativa como revisiones posteriores frente a incidentes.

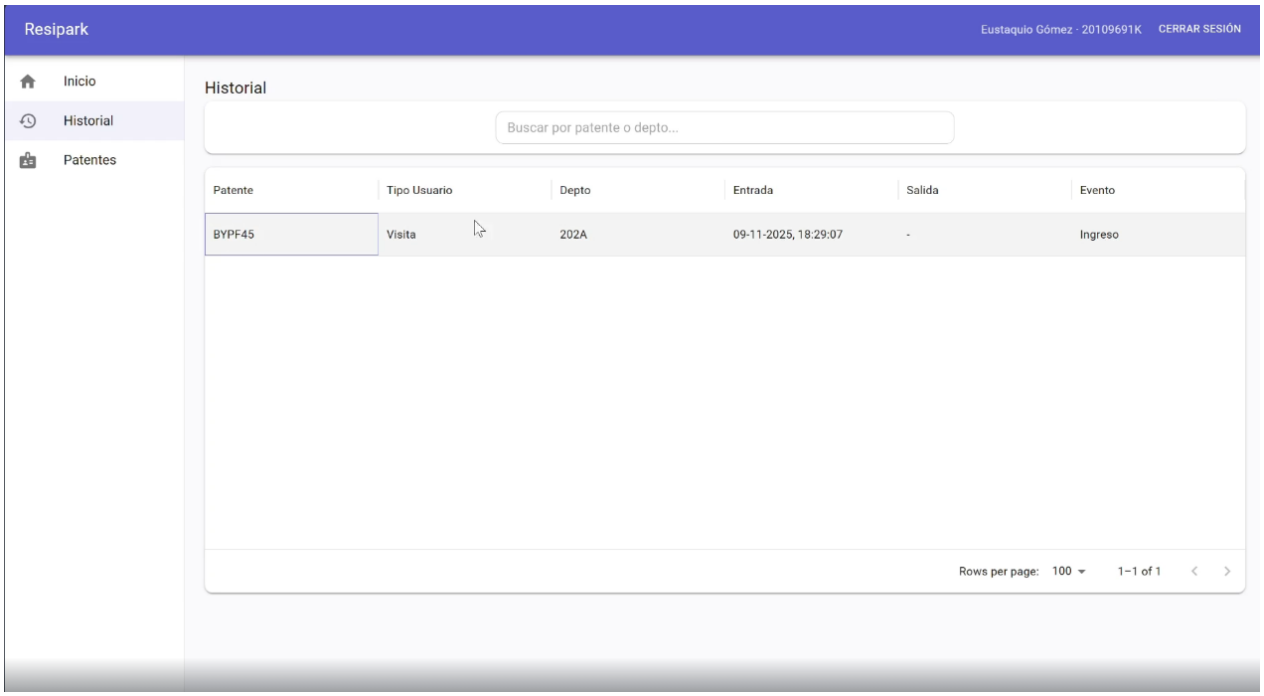


Figura 3.5: Prototipo del módulo de historial de accesos vehiculares.

Finalmente, se propone un módulo de administración de patentes, mostrado en la Figura 3.6, orientado a mantener actualizada la relación entre vehículos autorizados y unidades residenciales. Esta vista refuerza el rol de la plataforma web como punto de control operativo y administrativo, al permitir revisar y modificar información clave para la validación de accesos.

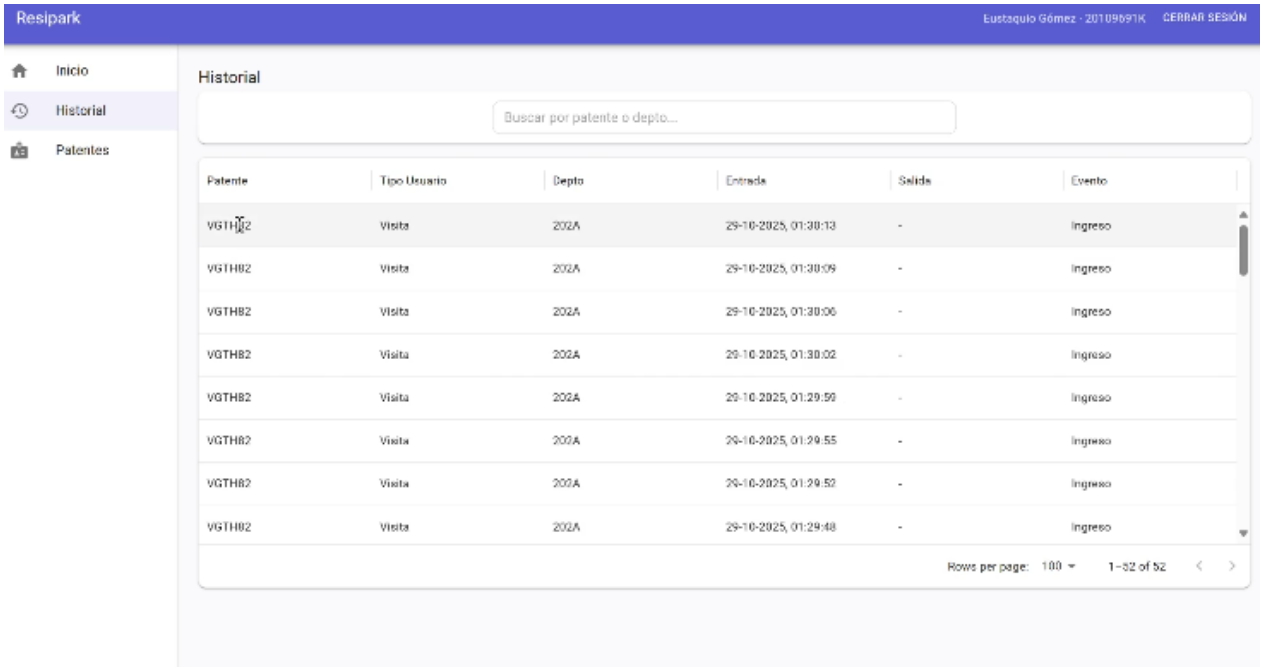
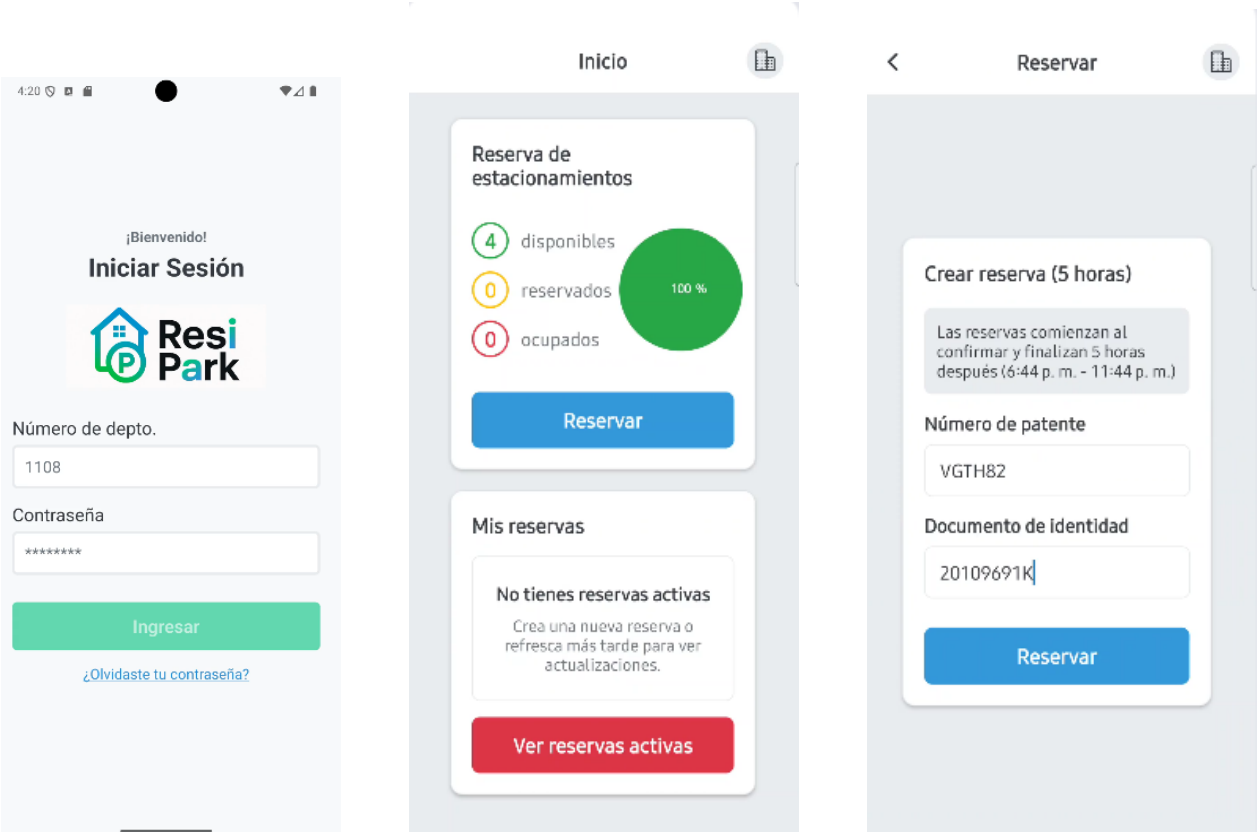


Figura 3.6: Prototipo del módulo de administración de patentes autorizadas.

Diseño de aplicación móvil para residentes

La aplicación móvil fue planteada como la interfaz de autogestión para residentes, privilegiando una experiencia simple y directa. A diferencia de la web de conserjería, cuyo foco es la supervisión continua, la interfaz móvil se diseña para interacciones breves, concretas y centradas en la consulta de disponibilidad y la gestión de reservas.



(a) Inicio de sesión. (b) Menú principal. (c) Reserva de estacionamiento.

Figura 3.7: Prototipos principales de la aplicación móvil para residentes: (a) inicio de sesión, (b) menú principal y (c) reserva de estacionamiento de visita.

La vista inicial de la aplicación corresponde a la pantalla de autenticación presentada en la Figura 3.7a. Esta interfaz considera un esquema tradicional de correo electrónico y contraseña, suficiente para un acceso comprensible y familiar para la mayoría de los usuarios. El diseño busca reducir barreras de uso y facilitar el ingreso a personas con distintos niveles de familiaridad tecnológica.

Tras iniciar sesión, el residente accede al menú principal de la aplicación, ilustrado en la Figura 3.7b. Desde esta vista se estructuran las funciones principales de la aplicación, tales como reservar estacionamiento, consultar disponibilidad y acceder a información asociada a sus gestiones. El objetivo de esta pantalla es actuar como punto de entrada único para las acciones más frecuentes, manteniendo una navegación clara y de baja complejidad.

Por último, la aplicación contempla una interfaz específica para la reserva de estacionamientos de visita, representada en la Figura 3.7c. En esta vista se propone un flujo de ingreso de datos acotado y comprensible, orientado a registrar la información mínima necesaria para concretar una reserva. Esta decisión busca disminuir la fricción del proceso y hacer que la funcionalidad principal del sistema sea ejecutable en pocos pasos desde el dispositivo móvil.

En conjunto, los prototipos propuestos muestran una separación clara entre funciones operacionales y funciones de autogestión. La plataforma web concentra monitoreo, control y trazabilidad para el personal de conserjería, mientras que la aplicación móvil entrega a los residentes una herramienta rápida para consultar información y realizar reservas. Esta distinción fortalece la coherencia del diseño general del sistema y permite alinear cada interfaz con las necesidades reales de sus usuarios.

3.5. Diseño de la infraestructura electrónica para adquisición de datos

En esta sección se presenta el diseño conceptual de la infraestructura electrónica y de comunicaciones que soporta la adquisición de datos del sistema. El foco aquí está en definir el despliegue físico (cámaras, sensores, microcontroladores y enlaces de red) y el flujo de datos extremo a extremo, de manera que la detección basada en modelos de IA y la actualización de estado operen de forma consistente y trazable. La implementación y evidencias de funcionamiento en entorno real se detallan en el Capítulo 4.

El sistema se diseña como un esqueleto ciberfísico compuesto por dos frentes de adquisición: (i) el acceso vehicular, donde la información crítica proviene de una cámara dedicada al reconocimiento automático (patente y atributos del vehículo), y (ii) el área de estacionamientos de visita, donde la ocupación se obtiene mediante sensores IoT por plaza. Ambos frentes se conectan a una capa de comunicación que debe ser robusta en condiciones típicas de edificios residenciales (subterráneos, interferencias y zonas con cobertura irregular), y finalmente convergen en servicios centrales encargados de validar, persistir y distribuir eventos a las aplicaciones.

Zonificación del despliegue físico

Para evitar ambigüedades y facilitar instalación y mantención, el diseño separa el entorno en tres zonas operacionales. La Figura 3.8 muestra la distribución conceptual de estas zonas y la ubicación relativa de los principales componentes físicos del sistema.

- Zona de acceso vehicular: incluye el punto de captura visual para el reconocimiento de patentes y el control del actuador (portón/barrera). En esta zona se prioriza calidad de imagen y control de condiciones (ángulo, distancia, iluminación).
- Zona de estacionamientos de visita: incluye sensores por plaza y nodos de transmisión. Se prioriza robustez del sensado (ocupado/libre) y estabilidad del enlace inalámbrico.
- Zona de conserjería o cuarto técnico: alberga el punto de agregación (gateway o receptor), conectividad a internet (WiFi/Ethernet) y, cuando aplica, el equipo local de

soporte para operación y monitoreo.

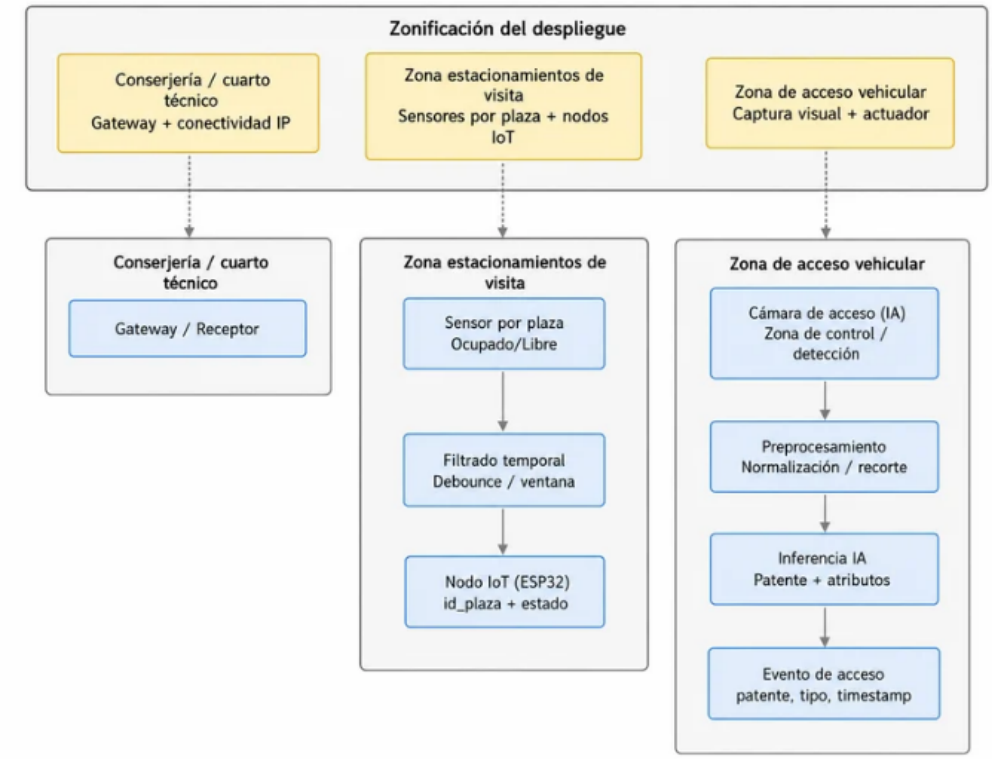


Figura 3.8: Zonificación conceptual del despliegue físico del sistema.

Cámara de acceso para IA: criterios de ubicación y captura

La cámara utilizada para el reconocimiento no se concibe como un elemento de vigilancia pasiva, sino como un sensor de medición. Su ubicación se diseña para maximizar la probabilidad de lectura correcta y minimizar falsas detecciones. Para ello, se define una región de captura en la que el vehículo reduce velocidad o se detiene (zona de control), y se ajusta la cámara para que la patente aparezca con tamaño suficiente dentro del encuadre, evitando perspectivas extremas y reflejos. En términos de diseño, se consideran: inclinación moderada respecto al plano de la patente, distancia acotada al punto de detención, iluminación controlable para condiciones nocturnas, y configuración de captura que reduzca desenfoque por movimiento. Este diseño permite que el pipeline de IA reciba imágenes comparables en el tiempo, simplificando calibración y umbrales de decisión.

Sensores por plaza: criterios de instalación y confiabilidad

El módulo de estacionamientos se diseña para que cada plaza de visita sea un punto de medición independiente. A nivel conceptual, cada plaza incorpora un sensor que entrega un estado discreto (ocupado/libre) y un microcontrolador que empaqueta dicha lectura con un identificador único del dispositivo o plaza. La colocación del sensor se define para reducir falsos cambios de estado y tolerar variaciones del entorno (subterráneos, polvo, humedad

y ruido electromagnético). Para lograr estabilidad, el diseño contempla: montaje protegido (caja o encapsulado), criterio de orientación/altura consistente entre plazas, filtrado temporal (debounce o ventana de confirmación) antes de publicar un cambio de estado, y una estrategia de reintento cuando existan pérdidas de enlace. Con esto, el sistema evita que una lectura puntual errónea se traduzca inmediatamente en un estado incorrecto visible para residentes o conserjería.

3.5.1. Diseño de comunicación: topología, gateway y flujo extremo a extremo

El diseño de comunicaciones se centra en desacoplar la adquisición local del consumo por aplicaciones. En estacionamientos, los nodos IoT transmiten eventos hacia un receptor/gateway ubicado estratégicamente para maximizar cobertura (idealmente en un punto alto o con menor atenuación estructural), mientras que en el acceso vehicular la cámara entrega frames o capturas al componente de procesamiento asociado. Desde el gateway y/o servicios locales, la información viaja a los servicios centrales mediante conectividad IP (WiFi o Ethernet), donde se valida, persiste y se publica como estado consultable.

En este esquema, el enlace inalámbrico de bajo consumo se utiliza para el tramo "difícil" (plazas en subterráneo o zonas sin cableado), y el enlace IP se utiliza como backhaul para llegar a la base de datos y a los servicios. El diseño define además un formato de mensaje mínimo para sensores (id de plaza, timestamp, estado) y un flujo de confirmación lógica a nivel de aplicación para mantener consistencia entre el mundo físico y el estado digital. La consecuencia directa es que las aplicaciones dejan de inferir disponibilidad "por estimación" y pasan a operar sobre datos medidos y trazables.

La Figura 3.9 resume el flujo de comunicación entre los sensores de ocupación, el gateway LoRa y los servicios centrales del sistema.

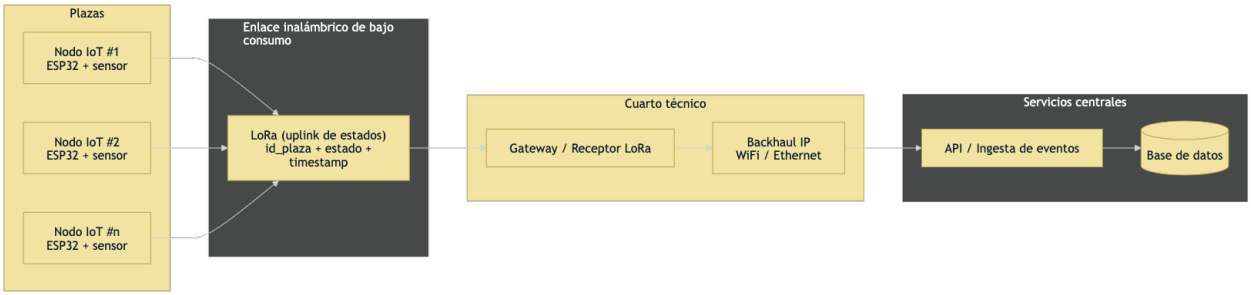


Figura 3.9: Flujo de los sensores en el estacionamiento

3.5.2. Flujo operativo de datos y control

El objetivo de este diseño es que cualquier evento relevante quede representado como un flujo verificable: una medición (sensor o cámara) se transforma en un evento normalizado, se valida contra reglas (autorizaciones, políticas de uso), se registra en persistencia y finalmente se refleja como acción (apertura/alerta) y como estado visible en interfaces. La Figura 3.10 sintetiza este flujo operativo de datos y control extremo a extremo.

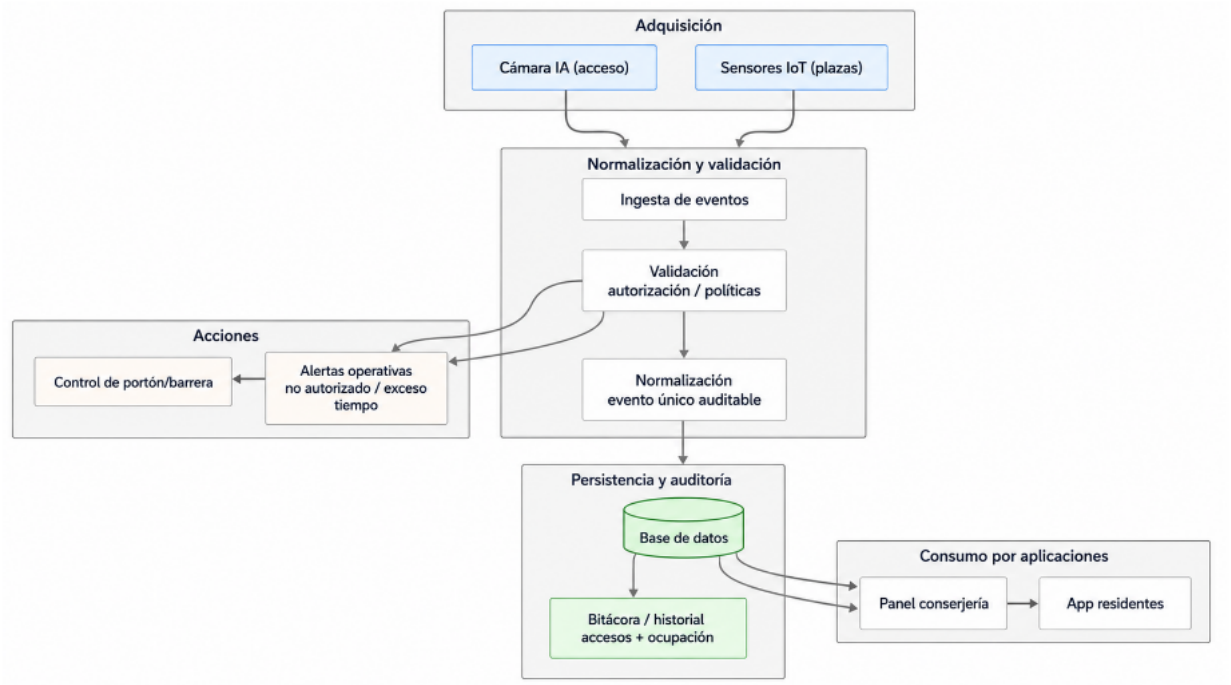


Figura 3.10: Flujo de datos.

3.5.3. Diagrama de la arquitectura y diseño físico

El diagrama mostrado en la Figura 3.11 resume cómo la arquitectura del sistema se materializa en componentes físicos y cómo estos se conectan con la capa de software para cerrar el ciclo completo de detección, validación y acción. La figura separa explícitamente dos planos: (i) la capa superior, "Aplicación y soporte", donde se ubican las interfaces (web y móvil), el backend y la persistencia de datos; y (ii) la capa inferior, "Arquitectura y diseño físico", donde se despliegan los elementos electrónicos en terreno (sensores, microcontroladores, cámara IP, red local y actuador del portón). Esta separación permite entender que la operación digital del sistema depende directamente de decisiones de instalación y conectividad en el entorno real.

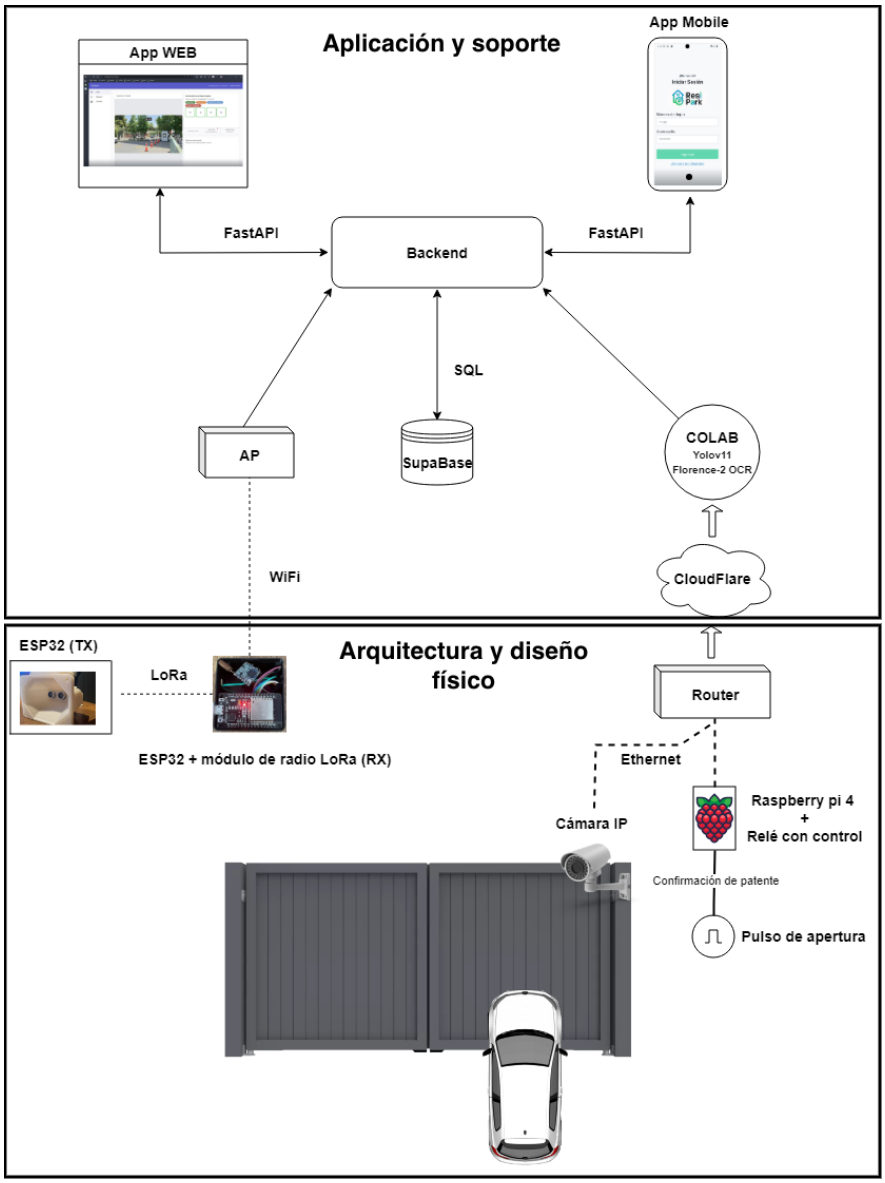


Figura 3.11: Diseño de la Arquitectura.

En el plano físico, los estacionamientos de visita se instrumentan mediante un nodo IoT transmisor (ESP32 TX) que detecta ocupación y envía el estado por LoRa hacia un receptor (ESP32 RX con módulo LoRa). Este receptor actúa como punto de agregación local y, usando conectividad WiFi, entrega los cambios de estado al backend. En paralelo, el acceso vehicular se resuelve con una cámara IP conectada por Ethernet a un router y a un equipo de borde (Raspberry Pi 4), que cumple dos funciones críticas: (1) servir como puente confiable entre el mundo físico (cámara/portón) y los servicios centrales; y (2) ejecutar la lógica de actuación mediante un relé, generando un pulso de apertura únicamente cuando la patente ha sido confirmada como autorizada. Esta decisión de diseño busca que la apertura del portón no dependa de interacción humana directa, sino de un evento verificable y trazable.

En el plano de soporte, el backend expone una API que consumen tanto la aplicación web de conserjería como la aplicación móvil de residentes. El backend persiste información

en una base de datos (Supabase) y se integra con el componente de IA alojado en Colab (modelos YOLOv11 y Florence-2 OCR), accesible mediante un canal seguro publicado hacia internet (Cloudflare). Con esto, el sistema puede recibir datos físicos (sensores/cámara), procesarlos con IA o reglas de negocio, registrar evidencia y finalmente accionar el actuador. En consecuencia, el siguiente capítulo se enfoca en la implementación y despliegue de esta infraestructura física: ubicación estratégica de cámara y sensores, cableado y conectividad, integración con el router, configuración del gateway LoRa y validación del circuito de control del portón, ya que estas decisiones determinan la calidad del dato capturado y la confiabilidad del comportamiento del sistema en operación real.

Este enfoque permite auditar incidentes, depurar fallas de sensores/red, y aislar problemas por zona (captura, transmisión o servicios centrales), lo que prepara el terreno para la etapa de implementación y despliegue descrita en el capítulo siguiente.

En conjunto, este diseño responde a los principios de sistemas ciberfísicos distribuidos, donde la separación entre percepción, comunicación y cómputo permite desacoplar la complejidad del hardware del procesamiento lógico del sistema. Esta descomposición no solo facilita la escalabilidad del sistema en términos de número de sensores y puntos de acceso, sino que también mejora la tolerancia a fallos, ya que cada capa puede ser monitoreada y depurada de forma independiente.

En este sentido, la adopción de tecnologías heterogéneas en la capa de comunicación se justifica por la necesidad de adaptar el medio de transmisión al tipo de tráfico generado, diferenciando explícitamente entre eventos discretos de baja tasa y flujos continuos de mayor demanda.

3.6. Simulación de la tecnología de comunicación para M2

Con el fin de respaldar la selección de la tecnología de comunicación propuesta para el módulo de gestión de estacionamientos de visita (M2), se realizó una campaña de simulación centrada en el tramo sensor–gateway. El objetivo de esta evaluación no fue identificar una tecnología "mejor" en términos generales, sino analizar cuál se ajusta de forma más adecuada al perfil operativo real del subsistema: mensajes discretos, de tamaño reducido, baja frecuencia de transmisión y necesidad de operar en zonas con condiciones de propagación potencialmente adversas, como subterráneos o áreas con atenuación estructural.

En la arquitectura propuesta, cada plaza de estacionamiento se modela como un nodo sensor que determina un estado binario (*ocupado/libre*) y transmite únicamente eventos relevantes hacia un receptor local. En consecuencia, el enlace asociado al módulo M2 no requiere alto caudal, sino confiabilidad suficiente para transportar estados breves y esporádicos con una cobertura robusta. Esta condición lo diferencia del subsistema de video, cuyo perfil de tráfico es continuo y de mayor demanda de capacidad, por lo que ambas decisiones tecnológicas deben justificarse por separado.

Para esta etapa del diseño se utilizó *LoRaEnergySim*, un *framework* de simulación orientado al estudio de nodos, interfaz aérea y *gateway* para enlaces LoRa, originalmente presentado por Callebaut, Ottoy y Van der Perre [40]. Este entorno fue complementado con procesamiento y visualización de resultados en Python/*matplotlib*, con el propósito de estu-

diar el comportamiento esperado de un enlace LoRa punto a punto en una topología tipo estrella entre nodos de estacionamiento y un *gateway* local. La simulación se planteó como una aproximación funcional del comportamiento del enlace inalámbrico y no como una emulación bit a bit del *firmware* final del prototipo. En particular, se tomó como referencia el uso de módulos SX1276/78 operando en modo LoRa P2P, correspondientes a la solución definida para el subsistema M2.

3.6.1. Diseño experimental

La evaluación se estructuró en tres campañas complementarias. La primera correspondió a un barrido de distancia entre un nodo sensor y el *gateway*, con el fin de analizar cómo evoluciona la tasa de entrega de paquetes al aumentar la separación física entre ambos. La segunda evaluó el efecto de incrementar el número de nodos activos en una topología tipo estacionamientos, con el fin de observar si el patrón de tráfico esperado introduce degradación significativa en la confiabilidad del enlace. La tercera se orientó al comportamiento temporal del sistema bajo incremento de carga, evaluando la latencia del enlace y la degradación de la tasa de entrega al disminuir el intervalo entre transmisiones.

Para los experimentos se consideró un mensaje corto de 12 bytes, consistente con el tipo de dato que transmite el sistema: identificador de plaza, marca temporal y estado discreto. Asimismo, se modeló un patrón de tráfico esporádico, coherente con un sistema que reporta cambios de estado y no un flujo continuo de datos. En la campaña de distancia se evaluaron tres configuraciones de *spreading factor* (SF7, SF9 y SF12), con el objeto de observar el compromiso entre robustez del enlace y costo energético. En la campaña de sobrecarga se utilizaron principalmente SF7 y SF9, aumentando tanto la cantidad de nodos como la frecuencia de transmisión.

Las métricas utilizadas para interpretar los resultados fueron la tasa de entrega de paquetes (DER, *Data Extraction Rate*), la latencia media de los paquetes recibidos exitosamente, el consumo energético promedio por byte transmitido y los mecanismos de pérdida asociados a colisiones o paquetes débiles.

3.6.2. Robustez del enlace en función de la distancia

La Figura 3.12 presenta la DER del enlace LoRa para distintas distancias entre el nodo sensor y el *gateway*, comparando tres valores de *spreading factor*: SF7, SF9 y SF12. Para SF7, la tasa de entrega de paquetes se mantuvo en 1.0 entre 10 y 50 metros, descendiendo a 0.9565 en 60 y 80 metros, y a 0.8696 en 100 metros. En el caso de SF9, la DER se mantuvo en 1.0 en prácticamente todo el rango evaluado, con una caída puntual a 0.9565 en 80 metros. Por su parte, SF12 mantuvo una DER igual a 1.0 en todo el barrido considerado entre 10 y 100 metros.

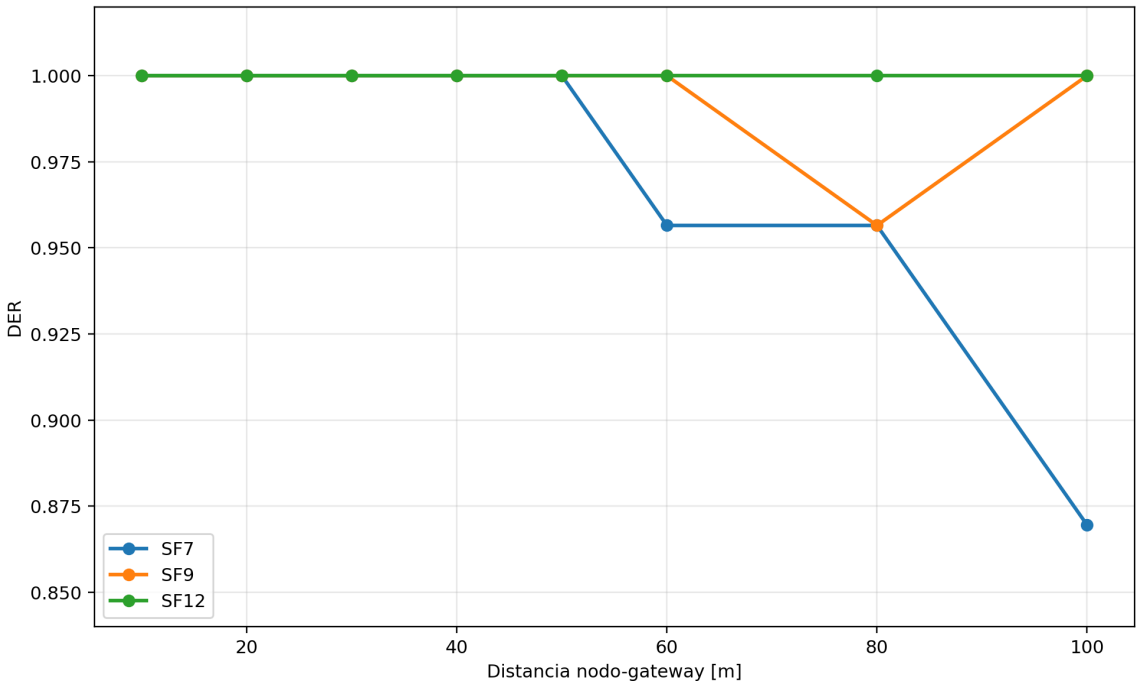


Figura 3.12: DER del enlace LoRa en función de la distancia entre nodo y *gateway* para distintas configuraciones de *spreading factor*.

Los resultados de la Figura 3.12 muestran que el aumento del *spreading factor* mejora la robustez del enlace frente al incremento de distancia. Mientras SF7 presenta degradación observable en el extremo superior del rango evaluado, SF9 y SF12 mantienen una entrega más estable. Esto evidencia el comportamiento esperado de LoRa: mayores valores de SF permiten sostener la comunicación en condiciones más exigentes de propagación, aunque dicha mejora debe analizarse junto con su costo energético.

3.6.3. Compromiso entre robustez y costo energético

La mejora de robustez observada con mayores valores de SF implica un incremento en el costo energético por byte transmitido. La Figura 3.13 muestra que SF7 presentó un consumo promedio cercano a 2.50 mJ/byte, SF9 alcanzó aproximadamente 4.64 mJ/byte y SF12 llegó a valores del orden de 21.55 mJ/byte. Esta diferencia permite observar el compromiso clásico de LoRa: un *spreading factor* mayor aumenta el tiempo en aire y la energía consumida, pero mejora la probabilidad de recepción exitosa en condiciones de propagación desfavorables.

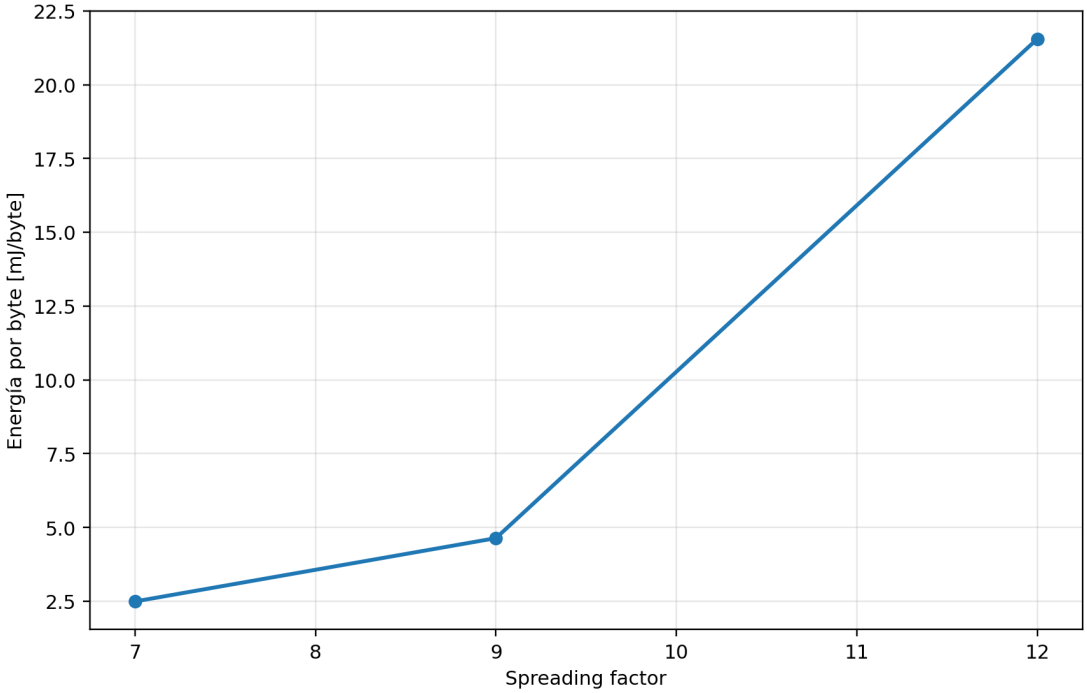


Figura 3.13: Energía promedio consumida por byte transmitido según *spreading factor*.

La Figura 3.13 permite complementar la lectura de la campaña de distancia. Aunque SF12 ofrece la mayor robustez en el rango evaluado, también presenta un costo energético significativamente mayor. En cambio, SF7 y SF9 muestran consumos más moderados, por lo que el análisis de la tecnología no puede limitarse a maximizar la DER, sino que debe equilibrar robustez, tiempo en aire y consumo energético.

3.6.4. Escalamiento con la cantidad de nodos

La segunda campaña evaluó la DER al aumentar el número de nodos activos en un escenario tipo estacionamientos, utilizando SF7 y una topología de 10, 20 y 30 sensores. Como se observa en la Figura 3.14, los resultados fueron 1.0000, 0.9978 y 0.9809, respectivamente. Esto indica que, bajo un patrón de tráfico de baja frecuencia y mensajes pequeños, el enlace mantiene un nivel alto de confiabilidad incluso al aumentar la cantidad de sensores desplegados.

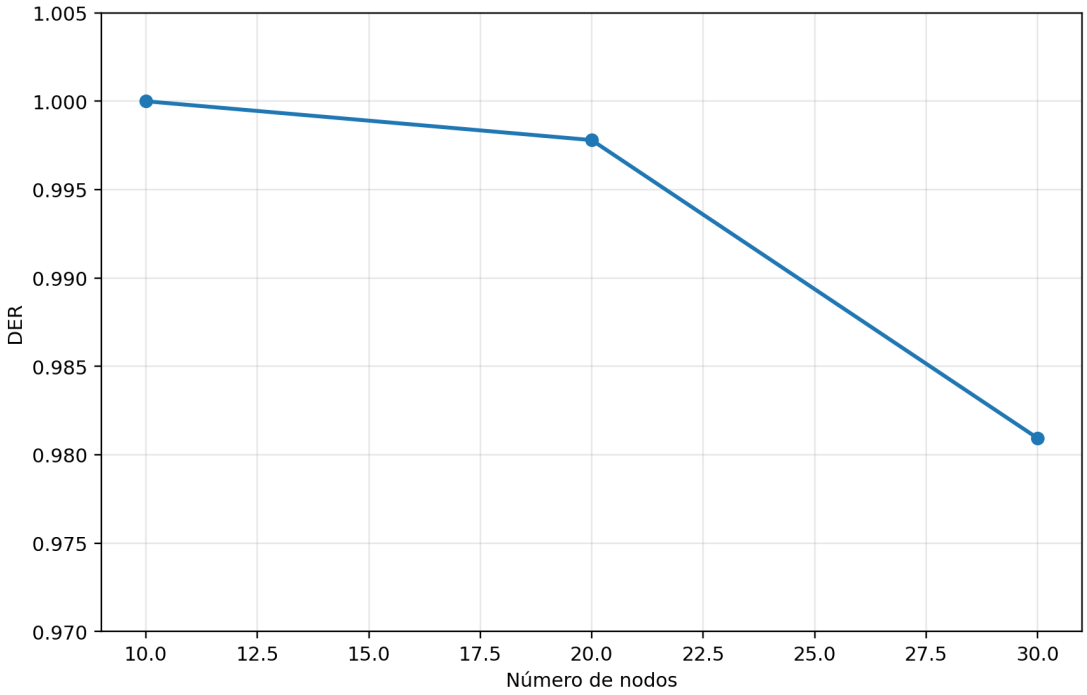


Figura 3.14: DER del enlace LoRa al incrementar la cantidad de nodos sensores en la topología de estacionamientos.

La lectura principal de la Figura 3.14 es que el incremento de nodos genera una degradación gradual, pero no abrupta, de la tasa de entrega. Esto es coherente con un escenario de sensado discreto, donde los nodos no transmiten flujos continuos, sino mensajes breves asociados a estados de ocupación. En consecuencia, el comportamiento observado sugiere que el sistema puede tolerar un número moderado de plazas instrumentadas sin entrar inmediatamente en un régimen severo de pérdida de paquetes.

3.6.5. Latencia del enlace y comportamiento bajo sobrecarga

Con el fin de evaluar el comportamiento temporal del enlace, se realizó una campaña adicional orientada a latencia bajo distintas cargas, variando el número de nodos, el *spreading factor* y el intervalo entre transmisiones. La Figura 3.15 muestra que la latencia de los paquetes entregados exitosamente permaneció prácticamente constante dentro de cada configuración de SF, situándose en torno a 97.6 ms para SF7 y 233.5 ms para SF9.

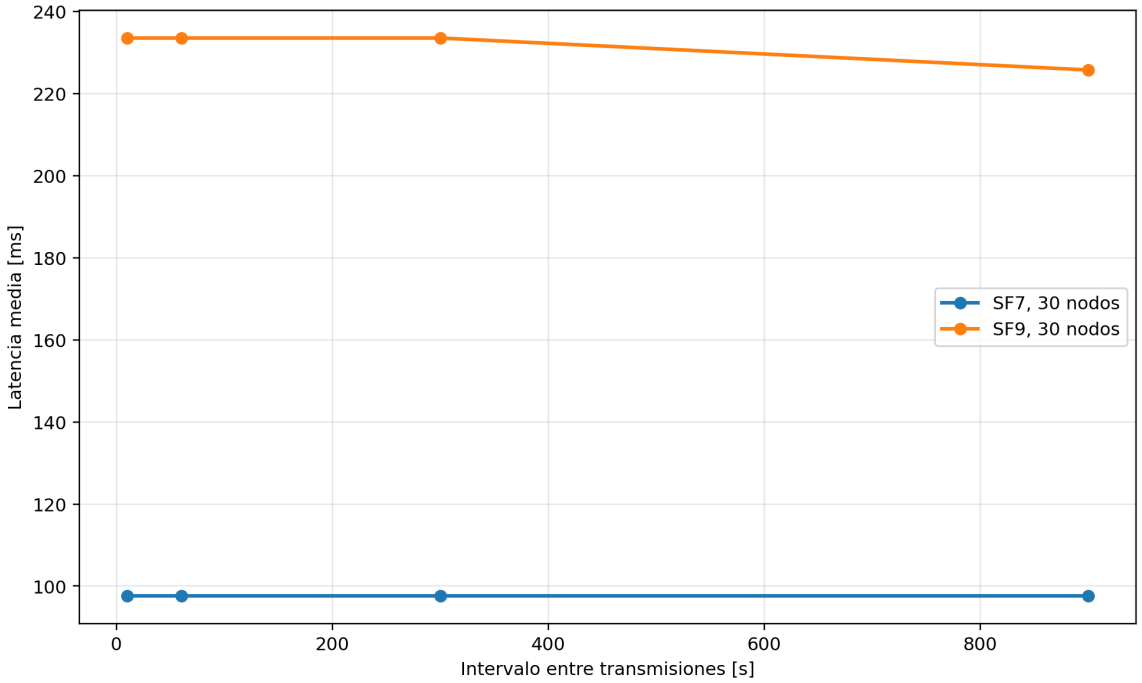


Figura 3.15: Latencia media del enlace LoRa bajo distinta carga, expresada mediante el intervalo entre transmisiones.

La Figura 3.15 indica que, en la configuración simulada, la latencia observada se encuentra dominada principalmente por el tiempo propio de transmisión y recepción de cada paquete exitoso, más que por una acumulación significativa de espera en cola. Dicho de otro modo, al aumentar la carga del sistema, el efecto adverso principal no aparece como un crecimiento pronunciado de la latencia media, sino que debe analizarse a través de la tasa de entrega de paquetes.

La Figura 3.16 complementa este resultado mostrando la DER bajo distintos intervalos entre transmisiones. Al reducir el intervalo entre transmisiones y aumentar simultáneamente el número de nodos activos, la DER comenzó a decrecer de forma observable. Esta degradación fue más marcada en las configuraciones con SF9, lo que resulta coherente con su mayor tiempo en aire y, por tanto, con una ocupación más prolongada del canal.

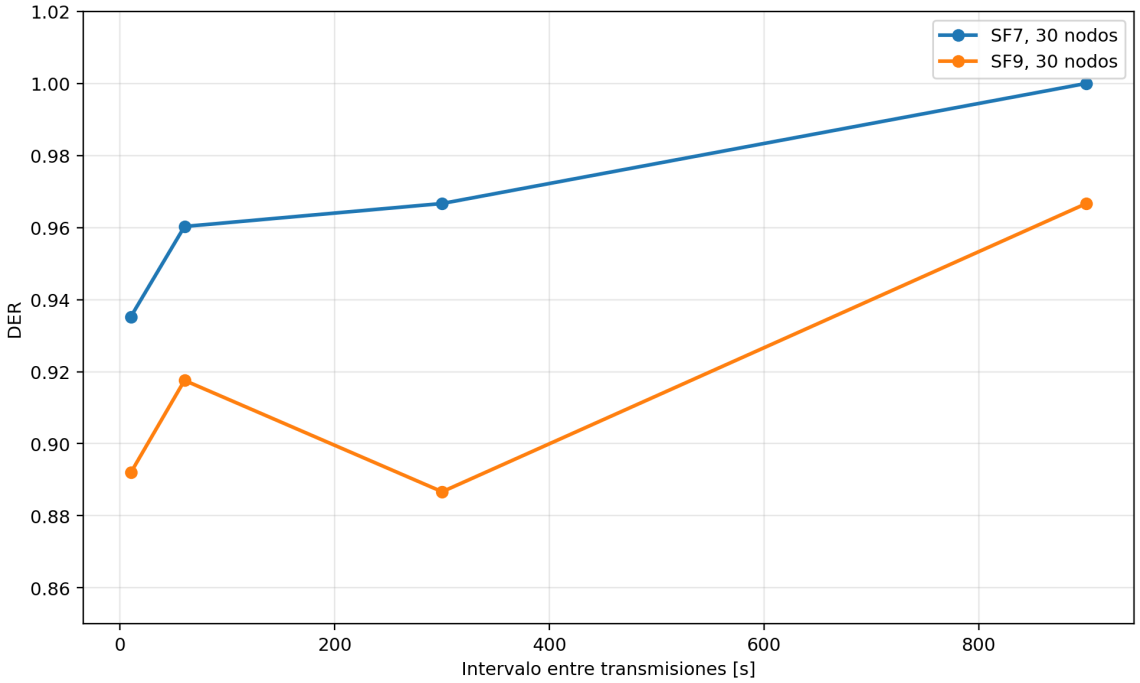


Figura 3.16: DER del enlace LoRa bajo distinta carga, expresada mediante el intervalo entre transmisiones.

La lectura conjunta de las Figuras 3.15 y 3.16 muestra que la sobrecarga no se manifiesta, en primera instancia, como una latencia media excesiva en los paquetes exitosamente recibidos, sino como una disminución gradual de la confiabilidad del enlace. Esto es relevante porque permite distinguir entre dos efectos distintos: los paquetes que llegan mantienen una latencia acotada, pero el porcentaje de paquetes entregados disminuye cuando el canal se presiona con mayor frecuencia de transmisión.

3.6.6. Pérdidas observadas bajo incremento de carga

La Figura 3.17 permite observar con mayor detalle los mecanismos asociados a la degradación de la DER bajo incremento de carga. En particular, se muestran colisiones y paquetes débiles para configuraciones representativas del enlace. A medida que disminuyó el intervalo entre transmisiones y aumentó la cantidad de nodos activos, ambos mecanismos de pérdida se volvieron más visibles, especialmente en configuraciones de mayor tiempo en aire.

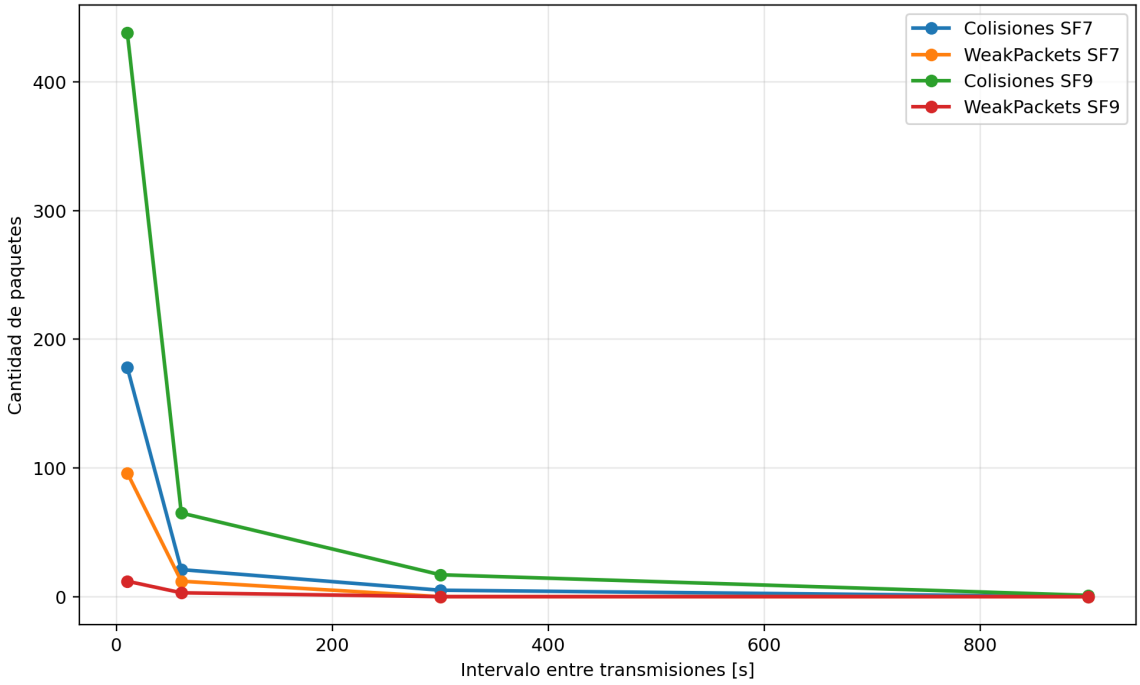


Figura 3.17: Colisiones y paquetes débiles observados bajo distinta carga para configuraciones representativas del enlace LoRa.

La Figura 3.17 confirma que la pérdida de confiabilidad bajo sobrecarga no depende de un único factor, sino de la combinación entre mayor ocupación del canal, colisiones y recepción de paquetes con condiciones débiles. Este comportamiento es consistente con los resultados de la Figura 3.16, donde la DER disminuye al reducirse el intervalo entre transmisiones. Por tanto, la campaña de sobrecarga permite identificar el límite práctico del enlace: el sistema se comporta adecuadamente bajo tráfico discreto, pero se degrada cuando se fuerza hacia un régimen de transmisión más intensivo.

3.6.7. Síntesis de resultados de simulación

Los resultados obtenidos permiten resumir tres observaciones principales. En primer lugar, el aumento del *spreading factor* mejora la robustez del enlace frente a distancia y condiciones de propagación más exigentes, como se observó en la Figura 3.12. En segundo lugar, dicha mejora tiene un costo energético claro, evidenciado por la Figura 3.13, donde SF12 presenta un consumo considerablemente mayor que SF7 y SF9. En tercer lugar, las campañas de carga muestran que el incremento de nodos o de frecuencia de transmisión afecta principalmente la DER y los mecanismos de pérdida, más que la latencia media de los paquetes recibidos exitosamente.

En consecuencia, desde una perspectiva estrictamente experimental, los resultados muestran que LoRa presenta un comportamiento favorable para mensajes breves y esporádicos, pero también evidencian restricciones importantes cuando se incrementa artificialmente la carga del canal. Esta observación resulta clave para separar la lectura de resultados de la decisión tecnológica: las simulaciones no indican que LoRa sea adecuada para todo tipo de

tráfico, sino que muestran su comportamiento bajo un perfil específico de sensado discreto.

3.6.8. Estimación de autonomía energética del nodo LoRa

Además de la confiabilidad del enlace, la selección de LoRa para el módulo M2 debe analizarse desde el punto de vista energético. En el subsistema de estacionamientos, cada nodo sensor no necesita permanecer transmitiendo de forma continua, sino que puede operar bajo un esquema de bajo consumo: permanecer en modo *sleep* durante la mayor parte del tiempo, despertar para realizar una medición, transmitir el estado de la plaza y volver nuevamente a reposo. Esta lógica es coherente con el patrón de tráfico del sistema, ya que la ocupación de un estacionamiento cambia de manera esporádica y no requiere un flujo permanente de datos.

Para estimar la autonomía se consideró una batería de 2500 mAh y un nodo compuesto por un microcontrolador, un sensor de ocupación y un módulo LoRa basado en SX1276/78. El cálculo no se realiza únicamente con el consumo del transceptor LoRa, ya que en una implementación real también influyen el microcontrolador, el sensor, el regulador de voltaje y las pérdidas asociadas al circuito. Por esta razón, se utiliza un modelo simplificado de nodo completo.

La autonomía se estimó mediante el consumo promedio del ciclo de trabajo, calculado como:

$$I_{prom} = \frac{I_{sense} \cdot t_{sense} + I_{tx} \cdot t_{tx} + I_{sleep} \cdot (T - t_{sense} - t_{tx})}{T} \tag{3.1}$$

donde I_{prom} corresponde a la corriente promedio del nodo, I_{sense} a la corriente durante la etapa de sensado, I_{tx} a la corriente durante transmisión, I_{sleep} a la corriente en reposo, t_{sense} al tiempo durante el cual el nodo permanece activo midiendo, t_{tx} al tiempo de transmisión y T al intervalo total entre ciclos de sensado. La autonomía se obtiene luego como:

$$Autonomia = \frac{C_{util}}{I_{prom}} \tag{3.2}$$

Para la estimación se asumió una capacidad utilizable del 80 % de la batería nominal, es decir, $C_{util} = 2000$ mAh. Esta aproximación permite considerar pérdidas del regulador, caída de tensión útil y degradación práctica de la batería. Se utilizaron como valores de referencia $I_{sleep} = 0{,}15$ mA, $I_{sense} = 60$ mA, $I_{tx} = 140$ mA y $t_{tx} = 0{,}25$ s. Estos valores representan una estimación conservadora del nodo completo, no solamente del chip LoRa. En particular, aunque el SX1276/78 puede alcanzar consumos de reposo del orden de microamperios, el consumo real del nodo queda determinado por la suma del transceptor, el microcontrolador, el sensor y la electrónica auxiliar.

La Figura 3.18 presenta la autonomía estimada para distintos intervalos entre sensados/transmisiones, considerando ventanas activas de 1.25s, 2.25s y 5.25s. La ventana activa incluye el tiempo de sensado y una transmisión breve del estado de ocupación.

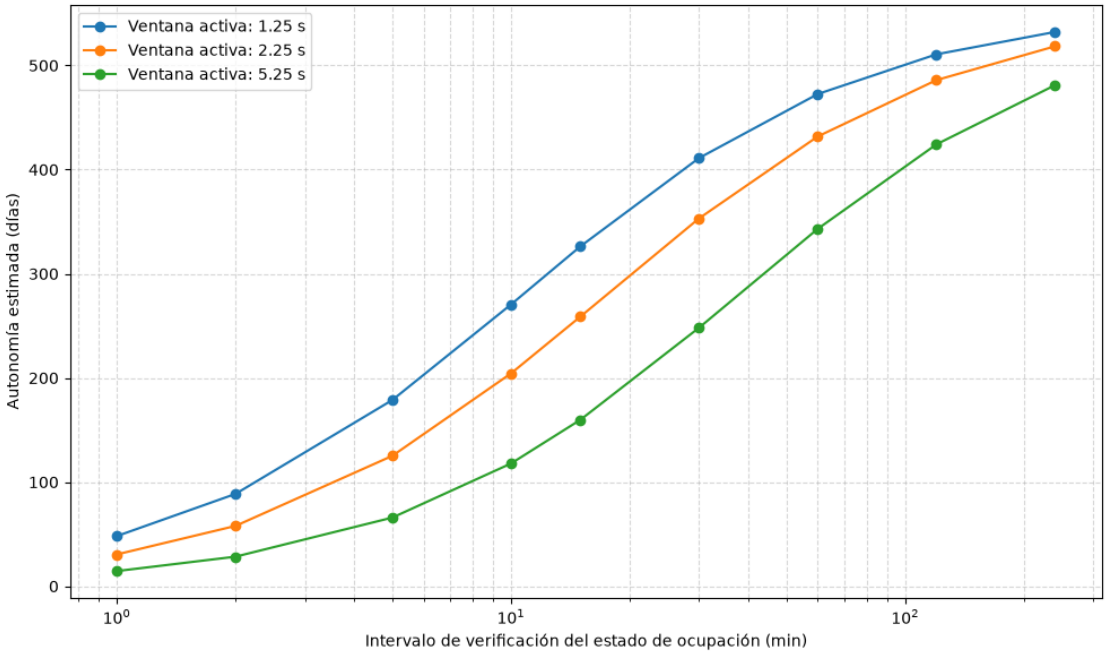


Figura 3.18: Autonomía estimada de un nodo LoRa alimentado con batería de 2500 mAh según la frecuencia con que despierta para sensar y transmitir el estado de ocupación.

Los resultados muestran que la autonomía depende fuertemente del intervalo entre ciclos de sensado. Para una ventana activa aproximada de 2.25 s, un intervalo de sensado de 1 minuto entrega una autonomía cercana a un mes, mientras que intervalos de 5, 15 y 30 minutos elevan la duración estimada a varios meses. Esto ocurre porque, al aumentar el intervalo entre mediciones, el nodo permanece una mayor proporción del tiempo en modo *sleep*, reduciendo la corriente promedio del sistema.

En el contexto de ResiPark, este resultado permite justificar que el módulo de ocupación no debe operar mediante transmisión continua. Una estrategia más adecuada consiste en mantener el nodo en bajo consumo y despertarlo periódicamente o ante cambios de estado relevantes. De esta forma, el sistema conserva la capacidad de actualizar la disponibilidad de estacionamientos con una frecuencia suficiente para el caso residencial, pero evita consumir energía de manera innecesaria.

No obstante, esta estimación debe interpretarse como una aproximación de diseño y no como una medición final de laboratorio. La autonomía real dependerá de factores como la calidad de la batería, el consumo específico del regulador, la corriente en reposo del microcontrolador, la calibración del sensor, la potencia de transmisión, la frecuencia real de cambios de estado y las condiciones de propagación. Aun así, el análisis muestra que el uso de LoRa con operación en modo *sleep* es coherente con un nodo de sensado de baja tasa, ya que permite proyectar autonomías de semanas o meses bajo ciclos de transmisión discretos.

3.6.9. Estimación conceptual de latencia en enlace punto a punto para un nodo único

Con el fin de complementar la evaluación de la capa de comunicación, se analizó el comportamiento de la latencia en un escenario simplificado compuesto por un único nodo sensor y un *gateway* local, comunicados mediante LoRa P2P. Este escenario permite aislar el efecto de la distancia sin introducir interferencia por múltiples nodos, colisiones o competencia por el canal.

La Figura 3.19 muestra la latencia media estimada para paquetes recibidos exitosamente al aumentar la distancia entre el nodo y el *gateway*. Los resultados se mantienen cercanos a 233–234 ms en el rango evaluado entre 10 y 100 metros, evidenciando que, para un único enlace P2P, la distancia no introduce un aumento significativo de latencia.

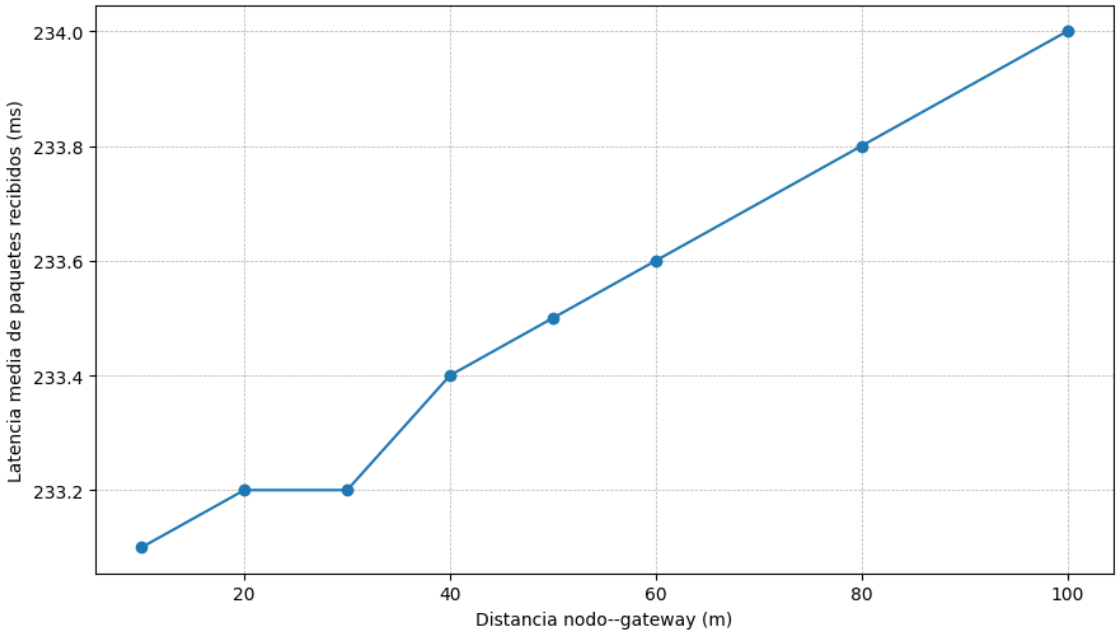


Figura 3.19: Latencia media del enlace LoRa P2P para un único nodo sensor al variar la distancia respecto del *gateway*.

La Figura 3.19 permite observar que la latencia permanece prácticamente estable a medida que aumenta la separación física entre ambos dispositivos. Este comportamiento es coherente con el funcionamiento esperado de un enlace LoRa P2P de baja tasa, donde la latencia de los paquetes recibidos está dominada principalmente por el tiempo en aire de la trama y por el procesamiento local, más que por el retardo de propagación de la señal.

Desde el punto de vista del diseño, este resultado es relevante porque confirma que, en un escenario de sensado discreto con un nodo único, el aumento de distancia afecta en mayor medida la robustez o probabilidad de entrega del enlace que la latencia de los paquetes exitosamente recibidos. Por lo tanto, para el módulo M2, la selección de LoRa P2P sobre módulos SX1276/78 no se justifica por una reducción extrema de latencia, sino por su capacidad de mantener comunicación robusta entre sensores distribuidos y un *gateway* local en entornos residenciales.

En consecuencia, el análisis de latencia punto a punto refuerza la separación entre dos criterios de diseño: por una parte, la latencia se mantiene dentro de un rango compatible con la actualización de estados de ocupación; por otra, la distancia debe evaluarse principalmente mediante métricas de confiabilidad, como la tasa de entrega de paquetes, RSSI o pérdida de paquetes. Esta distinción evita interpretar la comunicación LoRa como un enlace de baja latencia en sentido estricto y la sitúa correctamente como una tecnología adecuada para eventos discretos, de bajo tamaño y baja frecuencia.

3.6.10. Discusión y justificación de la decisión tecnológica

A partir de los resultados anteriores, la decisión tecnológica puede formularse de manera más precisa. El módulo M2 no requiere transportar video ni flujos continuos de datos, sino mensajes breves asociados a cambios de estado de ocupación. Bajo ese perfil, los resultados de DER, energía, latencia y pérdidas observadas respaldan el uso de LoRa P2P para el tramo sensor–gateway, siempre que el sistema opere dentro de un régimen de transmisión discreto y de baja frecuencia.

En particular, SF7 se adopta como configuración base para el despliegue residencial esperado, ya que ofrece una DER alta en distancias cortas y medias, con menor consumo energético y menor tiempo en aire que configuraciones más robustas. SF9 se considera una configuración de respaldo para escenarios con mayor atenuación estructural, como subterráneos o ubicaciones donde la distancia efectiva entre sensor y *gateway* degrade la confiabilidad del enlace. En cambio, SF12 no se propone como configuración normal de operación, debido a que, aunque entrega la mayor robustez, su costo energético y ocupación del canal resultan desproporcionados para un sistema que transmite mensajes breves, discretos y poco frecuentes.

La Tabla 3.1 resume esta política de configuración del *spreading factor*. La tabla no representa una regla fija para todos los entornos, sino una traducción de los resultados simulados a criterios prácticos de despliegue.

Tabla 3.1: Criterio de selección del *spreading factor* para el módulo M2.

Escenario	**SF propuesto**	**Justificación**
Despliegue residencial típico	SF7	Menor consumo energético y menor tiempo en aire, con confiabilidad suficiente para mensajes discretos de ocupación.
Entorno con mayor atenuación o mayor distancia sensor–gateway	SF9	Mejora la robustez del enlace con un costo aún aceptable para el tráfico esperado.
Escenario extremo / pruebas de cobertura	SF12	No recomendado para operación normal por su alto costo energético y mayor ocupación del canal. Puede utilizarse como referencia para pruebas de cobertura.

La comparación con alternativas como Zigbee o BLE debe realizarse en términos fun-

cionales y no solo nominales. BLE, aun siendo eficiente para proximidad y bajo consumo, presenta un alcance operativo más limitado y, por tanto, un menor margen para escenarios con atenuación estructural o separación física entre nodo y *gateway*. Zigbee constituye una alternativa válida para redes locales de corto o medio alcance, pero depende en mayor medida de un mallado estable y de la ubicación de nodos intermedios. En contraste, LoRa P2P se alinea mejor con un escenario donde el criterio dominante es cobertura sensor–gateway para mensajes discretos.

Tabla 3.2: Comparación funcional de tecnologías de comunicación consideradas en la arquitectura.

Tecnología	**Fortaleza principal**	**Limitación principal**	**Adecuación al sistema**
LoRa P2P	Alta cobertura y robustez para mensajes de baja tasa.	Mayor tiempo en aire y menor capacidad para tráfico intensivo.	Alta para M2. Se ajusta al envío de estados discretos de ocupación, priorizando alcance y confiabilidad por sobre caudal.
Zigbee	Buena alternativa para redes locales de corto o medio alcance.	Dependencia de mallado estable y posible necesidad de repetidores.	Media. Puede ser útil en despliegues locales, pero menos favorable cuando la prioridad es alcance sensor–gateway en entornos estructuralmente complejos.
BLE	Bajo consumo y buena operación en proximidad.	Alcance más limitado y menor margen ante atenuación estructural.	Baja para M2 cuando existe separación física relevante entre sensores y *gateway*.
Ethernet Cat 6	Alta estabilidad y capacidad sostenida.	Requiere infraestructura cableada.	Muy alta para el canal de video, pero no necesaria para el sensado discreto por plaza.

La Tabla 3.2 sintetiza esta comparación desde el punto de vista del caso de uso. La inclusión de Ethernet Cat 6 no busca presentarla como alternativa al sensado inalámbrico por plaza, sino contrastar su rol dentro de la arquitectura general: es adecuada para tráfico continuo y de mayor caudal, como el video, pero innecesaria para mensajes discretos de ocupación.

En el caso del subsistema de video, el perfil de tráfico es completamente distinto al de los sensores de ocupación. Mientras M2 transmite eventos discretos y de baja tasa, el video requiere capacidad sostenida, estabilidad y menor sensibilidad a interferencias. Por esta razón, en la arquitectura propuesta el transporte de video se resuelve mediante Ethernet sobre cableado Cat 6, dejando LoRa P2P exclusivamente para el tramo de sensado inalámbrico de baja tasa.

La Tabla 3.3 resume la decisión final de comunicación por tramo. Esta separación permite asignar cada tecnología al tipo de tráfico que mejor resuelve, evitando sobredimensionar los sensores de ocupación y, al mismo tiempo, asegurando estabilidad para el canal de video.

Tabla 3.3: Síntesis comparativa de las decisiones de comunicación en la arquitectura propuesta.

Tramo	**Tecnología seleccionada**	**Justificación principal**
Sensor de estacionamiento – *gateway* local	LoRa P2P sobre SX1276/78	Adecuada para mensajes discretos, de bajo tamaño y baja frecuencia, con prioridad en cobertura y robustez del enlace por sobre ancho de banda.
Canal de video hacia nodo de borde / backend	Ethernet Cat 6	Adecuada para tráfico continuo y de mayor caudal, al ofrecer estabilidad, capacidad sostenida y menor sensibilidad a interferencias que un medio inalámbrico compartido.

En consecuencia, la arquitectura híbrida propuesta no responde a una decisión arbitraria, sino a la asignación de cada tecnología al tipo de tráfico que resuelve de mejor manera. La evaluación de resultados muestra que LoRa P2P es adecuada para sensado discreto de baja tasa, mientras que la discusión tecnológica permite justificar Ethernet Cat 6 para el transporte de video. Esta separación cierra el diseño conceptual de la capa de comunicación del sistema y prepara el terreno para el capítulo siguiente, donde se aborda la implementación y despliegue del prototipo en entorno real.

3.6.11. Evaluación de la capa de comunicación

Los resultados obtenidos permiten analizar el comportamiento del enlace LoRa P2P para el módulo M2 bajo el perfil de tráfico esperado: mensajes discretos, de bajo tamaño y asociados al estado de ocupación de una plaza. En este subsistema, el requerimiento dominante no es el ancho de banda, sino la capacidad de mantener una comunicación confiable entre los nodos sensores y el *gateway* local en condiciones de propagación potencialmente adversas.

A partir de las campañas anteriores, se observa que el *spreading factor* actúa como un parámetro de ajuste entre robustez, latencia y consumo energético. En este trabajo, los resultados permiten adoptar una configuración intermedia como punto de partida para el prototipo, evitando tanto el menor margen de robustez de configuraciones más rápidas como el mayor tiempo en aire de configuraciones más conservadoras. Esta decisión no constituye el foco principal de la arquitectura, sino una configuración práctica del enlace LoRa implementado sobre los módulos SX1276/78.

En segundo lugar, la evaluación de densidad mostró que el enlace mantiene una DER alta incluso al aumentar el número de nodos, siempre que el sistema opere en un régimen de tráfico compatible con el sensado discreto. Esto indica que la arquitectura puede escalar a un

número razonable de plazas instrumentadas sin entrar inmediatamente en un régimen severo de degradación.

En tercer lugar, la campaña de latencia bajo incremento de carga permitió observar que la latencia de los paquetes recibidos exitosamente permaneció prácticamente constante dentro de cada configuración evaluada. La degradación del sistema no se manifestó principalmente como un crecimiento abrupto de la latencia, sino como una caída progresiva de la DER cuando aumentó la frecuencia de transmisión y la cantidad de nodos activos.

Esta observación es relevante porque permite separar dos efectos distintos. Por una parte, los paquetes que llegan correctamente mantienen una latencia compatible con el caso de uso de ocupación de estacionamientos. Por otra, al aumentar la carga del canal, la confiabilidad disminuye debido al incremento de colisiones y paquetes débiles. En consecuencia, LoRa P2P se comporta adecuadamente para sensado discreto de baja frecuencia, pero no debe utilizarse como tecnología para tráfico intensivo o continuo.

Desde esta perspectiva, la evaluación de la capa de comunicación permite pasar desde los resultados de simulación hacia una decisión concreta de implementación: utilizar módulos LoRa basados en SX1276/78 para el tramo sensor–gateway del módulo M2. Estos módulos permiten implementar el enlace inalámbrico P2P entre el nodo de sensado y el receptor local, configurando los parámetros de radio de acuerdo con las condiciones reales del despliegue.

La Tabla 3.4 sintetiza la comparación funcional entre las tecnologías consideradas para el tramo sensor–gateway. La comparación no busca establecer una jerarquía universal, sino evaluar la adecuación de cada tecnología al patrón de tráfico del módulo M2.

Tabla 3.4: Comparación de tecnologías para el tramo sensor–gateway del módulo M2.

Tecnología	**Fortaleza principal**	**Limitación principal**	**Adecuación al caso M2**
LoRa P2P	Alta cobertura y robustez para mensajes de baja tasa.	Mayor tiempo en aire y menor capacidad para tráfico intensivo.	Alta. Se ajusta al envío de estados discretos de ocupación, priorizando alcance y confiabilidad por sobre caudal.
Zigbee	Buena alternativa para redes locales de corto o medio alcance.	Dependencia de mallado estable y posible necesidad de repetidores.	Media. Puede ser útil en despliegues locales, pero menos favorable cuando la prioridad es alcance sensor–gateway en entornos estructuralmente complejos.
BLE	Bajo consumo y buena operación en proximidad.	Alcance más limitado y menor margen ante atenuación estructural.	Baja. Resulta menos conveniente para enlaces entre sensores y *gateway* cuando existe separación física relevante o entornos adversos.
Ethernet Cat 6	Alta estabilidad y capacidad sostenida.	Requiere infraestructura cableada.	Muy alta para el canal de video, pero no necesaria para el sensado discreto por plaza.

La Tabla 3.4 evidencia que LoRa P2P es la alternativa más alineada con el tramo sensor–gateway del módulo M2, ya que privilegia cobertura y robustez para mensajes discretos. En cambio, Ethernet Cat 6 se considera dentro de la comparación para destacar su rol complementario en la arquitectura: no como alternativa para cada sensor de ocupación, sino como medio adecuado para tráfico continuo y de mayor caudal, como el transporte de video.

3.6.12. Discusión y justificación de la decisión tecnológica

En conjunto, los resultados de simulación permiten sostener que la elección de LoRa P2P para el módulo M2 responde de forma coherente al perfil funcional del sistema. En este subsistema, el requerimiento dominante no es la transmisión de grandes volúmenes de datos, sino la capacidad de transportar mensajes cortos de manera confiable a través de distancias medias y en entornos con condiciones de propagación potencialmente adversas.

Esta decisión se materializa en la arquitectura mediante el uso de módulos SX1276/78, los cuales actúan como transceptores LoRa en el enlace P2P entre el nodo sensor y el *gateway*. En el extremo de sensado, el microcontrolador obtiene el estado de ocupación de la plaza y lo empaqueta como un mensaje breve. Luego, el módulo SX1276/78 transmite dicho mensaje mediante LoRa hacia el receptor local. En el extremo opuesto, otro módulo compatible recibe

la trama y permite que el *gateway* publique el evento hacia los servicios centrales mediante la red IP del edificio.

De esta manera, el módulo LoRa no se entiende como un componente aislado, sino como la pieza de comunicación que conecta la capa física de sensado con la capa de soporte del sistema. Esta separación permite que los sensores operen con una tecnología adecuada para bajo consumo y cobertura, mientras que el *gateway* se encarga de traducir los eventos recibidos hacia el backend, donde se actualiza la disponibilidad de estacionamientos y se registra la trazabilidad correspondiente.

En contraste, el canal de video responde a un perfil completamente distinto, caracterizado por tráfico continuo y mayores requerimientos de capacidad y estabilidad. Por esta razón, en la implementación del prototipo el transporte de video se resolvió mediante *Ethernet* sobre cableado Cat 6. Esta decisión se adoptó por ser una alternativa más adecuada para flujos continuos de mayor caudal, al ofrecer estabilidad, menor sensibilidad a interferencias y capacidad sostenida superior a la de un canal inalámbrico compartido.

La Tabla 3.5 resume la decisión final de comunicación por tramo. Esta separación permite asignar cada tecnología al tipo de tráfico que mejor resuelve, evitando sobredimensionar los sensores de ocupación y, al mismo tiempo, asegurando estabilidad para el canal de video.

Tabla 3.5: Síntesis comparativa de las decisiones de comunicación en la arquitectura propuesta.

Tramo	**Tecnología seleccionada**	**Justificación principal**
Sensor de estacionamiento – *gateway* local	LoRa P2P sobre SX1276/78	Adecuada para mensajes discretos, de bajo tamaño y baja frecuencia, con prioridad en cobertura y robustez del enlace por sobre ancho de banda.
Canal de video hacia nodo de borde / backend	Ethernet Cat 6	Adecuada para tráfico continuo y de mayor caudal, al ofrecer estabilidad, capacidad sostenida y menor sensibilidad a interferencias que un medio inalámbrico compartido.

En consecuencia, la arquitectura híbrida propuesta no responde a una decisión arbitraria, sino a la asignación de cada tecnología al tipo de tráfico que resuelve de mejor manera. LoRa P2P sobre SX1276/78 se utiliza para el sensado discreto de ocupación porque permite conectar nodos distribuidos con un *gateway* local de forma robusta y eficiente. Ethernet Cat 6 se reserva para el transporte de video porque responde a un perfil de tráfico continuo y de mayor demanda. Esta separación cierra el diseño conceptual de la capa de comunicación del sistema y prepara el terreno para el capítulo siguiente, donde se aborda la implementación y despliegue del prototipo en entorno real.

Capítulo 4

Implementación y Despliegue de la Solución

4.1. Implementación de la Arquitectura

4.1.1. Contexto General del Prototipo

En este capítulo se describe la implementación del sistema ResiPark a partir del diseño presentado en el Capítulo 3. En esta etapa, el foco está puesto en el prototipo funcional que integra los distintos módulos de visión por computador, sensado IoT, servicios centrales e interfaces de usuario.

Actualmente, el prototipo permite:

- Captura de video en tiempo real desde cámaras instaladas en el acceso vehicular.
- Procesamiento en Google Colab de modelos de detección para patentes y tipo de vehículo.
- Validación y almacenamiento de resultados en una base de datos SQL centralizada.
- Gestión de reservas de estacionamientos de visita por parte de los residentes.
- Uso de una interfaz web orientada a conserjería y una aplicación móvil para residentes.
- Monitoreo de ocupación de estacionamientos mediante un nodo IoT basado en ESP32 con comunicación LoRa hacia el backend.

Estas capacidades implementadas permiten cubrir el flujo principal del sistema: desde la captura de imágenes y el sensado de los estacionamientos, hasta la presentación de la información procesada a los distintos perfiles de usuario.

4.1.2. Descripción de Componentes

Siguiendo la descomposición modular planteada en el diseño, la implementación del prototipo se estructura en torno a cuatro grandes módulos:

- **M1 – Visión por Computador y OCR**: incluye los modelos de YOLOv11 utilizados para la detección de vehículos y patentes, junto con el modelo Microsoft Florence-2 para la transcripción OCR de las placas. Este módulo se ejecuta en un entorno de cómputo en la nube (Colab) y expone sus resultados a través de la API.

- **M2 – Sensado**: está compuesto por un ESP32 conectado a un sensor de proximidad ultrasónico HC-SR04 encargado de detectar la ocupación de los estacionamientos de visitas. La información capturada se envía periódicamente al backend para mantener actualizado el estado de cada plaza.
- **M3 – Interfaces de Usuario**: considera la aplicación móvil para residentes (registro y autenticación, reservas, consulta de historial) y la aplicación web para conserjería (monitoreo de accesos, visualización de historial y administración de patentes). Ambas se comunican con los servicios centrales mediante la API REST.
- **M4 – Servicios Centrales**: corresponde a la API REST, la base de datos SQL y la lógica de integración entre los módulos de IA, sensado e interfaces. Este módulo articula los flujos de información, aplica reglas de negocio y persiste los datos relevantes para el funcionamiento del sistema.

Esta organización modular permite que el desarrollo y la evolución de cada componente se realicen de forma relativamente independiente, facilitando la integración y futuras extensiones del sistema.

4.2. Procesamiento de datos

4.2.1. Infraestructura de adquisición de datos y trazabilidad operativa

Esta sección describe el diseño e implementación de la infraestructura física que permite que ResiPark opere como un sistema ciberfísico: sensores y cámara capturan señales del entorno, un componente de borde normaliza y transporta los datos, y el backend transforma esas lecturas en eventos auditables que habilitan acciones (notificación y apertura del portón). El énfasis está en el esqueleto de adquisición y comunicación, ya que la confiabilidad del sistema depende más de cómo se mide y transporta la información que de cómo se representa internamente.

El diseño considera dos canales de adquisición independientes pero convergentes. El primero corresponde al acceso vehicular: una cámara IP captura frames en una zona de control definida y un equipo de borde (Raspberry Pi) se encarga del muestreo, envío a inferencia y ejecución del actuador. El segundo corresponde a la ocupación de estacionamientos de visita: un nodo por plaza mide el estado físico ocupado/libre y transmite cambios por LoRa hacia un receptor ubicado en un punto con mejor conectividad, desde donde el estado se publica hacia los servicios centrales.

Canal de acceso vehicular: cámara, borde y control del actuador

En el acceso, la cámara IP se instala de forma que la patente quede capturada en un encuadre estable cuando el vehículo reduce velocidad o se detiene en la zona de control. La Raspberry Pi consume el stream de la cámara y aplica un muestreo para evitar saturación del enlace y del servicio de inferencia, priorizando estabilidad por sobre frecuencia máxima.

Cada frame seleccionado se envía al servicio de detección, y el resultado se normaliza en un evento de acceso que incluye, al menos, la patente detectada, el tipo de vehículo estimado, un timestamp y un nivel de confianza asociado.

La apertura del portón se implementa como una acción condicionada. La Raspberry Pi controla un relé que genera un pulso de apertura únicamente cuando el evento de acceso pasa las validaciones definidas: presencia real de vehículo, lectura de patente no vacía y verificación de autorización vigente. Este enfoque reduce falsos positivos, ya que evita accionar el portón ante lecturas espurias o ante escenas sin vehículo, y concentra la responsabilidad del control físico en un componente de borde que mantiene el acoplamiento con el actuador bajo control.

Canal de estacionamientos: sensado por plaza y enlace LoRa

Para estacionamientos de visita, el sistema instrumenta cada plaza como un punto de medición independiente. Un nodo IoT basado en ESP32 toma lecturas periódicas del sensor instalado en la plaza y deriva un estado discreto (libre u ocupado). Para mejorar confiabilidad, las transiciones de estado se filtran temporalmente antes de ser transmitidas, evitando cambios espurios por ruido o lecturas puntuales. Una vez confirmado un cambio, el nodo transmite un mensaje por LoRa hacia un receptor (ESP32 con módulo LoRa) ubicado estratégicamente en una zona con mejor cobertura y acceso estable a la red local.

El receptor funciona como gateway: recibe estados LoRa, los encapsula y los publica hacia los servicios centrales usando conectividad IP. Este desacoplamiento permite que el tramo más complejo (subterráneos o zonas sin cableado) se resuelva con radio de bajo consumo, mientras que el tramo hacia backend y aplicaciones utilice un enlace estándar (WiFi o Ethernet) con mejor capacidad y estabilidad.

Convergencia de eventos y persistencia mínima

Ambos canales convergen en el backend como eventos normalizados: accesos vehiculares y cambios de ocupación. A partir de estos eventos, el sistema actualiza el estado operativo visible para conserjería y residentes, y mantiene trazabilidad para auditoría y verificación en terreno. En esta memoria, la base de datos se utiliza principalmente como soporte de persistencia y reconstrucción histórica (eventos de acceso, estados de ocupación, reservas activas), sin que el valor del sistema dependa de complejidad relacional, sino de la consistencia entre el estado físico observado y el estado digital publicado.

Nota sobre identidades y roles

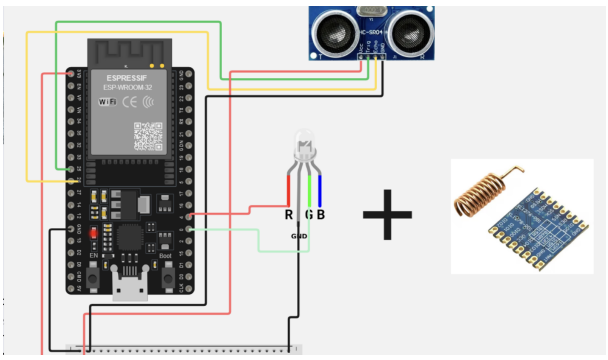
A nivel de operación, ResiPark distingue perfiles de uso para controlar funcionalidades y trazabilidad. En la aplicación móvil, los residentes operan a nivel de departamento para gestionar reservas y consultas. En la aplicación web, el personal de conserjería opera con cuenta individual para monitorear eventos y gestionar acciones. Los datos de visitas se registran en el contexto de una reserva activa, ya que no requieren autenticación persistente. Estos elementos se presentan como soporte operativo y no como eje central del capítulo, dado que el foco de la implementación está en la adquisición de datos, comunicación y control físico del sistema.

4.3. Infraestructura electrónica y montaje en terreno

Esta sección documenta el despliegue físico de la capa de obtención de datos y de la capa de comunicación, detallando los componentes electrónicos instalados, su cableado, encapsulado y rol dentro del flujo extremo a extremo. La presentación se organiza por subsistemas, combinando (i) diagramas de conexión, que especifican la integración eléctrica y de señal, y (ii) evidencia fotográfica del montaje real, con el objetivo de justificar las decisiones de diseño adoptadas durante las pruebas en terreno.

4.3.1. Nodo de sensado de ocupación (TX): sensor ultrasónico y transmisor LoRa

El primer subsistema mostrado en 4.1 corresponde al nodo transmisor instalado en la plaza de estacionamiento. Este nodo mide la distancia al obstáculo (vehículo) mediante un sensor ultrasónico, discretiza el resultado en un estado ocupado/libre y transmite únicamente los cambios de estado a través de LoRa. Esta decisión reduce el tráfico y el consumo energético, manteniendo un canal robusto para subterráneos o zonas con atenuación estructural. Para proteger la electrónica y facilitar la instalación, el conjunto se encapsuló en una carcasa impresa en 3D adaptada a la geometría y condiciones del lugar.



(a) Diagrama de conexiones: ESP32 + HC-SR04 + módulo LoRa (TX) + Led Indicador de ocupación.

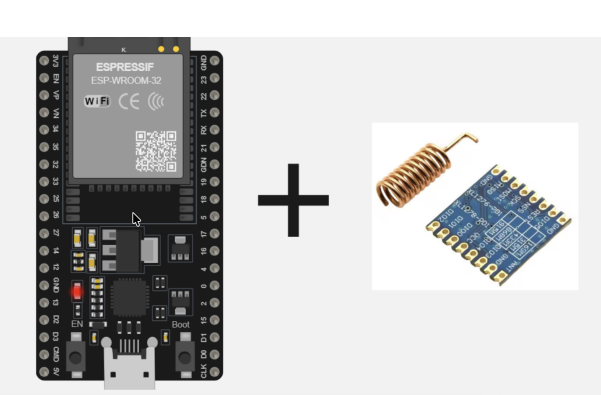


(b) Montaje real del nodo TX en carcasa impresa en 3D.

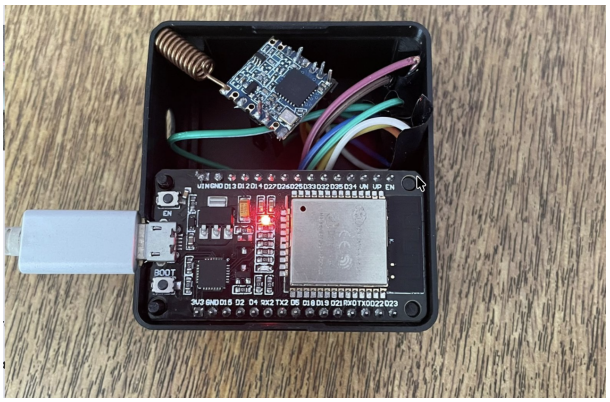
Figura 4.1: Nodo de sensado de ocupación (TX) utilizado para medir presencia vehicular y transmitir el estado por LoRa.

4.3.2. Nodo receptor/gateway (RX): recepción LoRa y publicación vía WiFi

El segundo subsistema mostrado en 4.2 corresponde al receptor LoRa, ubicado en una zona con mejor cobertura y acceso estable a la red local. Su función es recibir tramas LoRa desde el/los transmisores, normalizar el mensaje (identificador del dispositivo/plaza, estado y timestamp) y publicar el evento hacia el backend mediante WiFi. Este desacoplamiento permite usar LoRa en el tramo físicamente adverso y WiFi/IP como backhaul hacia los servicios centrales. Al igual que el transmisor, el receptor se encapsuló en una carcasa impresa en 3D para protección mecánica y montaje rápido.



(a) Diagrama de conexiones: ESP32 + módulo LoRa (RX) + WiFi integrado.

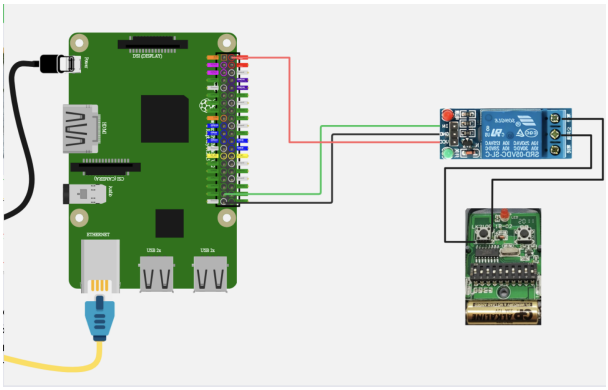


(b) Montaje real del nodo RX en carcasa impresa en 3D.

Figura 4.2: Nodo receptor/gateway (RX) encargado de recibir estados por LoRa y enviarlos al backend mediante WiFi.

4.3.3. Nodo de borde para actuación: Raspberry Pi 4B y relé de apertura

El tercer subsistema mostrado en 4.3 corresponde al nodo de borde instalado en el acceso vehicular, implementado con una Raspberry Pi 4B y un relé de un canal. La Raspberry opera en modo de escucha activa (servicio residente) para recibir instrucciones de apertura desde la nube/backend, y generar el pulso eléctrico sobre el relé que emula el accionamiento del control remoto del portón. Esta decisión permite integrar el actuador sin intervenir el motor ni modificar la infraestructura existente, manteniendo el control en una interfaz eléctrica simple y reversible. Dado que el nodo opera en un entorno con polvo y exposición ambiental indirecta, se incorporó una caja protectora para reducir ingreso de humedad y partículas sobre los componentes electrónicos.



(a) Diagrama de conexiones: Raspberry Pi 4B + relé 1CH + control remoto.

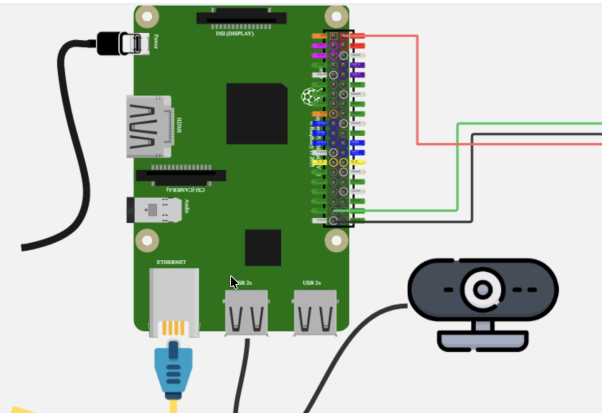


(b) Montaje real del nodo de actuación en caja protectora.

Figura 4.3: Nodo de borde para actuación: ejecución del pulso de apertura condicionado por una instrucción validada desde el backend.

4.3.4. Nodo de borde para captura: Raspberry Pi y cámara para entrada de IA

El cuarto subsistema mostrado en 4.4 describe la captura de video utilizada como entrada del pipeline de IA. El diseño está concebido para operar con una cámara IP de exterior con certificación de protección (por ejemplo, IP66/IP67 o superior, incluyendo alternativas IP68), adecuada para polvo y lluvia. Sin embargo, para las pruebas en terreno se optó por una webcam debido a criterios de practicidad, facilidad de montaje, disponibilidad inmediata y reducción de complejidad de configuración en una fase de validación experimental. La arquitectura mantiene compatibilidad para migrar a una cámara IP en despliegues posteriores, ya sea consumiendo el stream por RTSP/ONVIF o mediante captura equivalente en el nodo de borde.



(a) Diagrama de conexiones: Raspberry Pi + cámara (webcam en pruebas / cámara IP en diseño).



(b) Montaje real de la cámara en el acceso durante las pruebas en terreno.

Figura 4.4: Subsistema de captura de video para IA: configuración usada en terreno y diseño compatible con cámara IP de exterior.

4.3.5. Justificación de decisiones de diseño

Las decisiones de diseño asociadas al despliegue físico se fundamentan directamente en la arquitectura propuesta en la Figura 3.11, integrando el módulo de estacionamiento inteligente basado en WSN/IoT con los pipelines de percepción orientados al reconocimiento automático de patentes (ALPR) en tiempo real. En el caso de la ocupación de plazas, la elección de una red de sensores con nodos por espacio responde directamente a patrones reportados en sistemas de estacionamiento inteligente, donde la sensorización distribuida permite medir disponibilidad de manera verificable y reducir dependencia de inferencias visuales continuas [10, 11, 12]. En esa misma línea, la separación por capas entre sensado, comunicación y servicios centrales replica el enfoque de arquitecturas en capas descrito en revisiones y propuestas del dominio, facilitando integración con aplicaciones y reglas de negocio [11, 35]. Bajo este marco, se adopta un enlace LoRa para el tramo más adverso (plazas alejadas o con atenuación estructural) y un backhaul IP (WiFi/Ethernet) desde el gateway hacia el backend, manteniendo un desacoplamiento entre la red de campo y los servicios centrales, tal como se recomienda en diseños orientados a escalabilidad y mantenibilidad [11, 12].

Para el acceso vehicular, el diseño se alinea con la literatura que sugiere pipelines modulares para ALPR, donde la percepción (detección/lectura) se desacopla de la lógica de decisión (autorización y registro) para mejorar trazabilidad y permitir calibración por etapas [20, 37]. En el prototipo, la captura de video se concentra en una zona de control para estabilizar condiciones de entrada, y el nodo de borde (Raspberry Pi) realiza muestreo y orquestación del flujo hacia inferencia, con el objetivo de controlar latencia y consumo de recursos. La decisión de accionar el portón mediante un relé que emula el pulso del control remoto responde a un criterio de integración no intrusiva: permite automatizar sin intervenir el motor ni modificar la infraestructura base, manteniendo reversibilidad y reduciendo riesgos de instalación. Esta aproximación es coherente con arquitecturas de edificios inteligentes que integran dispositivos distribuidos, servicios centrales y clientes de operación, privilegiando la interoperabilidad con

el entorno físico existente [4, 1].

Finalmente, la selección de una webcam para las pruebas se explica por criterios experimentales y de practicidad en terreno (montaje rápido y disponibilidad inmediata), sin alterar la arquitectura objetivo: el sistema queda preparado para migrar a una cámara de exterior con certificación IP66/IP67 o superior (IP68), compatible con despliegues prolongados bajo polvo y lluvia. En conjunto, estas decisiones evidencian una implementación guiada por la literatura revisada: sensado distribuido para disponibilidad verificable, arquitectura por capas para integración, y pipeline ALPR modular con control de actuación condicionado por validación, priorizando seguridad, trazabilidad y factibilidad de despliegue en comunidades residenciales.

4.4. Despliegue de la solución

En esta sección se describe el despliegue del prototipo de ResiPark en un entorno real, así como las pruebas de funcionamiento, los mecanismos de gestión de riesgos implementados en la solución y algunos casos borde identificados durante la experimentación. El entorno elegido corresponde a un pasaje residencial con portón vehicular motorizado, cuyas características son comparables a las de un condominio de la inmobiliaria Norte Verde pero a pequeña escala.

4.4.1. Validación del sistema en un entorno real

Para validar el sistema en condiciones reales se instalaron los componentes del prototipo en un pasaje residencial que cumple con los requisitos mínimos de infraestructura:

- **Portón vehicular motorizado** con control remoto, compatible con el relé controlado desde la Raspberry Pi.
- **Alimentación eléctrica** estable en el sector del portón, necesaria para la Raspberry Pi, la cámara IP y el módulo de relé.
- **Conectividad Ethernet** cercana al portón, utilizada para enlazar la cámara IP y la Raspberry Pi con el router del pasaje, reduciendo la latencia de transmisión de video.
- **Cobertura WiFi** en la zona de estacionamientos de visita, requerida para la comunicación entre el nodo IoT de ocupación y el backend.

La Figura 4.5 muestra el portón utilizado como entorno de pruebas. El círculo rojo de la izquierda indica la ubicación de la caja de componentes (Raspberry Pi, relé y conexiones eléctricas), mientras que el círculo de la derecha señala la posición de la cámara IP orientada hacia la calle. Esta configuración permite capturar el vehículo antes de que llegue al portón y, al mismo tiempo, mantener los elementos electrónicos protegidos y con acceso directo al motor del portón a través del control remoto cableado.

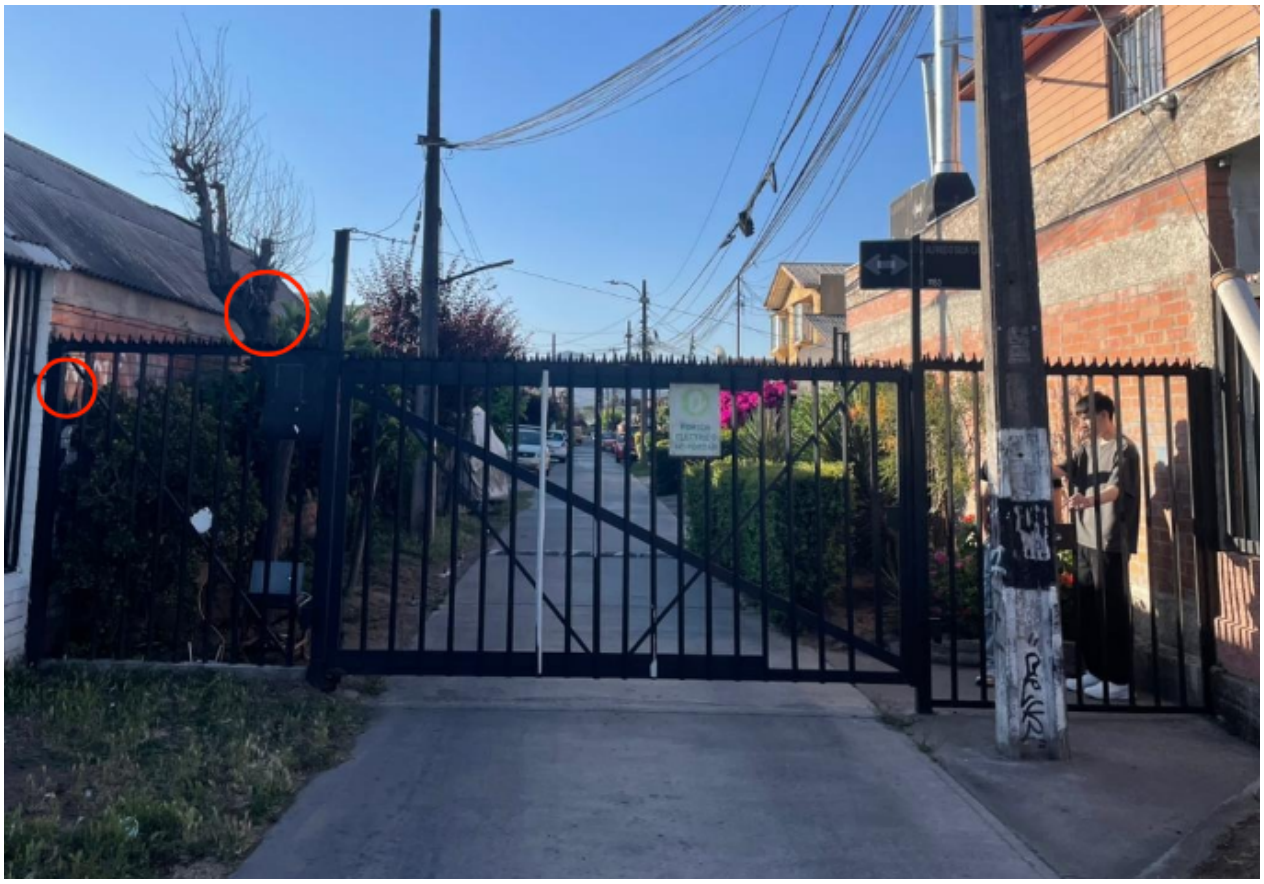


Figura 4.5: Entorno real de despliegue del sistema en un pasaje residencial. En rojo se destaca la caja de componentes y la cámara IP.

La validación en este entorno se centró en verificar que el flujo de acceso funcionara de extremo a extremo:

1. El vehículo se aproxima al portón y es capturado por la cámara IP.
2. El flujo de video se envía al servicio de detección en la nube, que identifica la presencia de un vehículo y lee la patente.
3. El backend verifica la patente contra la base de datos de vehículos residentes y reservas vigentes.
4. En caso de ser válida, se registra el ingreso y la Raspberry Pi activa el relé para activar la apertura del portón.

Este flujo se ejercitó tanto con vehículos residentes como con patentes registradas como visitas, comprobando que el sistema sólo emite la señal de apertura cuando la patente detectada se encuentra efectivamente autorizada. Los resultados obtenidos durante las pruebas se encuentran alineados con las métricas de precisión reportadas para el sistema de detección de vehículos y patentes (99,7 % en detección de vehículo y 98,9 % en detección de patente)

4.4.2. Testing

El proceso de testing consideró tanto los componentes de acceso vehicular como el subsistema de estacionamientos de visita.

Testing del acceso vehicular

Para evaluar el comportamiento del sistema de visión por computador se utilizaron distintos escenarios de prueba:

- Videos de un vehículo aproximándose al portón desde distintas distancias y ángulos, en condiciones de iluminación diurna (Figura 4.6).
- Secuencias donde se presentan patentes en situaciones no válidas (por ejemplo, una persona sosteniendo una placa frente a la cámara) con el fin de verificar que el sistema no abra el portón en estos casos (Figura 4.8).
- Pruebas con videos sin vehículos, para confirmar que no se registran accesos ni se generan falsas aperturas.
- Sistema con video en vivo para comprobar que se espera a que ingrese un vehículo al recinto.



Figura 4.6: Secuencia de prueba con vehículo en movimiento utilizada para validar el flujo de detección y registro de ingresos.

En todos los casos se monitoreó el cliente de procesamiento ejecutado en la Raspberry Pi, observando que sólo cuando la detección incluía un vehículo con confianza suficiente y una

patente legible se avanzaba a la etapa de verificación contra la base de datos. Esto permitió confirmar que el flujo de decisión previo a la apertura del portón se comporta de acuerdo con lo diseñado.

Testing del sistema de estacionamientos

El subsistema de estacionamientos se probó en un espacio de estacionamiento real mostrado en la Figura 4.7, instalando el sensor ultrasónico HC-SR04 sobre un árbol cercano, protegido por una caja impresa en 3D y fijado mediante adhesivos.



Figura 4.7: Vista de la caja impresa en 3D que alberga el sensor ultrasónico HC-SR04 para detectar ocupación de estacionamiento.

El sensor se conectó a un ESP32 transmisor que envía periódicamente la distancia medida hacia un ESP32 receptor mediante LoRa. A partir de estas lecturas, el backend actualiza el estado del estacionamiento (libre u ocupado) en la base de datos, reflejando los cambios en el dashboard web. En las pruebas realizadas se comprobó que el sensor mantiene lecturas confiables para distancias menores a 60 cm, rango que resulta suficiente para discriminar entre plaza libre y vehículo presente.

4.5. Gestión de riesgos, validación en terreno y casos borde

La gestión de riesgos del prototipo se abordó de manera secuencial, siguiendo el mismo orden en que el sistema toma decisiones en operación real: primero se aseguran controles básicos a nivel de arquitectura, luego se endurece la lógica de detección y autorización, y finalmente se validan comportamientos mediante pruebas controladas en terreno. Esta organización permite mostrar explícitamente que las medidas de mitigación no son teóricas, sino que se reflejan en el flujo extremo a extremo implementado (captura → inferencia → verificación → actuación).

4.5.1. Controles a nivel de arquitectura

En el nivel de arquitectura se incorporaron medidas orientadas a reducir exposición de credenciales y proteger la comunicación entre componentes. En particular, se aplicaron:

- Uso de **hash de contraseñas** al momento de persistir credenciales, evitando almacenamiento en texto plano.
- Protección de la comunicación en red mediante **WPA2** en el enlace WiFi local y **HTTPS** para servicios expuestos hacia Internet. En el caso del enlace LoRa, el cifrado de payload (p. ej., AES-128) se considera una medida aplicable para endurecer el canal cuando el despliegue lo requiera, ya sea mediante esquema propio o bajo un perfil LoRaWAN.

Estos controles establecen una base mínima para operar el prototipo sin exponer información sensible en tránsito o reposo, y delimitan prácticas necesarias para un despliegue productivo.

4.5.2. Mitigación principal en operación: lógica escalonada de autorización

A nivel operativo, el riesgo más relevante identificado es la suplantación de identidad mediante patentes falsas o presentaciones engañosas frente a la cámara (por ejemplo, una placa impresa sostenida por una persona). Para acotar este riesgo, el cliente de procesamiento ejecutado en la Raspberry Pi implementa una secuencia de verificaciones que actúa como gating de seguridad: cada etapa habilita la siguiente y, ante cualquier fallo, el sistema detiene el proceso sin accionar el portón.

1. **Detección de vehículo**: El sistema solo continúa si el modelo identifica un objeto de tipo vehículo con confianza sobre un umbral definido. Si no se detecta vehículo, el proceso finaliza y no se intenta leer patente.
2. **Lectura de patente**: Únicamente si se detecta vehículo se ejecuta OCR sobre la región correspondiente. Lecturas vacías o marcadas como `''No detectada''` se descartan.

3. **Verificación en el backend**: Si se obtiene una patente válida, el cliente consulta al servicio de verificación. El backend evalúa autorización vigente considerando vehículos residentes y reservas activas de visita.

4. **Registro y apertura**: Solo cuando la verificación es exitosa se registra el evento y se ejecuta la acción física (script remoto) que activa el relé de apertura.

Esta lógica garantiza que la actuación (apertura) siempre ocurre después de una cadena de evidencia mínima: presencia real de vehículo, lectura de patente no vacía y autorización vigente.

4.5.3. Pruebas controladas del comportamiento esperado

Para comprobar que el flujo anterior se comporta de acuerdo con las restricciones de seguridad, se ejecutaron tres escenarios ilustrativos:

- **Video sin vehículos**: el cliente reporta ausencia de detecciones de vehículo y finaliza sin generar aperturas ni registros.
- **Persona sosteniendo una patente**: como se muestra en la Figura 4.8, el modelo no detecta un vehículo completo; por lo tanto, la etapa de OCR no se activa y el portón permanece cerrado.



Figura 4.8: Escenario de prueba: una persona sostiene una patente frente a la cámara. El sistema no abre el portón al no detectar un vehículo.

- **Vehículo real con patente no registrada**: el sistema detecta vehículo y lee la patente, pero la verificación falla al no existir autorización vigente. La salida del cliente refleja el fallo controlado y no se ejecuta el comando de apertura (Figura 4.9).

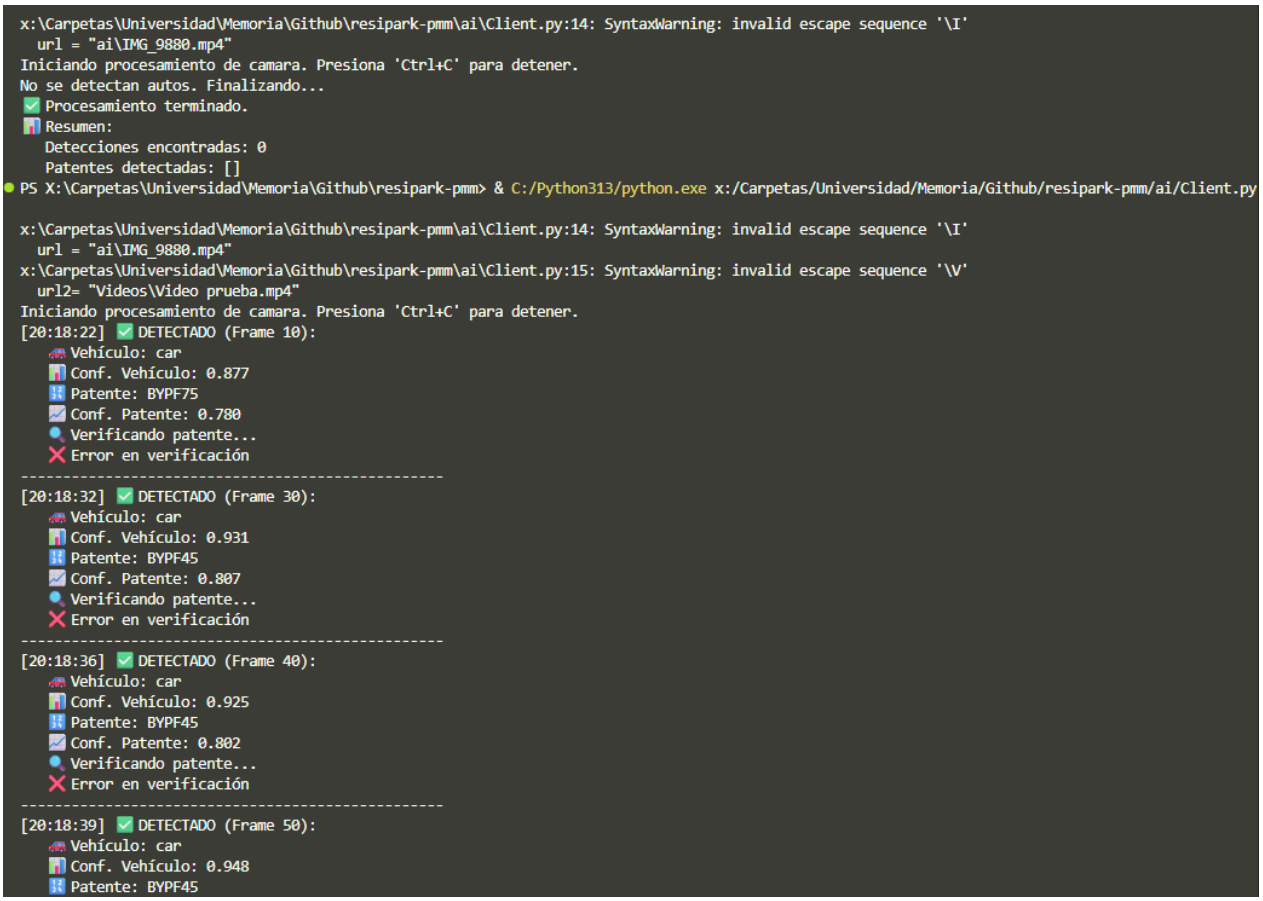


Figura 4.9: Salida del cliente en Raspberry Pi: detecciones de vehículo y fallos controlados durante la verificación de patente, sin accionar la apertura.

En conjunto, estos resultados acotan el alcance del prototipo: el sistema se comporta de manera conservadora y evita aperturas automáticas ante ausencia de vehículo o falta de autorización vigente, lo que confirma la efectividad del gating implementado para escenarios básicos de ataque o error.

4.5.4. Casos borde observados y degradación a human-in-the-loop

Además de los escenarios anteriores, las pruebas revelaron situaciones donde existe un vehículo en escena pero la evidencia visual no permite una decisión automática con seguridad aceptable. Estos casos no invalidan el prototipo, pero delimitan su cobertura y motivan una política explícita de degradación: ante ambigüedad, el sistema debe generar una alerta y transferir la decisión a conserjería, preservando trazabilidad y evidencia.

Vehículo con patente dibujada o adulterada

Se observó el caso en que la placa está dibujada, mostrado en la Figura 4.10, sobrepuesta o alterada (p. ej., caracteres con plumón o tipografía no estándar). Aunque el detector puede localizar un bounding box plausible, el OCR tiende a perder confiabilidad por variaciones de tipografía y espaciado, bordes irregulares y artefactos de iluminación o compresión. Para mitigar falsos positivos, el sistema aplica umbrales mínimos de confianza y verificación sintáctica de placa chilena (patrón alfanumérico y longitud esperada). Si la lectura no supera los umbrales o falla la validación del formato, el sistema no acciona la apertura y deriva el evento a revisión manual.



Figura 4.10: Caso borde ilustrativo: patente dibujada/adulterada. La detección puede ocurrir, pero la lectura OCR suele degradarse, por lo que no se considera evidencia suficiente para apertura automática.

Vehículo con la misma patente pero color distinto al registrado

Otro caso crítico ocurre cuando la patente coincide con un registro válido, pero el vehículo presenta atributos físicos inconsistentes (en particular, el color), como se muestra en la Figura 4.11. Esta condición es compatible con clonación de patente, con cambios estéticos no informados (wrap/pintura) o con sesgos de captura (noche, reflejos, balance de blancos). En el prototipo actual la validación se realiza principalmente por patente, lo que es suficiente para un MVP pero deja exposición ante suplantación. Como mejora, se propone incorporar un segundo factor visual basado en estimación de color como atributo y degradar a validación manual cuando la discrepancia sea consistente, preservando equilibrio entre usabilidad y seguridad.



Figura 4.11: Caso borde ilustrativo: misma patente, color distinto al registrado. Este patrón es compatible con suplantación, por lo que se recomienda incorporar atributos adicionales y degradar a revisión manual.

El presente capítulo consolida la transición desde el diseño conceptual hacia un despliegue funcional en terreno, poniendo el foco en la materialización de la infraestructura física y en el flujo extremo a extremo que habilita la operación del sistema. En particular, se documenta cómo la capa de obtención de datos se implementa mediante sensorización por plaza y captura en el acceso, cómo la capa de comunicación desacopla el tramo más adverso mediante LoRa y utiliza conectividad IP como backhaul, y cómo el nodo de borde orquesta la interacción entre percepción, verificación y actuación del portón. Esta implementación se sustenta en el marco revisado en el estado del arte, donde las arquitecturas por capas para smart parking y los enfoques modulares de ALPR se presentan como estrategias para mejorar trazabilidad, mantenibilidad e integración entre componentes heterogéneos.

A nivel de resultados, las pruebas realizadas permiten validar que las decisiones de ingeniería adoptadas cumplen su propósito operativo: las mediciones de ocupación se reflejan en los servicios centrales mediante un camino de comunicación estable, y el acceso vehicular se controla a partir de una cadena de verificación que privilegia seguridad por sobre automatización irrestricta. Al mismo tiempo, el despliegue evidencia limitaciones esperables en sistemas ciberfísicos de este tipo, principalmente asociadas a condiciones reales de captura, dependencia de conectividad y aparición de escenarios no cubiertos por una única señal de identificación. Estas observaciones no solo delimitan el alcance del MVP, sino que también conectan con lo discutido en la literatura respecto de la necesidad de complementar la lectura de patente con más evidencia y de establecer políticas de degradación cuando la confianza no es suficiente.

Capítulo 5

Conclusiones y Perspectivas

5.1. Conclusiones Generales

El trabajo desarrollado en esta memoria abordó la problemática de la gestión de accesos vehiculares y estacionamientos de visita en comunidades residenciales, un ámbito donde todavía predominan procesos manuales, validaciones poco sistemáticas y registros de baja trazabilidad. A partir de este diagnóstico, se propuso ResiPark como una arquitectura ciberfísica orientada a articular percepción, comunicación, validación y actuación dentro de un flujo operativo coherente para entornos residenciales.

La principal contribución de la memoria no reside únicamente en automatizar una apertura de portón o en detectar la ocupación de un estacionamiento, sino en demostrar que ambos procesos pueden integrarse dentro de una misma lógica de operación. En este sentido, el sistema permite conectar eventos del mundo físico con decisiones digitales verificables, asociando accesos vehiculares, autorizaciones, reservas, estados de ocupación y evidencia operativa. Esta integración permite avanzar desde un modelo basado en criterio humano y registros aislados hacia un esquema más trazable, auditable y consistente.

Desde una perspectiva arquitectónica, el trabajo demuestra la utilidad de organizar la solución mediante capas y módulos con responsabilidades diferenciadas. Esta separación permite desacoplar la adquisición de datos, la comunicación, la lógica de validación y las interfaces de usuario, evitando que el sistema dependa de un único componente monolítico. Como consecuencia, cada parte puede evolucionar, ser reemplazada o escalar de forma independiente, manteniendo la coherencia del flujo general. Este principio resulta especialmente relevante para comunidades residenciales, donde la infraestructura disponible suele ser heterogénea y las condiciones de despliegue varían entre edificios.

Otro aporte relevante corresponde a la interoperabilidad entre módulos heterogéneos. La solución integra mecanismos de percepción vehicular, sensado distribuido de ocupación, servicios centrales e interfaces para distintos perfiles de usuario bajo una misma cadena de operación. Esta articulación permite que la información capturada en terreno no quede aislada, sino que sea transformada en eventos útiles para conserjería, residentes y administración. De este modo, el sistema no solo registra datos, sino que los convierte en soporte para decisiones operativas y para la reconstrucción posterior de eventos.

La implementación y validación del prototipo permitieron comprobar la viabilidad de automatizar parcialmente el control de acceso vehicular sin reemplazar por completo la infraestructura existente. Esto es importante porque sitúa la propuesta como una solución incremental: no exige transformar radicalmente el edificio, sino añadir una capa de inteligencia sobre elementos ya presentes en muchas comunidades. Bajo esta lógica, ResiPark busca complementar la operación humana, no eliminarla, entregando información verificable y

mecanismos de alerta que permiten reducir ambigüedad en situaciones normales y degradar hacia revisión manual en casos dudosos.

Los resultados también permiten confirmar la importancia de la trazabilidad como eje de seguridad. En un sistema residencial, la apertura automática no debe entenderse como el único objetivo; igual de importante es saber qué evento ocurrió, bajo qué condiciones, qué información fue utilizada para validarlo y qué acción se ejecutó posteriormente. Esta cadena de evidencia mejora la capacidad de auditoría frente a incidentes, conflictos por uso de estacionamientos o intentos de acceso no autorizados.

Al mismo tiempo, el prototipo evidencia limitaciones propias de un sistema ciberfísico en etapa de validación. Su funcionamiento depende de condiciones de captura adecuadas, conectividad estable, calibración de sensores y manejo correcto de casos borde. Estas restricciones no invalidan la propuesta, sino que delimitan su alcance como MVP y permiten identificar los aspectos que deben fortalecerse para avanzar hacia un despliegue productivo. En particular, se vuelve necesario robustecer la disponibilidad de los servicios, mejorar la tolerancia a fallos, ampliar las pruebas en distintos entornos residenciales e incorporar mecanismos adicionales de verificación cuando una única señal de identificación no sea suficiente.

En conclusión, ResiPark demuestra que es posible diseñar e implementar una arquitectura modular para mejorar la seguridad y la gestión operativa de accesos vehiculares y estacionamientos de visita en comunidades residenciales. La memoria aporta una propuesta integradora que combina automatización, trazabilidad, modularidad e interoperabilidad, mostrando que el valor del sistema no está solo en sus componentes individuales, sino en la forma en que estos se coordinan para producir decisiones verificables sobre eventos reales del entorno.

5.2. Caso de estudio: Inmobiliaria Norte Verde

La colaboración con Inmobiliaria Norte Verde permitió contrastar el diseño de ResiPark con las necesidades de un cliente real que administra edificios en altura con un flujo de vehículos significativamente mayor que el del entorno de pruebas. En las reuniones iniciales se identificaron tres preocupaciones recurrentes: mejorar el control de acceso de residentes, ordenar el uso de los estacionamientos de visita y disponer de registros claros en caso de conflictos entre vecinos o incidentes de seguridad.

En ese contexto, el sistema propuesto dialoga bien con la realidad de Norte Verde por al menos dos motivos. En primer lugar, se apoya en elementos que ya están presentes en sus edificios: portones eléctricos que pueden accionarse por pulso, cámaras de vigilancia ubicadas en los accesos y conectividad de datos en conserjería. Esto evita exigir inversiones en infraestructura pesada y permite pensar en una instalación progresiva, edificio por edificio. En segundo lugar, ResiPark entrega una capa de digitalización que hoy no existe: registro de ingresos vehiculares asociado a patentes, historial de reservas de visitas y visualización en tiempo real de los estacionamientos disponibles.

Sin embargo, el análisis también mostró que el prototipo todavía está lejos de un despliegue inmediato en un edificio de la escala de Norte Verde. Para operar en ese entorno sería necesario robustecer al menos tres aspectos: la disponibilidad del entorno de detección (migrándolo desde un laboratorio de pruebas a una infraestructura de nube con monitoreo y respaldo); la capacidad de soportar múltiples accesos concurrentes y varios puntos de cámara

por edificio; y la integración con los procesos internos de la inmobiliaria (por ejemplo, con sus sistemas de administración y los protocolos de atención de conserjería). En la práctica, la implementación real en un condominio administrado por Norte Verde debiera abordarse como un proyecto de despliegue y operación en sí mismo, que incluya etapas de piloto controlado, capacitación a conserjes y acuerdos claros sobre soporte y mantenimiento.

5.3. Caso de estudio: Pasaje con portón eléctrico

El pasaje residencial utilizado como entorno de pruebas representó el escenario más cercano a la realidad que se pudo alcanzar dentro de los recursos del proyecto. A diferencia de los edificios administrados por una inmobiliaria, este pasaje cuenta con una organización más informal y una infraestructura mínima: un único portón eléctrico para acceso vehicular, un control remoto original del motor y conectividad a internet a través de un router doméstico.

En este contexto, el despliegue de ResiPark tuvo dos objetivos principales: verificar que el flujo de detección y apertura funcionara de extremo a extremo sobre un portón real, y comprobar que el módulo de estacionamientos pudiera operar en un espacio físico no diseñado originalmente para sensado IoT. La instalación de la cámara IP, la caja de componentes con la Raspberry Pi y el relé junto al portón, y el montaje del sensor ultrasónico en un árbol cercano, mostró que es posible adaptar el sistema a las restricciones físicas del lugar sin intervenir el motor ni realizar obras civiles.

Las pruebas realizadas en el pasaje permitieron observar tanto fortalezas como limitaciones. Por un lado, el sistema fue capaz de detectar vehículos en movimiento, leer sus patentes y registrar los intentos de ingreso en la base de datos, respetando la lógica de autorización definida. Por otro, quedaron en evidencia situaciones donde el sistema no podría responder tan bien: horarios nocturnos, presencia de lluvia, vehículos que se detienen demasiado cerca de la cámara o peatones que cruzan delante del auto. En estos casos, la figura del conserje o encargado del portón sigue siendo fundamental como respaldo, lo que refuerza la idea de que la solución debe entenderse como un asistente tecnológico y no como un reemplazo total de la supervisión humana.

Aun con estas limitaciones, la experiencia en el pasaje con portón eléctrico confirmó que el prototipo es instalable en escenarios reales de baja escala y que su impacto se percibe de inmediato: las personas dejan de depender sólo del control remoto y se empieza a construir un historial de accesos y uso de estacionamientos que antes simplemente no existía.

5.4. Perspectivas y trabajo a futuro

El estado actual de ResiPark corresponde a un prototipo funcional que demostró ser útil para validar la idea y explorar sus implicancias técnicas. El siguiente paso lógico es avanzar desde este prototipo de laboratorio hacia una solución más cercana a producto, con énfasis en la robustez, la mantenibilidad y la escalabilidad.

En el corto plazo, una línea de trabajo evidente es estabilizar el entorno de detección y reconocimiento de patentes. Esto implica migrar la ejecución de los modelos desde entornos experimentales como Google Colab hacia servicios en la nube diseñados para operación

continua, con monitoreo, registro de fallas, actualización controlada de modelos y mecanismos de *fallback* en caso de problemas de conectividad. En paralelo, sería deseable incrementar y diversificar el conjunto de datos de entrenamiento y prueba, incorporando más videos en distintas condiciones de iluminación, clima y tipos de vehículos, incluyendo motocicletas y vehículos con patentes sólo traseras.

Otra línea relevante es la mejora de las aplicaciones cliente. La aplicación móvil podría evolucionar hasta cumplir con los requisitos de publicación en tiendas oficiales (Android e iOS), lo que exige revisar dependencias, permisos y mecanismos de actualización. Por el lado de la aplicación web, es posible profundizar en funciones de administración, reportes de uso de estacionamientos y configuración flexible de reglas internas (horarios, tolerancias, listas blancas y negras de patentes, etc.).

En cuanto a la integración con casos de uso reales, un objetivo de mediano plazo es ejecutar un piloto acotado en un condominio administrado por Inmobiliaria Norte Verde, partiendo por un único acceso y un conjunto reducido de estacionamientos de visita. Este piloto debería incluir no sólo la instalación técnica, sino también la definición de protocolos de contingencia, capacitación a conserjes y mecanismos de recolección de retroalimentación por parte de los residentes.

Adicionalmente, como línea de trabajo a futuro orientada a robustez y seguridad, se propone abordar casos borde asociados a manipulación visual de la patente y posible suplantación por atributos del vehículo. En particular, el sistema debería incorporar mecanismos para detectar patentes dibujadas/adulteradas (donde el recuadro puede existir, pero la lectura OCR tiende a bajar su confiabilidad y puede fallar la validación de formato chileno) y para gestionar escenarios de misma patente con vehículo distinto, por ejemplo con color diferente al registrado. Esto implica reforzar el pipeline con validación sintáctica estricta de placa chilena, umbrales adaptativos de confianza, y un segundo factor visual basado en clasificación de atributos (p. ej., color/tipo de vehículo) para activar alertas y degradar a revisión por conserjería cuando existan discrepancias, reduciendo el riesgo de aperturas automáticas ante evidencias ambiguas o inconsistentes.

Finalmente, desde una perspectiva más amplia, ResiPark abre la puerta a explorar temas como la integración con otras plataformas de ciudad inteligente, la interoperabilidad con sistemas de administración de edificios, el uso de fuentes externas de verificación de patentes y el análisis estadístico del uso de estacionamientos para apoyar decisiones de gestión. Profundizar en estas líneas permitiría transformar el prototipo actual en un sistema más maduro, capaz de aportar valor sostenido tanto en condominios pequeños como en proyectos inmobiliarios de mayor envergadura.

Bibliografía

[1] Han Chen, Paul Chou, Sastry Duri, Hui Lei, and Johnathan Reason. The design and implementation of a smart building control system. In 2009 IEEE International Conference on e-Business Engineering, pages 255–262. IEEE, 2009.

[2] Fangyu Li, Hongyan Yang, Xuejin Gao, and Honggui Han. Towards iot-based sustainable digital communities. Intelligent and Converged Networks, 3(2):190–203, 2022.

[3] Qi Liu, Yi Zeng, Xingyao Zhou, Na Liu, and Mengxiang Wang. Design and implementation of smart community information management system. In 2023 International Seminar on Computer Science and Engineering Technology (SCSET), pages 636–639. IEEE, 2023.

[4] Nian Xue, Lulu Liang, Jie Zhang, and Xin Huang. An access control system for intelligent buildings. In Proceedings of the 9th EAI International Conference on Mobile Multimedia Communications (MOBIMEDIA), pages 11–17, 2016.

[5] Centro de Estudios y Análisis del Delito (CEAD). Informes estadísticos casos policiales. `https://cead.minsegpublica.gob.cl/informes-estadisticos-page`, 2026. Sitio institucional consultado en febrero de 2026.

[6] Ministerio de Seguridad Pública de Chile. La familia delictual de robos violentos disminuyó -9,8 % en su tasa en 2025, de acuerdo con las últimas cifras policiales unificadas. `https://minsegpublica.cl/la-familia-delictual-de-robos-violentos-disminuyo-98-en-su-tasa-en-2025-de-acuerdo-con-las-ultimas-cifras-policiales-unificadas/`, 2026. Publicación institucional consultada en febrero de 2026.

[7] Ministerio Público de Chile. Informe crimen organizado en chile. `https://www.fiscaliadechile.cl/sites/default/files/documentos/InformeCrimenOrganizado_%20FINAL.pdf`, 2025. Documento institucional consultado en febrero de 2026.

[8] ComunidadFeliz. Control de acceso vehicular y peatonal: doble seguridad. `https://www.comunidadfeliz.cl/post/control-de-acceso-vehicular-y-peatonal-duplica-la-seguridad-de-tu-condominio`, 2025. Consulta: 8 de marzo de 2026.

[9] Citofon. Detección de patentes - control de acceso, 2025. Sitio web consultado para describir funcionalidades de detección de patentes y control de acceso en Chile.

[10] S. V. Srikanth, P. J. Pramod, K. P. Dileep, S. Tapas, Mahesh U. Patil, and Sarat Chandra 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, pages 568–573, 2009.

[11] Hind Kadim Alaoui, Moussa Coulibaly, and Ahmed Errami. Analyzing smart parking system: Wsn technologies, power saving and guidance. In 2024 International Conference on Circuit, Systems and Communication (ICCSC), pages 1–6, 2024.

[12] Yi-Yun Chu and Kai-Hao Liu. Iot in vehicle presence detection of smart parking system. In 2nd IEEE Eurasia Conference on IOT, Communication and Engineering 2020, pages 56–61, 2020.

[13] Citofon. Detección de patentes, 2025. Descripción comercial del módulo LPR y registro de ingresos/salidas.

[14] Inmobiliaria Norte Verde. Norte verde. `https://norte-verde.cl/`, 2026. Sitio web consultado en febrero de 2026.

[15] Federal Access. Barrera de estacionamiento para tu empresa de san bernardo. `https://federalaccess.cl/blogs/noticias/barrera-de-estacionamiento-para-tu-empresa-de-san-bernardo`, 2025. Rangos de precios estimados (2025), consultado el 25 de febrero de 2026.

[16] Banco Central de Chile. Indicadores diarios: Unidad de fomento (uf) (pesos), año 2026. `https://si3.bcentral.cl/indicadoressiete/secure/Serie.aspx?gcode=UF`, 2026. UF del 25-02-2026 reportada como $39.773,62; consultado el 25 de febrero de 2026.

[17] Access Chile. Barreras automáticas. `https://accesschile.cl/barreras-automaticas/`, 2025. Descripción de suministro/instalación y ecosistema de control, consultado el 25 de febrero de 2026.

[18] Usharani Hareesh Govindarajan, Amy J. C. Trappey, and Charles V. Trappey. Immersive technology for human-centric cyberphysical systems in complex manufacturing processes: A comprehensive overview of the global patent profile using collective intelligence. Complexity, 2018:4283634, 2018.

[19] Swati Swati, S. D. Kawa, S. Kamble, D. Desai, P. H. Karelia, and P. Engineer. An efficient deep learning based license plate recognition for smart cities. Electronic Letters on Computer Vision and Image Analysis, 23(2), 2024. Artículo consultado para contextualizar reconocimiento de patentes basado en aprendizaje profundo.

[20] Zied Selmi, Mohamed Ben Halima, and Adel M. Alimi. Deep learning system for automatic license plate detection and recognition. In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), volume 1, pages 1132–1137. IEEE, 2017.

[21] Rayson Laroca, Evair Severo, Luiz A. Zanlorensi, Luiz S. Oliveira, Gabriel R. Goncalves, William Robson Schwartz, and David Menotti. A robust real-time automatic license plate recognition based on the yolo detector. In 2018 International Joint Conference on Neural Networks (IJCNN), 2018.

[22] ComunidadFeliz. Comunidadfeliz - software para la administración de edificios y condominios, 2025. Plataforma chilena para administración de comunidades, gastos comunes y módulos asociados.

[23] EdiPro. Edipro - software para la gestión de edificios y condominios, 2025. Plataforma chilena de administración de comunidades con módulos administrativos y operacionales.

[24] Kastor. Kastor - ecosistema de soluciones para la administración de comunidades, 2025. Plataforma chilena orientada a gestión de comunidades y operación administrativa.

[25] Federal Access. Barrera de estacionamiento con lector de patentes integrado, 2025. Artículo descriptivo del flujo LPR para validación y apertura automática de barrera.

[26] Access Chile. Control de proximidad, tag, patentes, 2025. Página comercial sobre soluciones de control de acceso y LPR en Chile.

[27] Subsecretaría de Prevención del Delito. La familia delictual de robos violentos disminuyó -9,8 % en su tasa en 2025, de acuerdo con las últimas cifras policiales unificadas, 2026. Publicación institucional.

[28] Emol. Maipú, puente alto y san bernardo las más afectadas: Fiscalía revela cifras de robos violentos de vehículos, 2026. Nota periodística basada en reporte de Fiscalía.

[29] Smartki. Smartki - control de acceso, 2025. Solución chilena de control de acceso para recintos, con gestión desde smartphone.

[30] Z. Liu et al. Comparative evaluation of the performance of zigbee and lora wireless networks in building environment. Electronics, 11(21):3560, 2022.

[31] LoRa Alliance. What is LoRaWAN Specification. `https://lora-alliance.org/about-lorawan-old/`, 2025. Consulta: 8 de marzo de 2026.

[32] Connectivity Standards Alliance. Zigbee: The Full-Stack Solution for All Smart Devices. `https://csa-iot.org/all-solutions/zigbee/`, 2026. Consulta: 8 de marzo de 2026.

[33] LoRa Alliance. Wi-Fi & LoRaWAN Deployment Synergies. `https://lora-alliance.org/wp-content/uploads/2020/11/wi-fi-and-lorawanr-deployment-synergies.pdf`, 2019. Consulta: 8 de marzo de 2026.

[34] Can Biyik, Zaheer Allam, Gabriele Pieri, Davide Moroni, Muftah O'Fraifer, Eoin O'Connell, Stephan Olariu, and Muhammad Khalid. Smart parking systems: Reviewing the literature, architecture and ways forward. Smart Cities, 4(2):623–642, 2021.

[35] Muhammad Zainal Abidin and Reza Pulungan. A systematic review of machine-vision-based smart parking systems. Scientific Journal of Informatics, 7(2):213–228, 2020.

[36] Microsoft Research. Florence-2: A novel vision foundation model. `https://huggingface.co/microsoft/Florence-2-large`, 2025. Accedido en noviembre de 2025.

[37] Natalya Kharina and Sergei 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.

[38] Zavier Ferodova Al Fitroh and Gunawan Ariyanto. Implementation of yolov11 for vehicle license plate detection and recognition. In Proceedings of the 2025 International Conference on Smart Computing, IoT and Machine Learning (SIML). IEEE, 2025.

[39] Madhur Dixit, Srimathi C, Robin Doss, Seng Loke, and M.A. Saleemdurai. Smart parking with computer vision and iot technology. In 2020 International Conference on Telecommunications and Signal Processing (TSP), pages 170–174. IEEE, 2020.

[40] G. Callebaut, G. Ottoy, and L. Van der Perre. Cross-layer framework and optimization for efficient use of the energy budget of iot nodes. In 2019 IEEE Wireless Communications and Networking Conference (WCNC), pages 1–6, 2019.

[41] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2024.

[42] Jiuhai Chen, Jianwei Yang, Haiping Wu, Dianqi Li, Jianfeng Gao, Tianyi Zhou, and Bin Xiao. Florence-vl: Enhancing vision-language models with generative vision encoder and depth-breadth fusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2025.

[43] J. B. Kim et al. Deep learning-based vehicle type and color classification for real-time road applications. Applied Sciences, 14(4):1600, 2024.

[44] Gabriel E. Lima, Rayson Laroca, Eduardo Santos, Eduil Nascimento Jr., and David Menotti. Toward enhancing vehicle color recognition in adverse conditions: A dataset and benchmark, 2024.

[45] JDAI-CV. Veri-776: Vehicle re-identification dataset (with attributes such as color, type, brand). GitHub repository, 2016.

[46] Python Software Foundation. Python language reference. `https://docs.python.org/3/`, 2025. Accedido en noviembre de 2025.

[47] Meta Platforms, Inc. React native documentation. `https://reactnative.dev/`, 2025. Accedido en noviembre de 2025.

[48] Expo. Expo documentation. `https://docs.expo.dev/`, 2025. Accedido en noviembre de 2025.

[49] OpenCV Team. Opencv (open source computer vision library). `https://opencv.org/`, 2025. Accedido en noviembre de 2025.

[50] Docker, Inc. Docker documentation. `https://docs.docker.com/`, 2025. Accedido en noviembre de 2025.

[51] Sebastián Ramírez. Fastapi - high performance, easy to learn, fast to code, ready for production. https://fastapi.tiangolo.com/, 2025. Accedido en noviembre de 2025.

[52] Encode. Uvicorn: The lightning-fast asgi server. https://www.uvicorn.org/, 2025. Accedido en noviembre de 2025.

[53] SQLAlchemy Project. Sqlalchemy - the python sql toolkit and object relational mapper. https://www.sqlalchemy.org/, 2025. Accedido en noviembre de 2025.

[54] BitPusher, LLC. Passlib: Password hashing library. https://passlib.readthedocs.io/, 2025. Accedido en noviembre de 2025.

[55] MagicStack Inc. and contributors. asyncpg - a fast postgresql database client library for python/asyncio. https://magicstack.github.io/asyncpg/current/, 2025. Accedido en noviembre de 2025.

[56] Saurabh Kumar and contributors. python-dotenv: Read key-value pairs from .env files. https://github.com/theskumar/python-dotenv, 2025. Accedido en noviembre de 2025.

[57] Michael Davis and contributors. Javascript object signing and encryption (jose) for python. https://github.com/mpdavis/python-jose, 2025. Accedido en noviembre de 2025.

[58] Arduino Team. Arduino official website. https://www.arduino.cc/, 2025. Accedido en noviembre de 2025.

[59] Raspberry Pi Foundation. Raspberry pi documentation. https://www.raspberrypi.com/documentation/, 2025. Accedido en noviembre de 2025.

[60] Prusa Research. Printables.com | 3d models database. https://www.printables.com/, 2025. Accedido en noviembre de 2025.

[61] All Solutions. Barreras de acceso vehicular. https://www.allsolutions.cl/automatizacion/barreras, 2026. Listado de precios web con IVA, consultado el 25 de febrero de 2026.

[62] Electronia. Barreras parking y control acceso. https://electronia.cl/categoria/barreras-parking/, 2026. Listado de productos con precios (incluye cámara lectora de patentes), consultado el 25 de febrero de 2026.

[63] Avotech Chile. Colección: Cámara lpr-anpr. https://avotech.cl/collections/camara-lpr-anpr, 2026. Catálogo con precios publicados, consultado el 25 de febrero de 2026.

[64] PCPlay Chile. Cámara bullet ip 3mp lpr con reconocimiento de patente de vehículo inteligente pkc2630@z28-p (uniview). https://www.pcplay.cl/producto/camara-bullet-ip-3mp-lpr-con-reconocimiento-de-patente-de-vehiculo-inteligente-pkc2630-at-z28-p-uniview, 2026. Precio publicado con IVA (oferta internet), consultado el 25 de febrero de 2026.

[65] LoRa Alliance. LoRaWAN Security Whitepaper. https://resources.lora-alliance.org/security/lorawan-security-whitepaper, 2025. Consulta: 8 de marzo de 2026.

[66] ComunidadFeliz. Residentes: toda tu comunidad en una sola app. https://www.comunidadfeliz.cl/residentes, 2026. Consulta: 8 de marzo de 2026.

[67] Kastor. Kastor: ecosistema de soluciones para la administración de comunidades. https://www.kastor.cl/, 2026. Consulta: 8 de marzo de 2026.

[68] EdiPro. Términos y condiciones de la plataforma EdiPro. https://www.edipro.cl/tos/, 2026. Consulta: 8 de marzo de 2026.

[69] Access Chile. Control Vehicular. https://accesschile.cl/control-vehicular/, 2025. Consulta: 8 de marzo de 2026.

[70] Federal Access. Control de Acceso Profesional. https://federalaccess.cl/, 2026. Consulta: 8 de marzo de 2026.

[71] Gilles Callebaut, Geoffrey Ottoy, and Liesbet Van der Perre. Loraenergysim. https://github.com/GillesC/LoRaEnergySim. GitHub repository, consultado en abril de 2026.

[72] NiceRF. 915mhz sx1276 front end lora module - lora1276, 2026. Consultado en 2026.

[73] The Things Network. Spreading factors, 2026. Consultado en 2026.