

**UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
SEDE VIÑA DEL MAR – JOSÉ MIGUEL CARRERA**

**PLATAFORMA PARA LA GESTIÓN Y CONTROL DE ESTACIONAMIENTOS
EN EDIFICIOS CORPORATIVOS**

Trabajo de titulación para optar al Título
Profesional de Ingeniero de Ejecución en
Software

Estudiante:

Eduardo Díaz Vásquez

Profesor Guía:

Pedro Godoy Barrera

2025



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 PARA LA GESTIÓN Y CONTROL DE ESTACIONAMIENTOS EN EDIFICIOS CORPORATIVOS

Nombre del candidato(a): EDUARDO DÍAZ VÁSQUEZ

Carrera / Grado: INGENIERÍA DE EJECUCIÓN EN SOFTWARE

Campus: SEDE VIÑA DEL MAR Departamento: DEPARTAMENTO DE ELECTROTECNIA E INFORMÁTICA

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, PEDRO FRANCISCO GODOY BARRERA, 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 Marzo 2026

Firma:

Estudiante o Candidato(a):

Fecha: 25 de Marzo 2026

Firma:

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

AGRADECIMIENTOS

Agradezco a todas y todos quienes han sido parte de este proceso de formación. En especial, a mi esposa Clau y a mi hija Sophy, quienes me acompañaron y brindaron su apoyo constante, permitiéndome culminar exitosamente este camino que contribuyó de manera significativa a mi crecimiento personal y profesional.

RESUMEN

KEYWORDS: GESTIÓN DE ESTACIONAMIENTOS, CONTROL DE ACCESOS, EDIFICIOS CORPORATIVOS.

El presente documento tiene como objetivo principal explicar en detalle el proceso de análisis y diseño de una solución informática destinada a optimizar la gestión de estacionamientos en edificios comerciales administrados por la empresa Arikok.

Arikok es una organización chilena especializada en la administración integral de edificios comerciales, con sede en Santiago y operaciones a nivel nacional. Su mercado abarca la gestión de propiedades comerciales, asegurando la valorización y un mantenimiento óptimo de los inmuebles.

El problema por solucionar es la gestión manual y poco eficiente de los estacionamientos en los edificios, lo que genera demoras en los accesos, es propenso a errores humanos, no cuenta con un control automatizado y la gestión del uso de los espacios es poco eficaz. Esto afecta tanto la experiencia de las empresas inquilinas como la eficiencia operativa.

El trabajo de título se estructura en tres capítulos fundamentales, seguidos de las conclusiones. El capítulo 1, "Aspectos Relevantes del Proyecto y su Gestión," introduce los antecedentes de la empresa Arikok, describe la situación actual de la gestión de estacionamientos y los problemas detectados. Además, detalla los requerimientos del usuario, los objetivos del proyecto, la evaluación y selección de alternativas de solución, y concluye con la planificación completa del proyecto, incluyendo la asignación de recursos, la administración de riesgos y la estimación de costos.

Posteriormente, el capítulo 2, "Aspectos Relevantes del Análisis," profundiza en la solución propuesta, presentando una descripción detallada de la misma, los requerimientos funcionales y no funcionales del sistema, y las condicionantes de diseño tecnológico. Este capítulo también abarca la estructura funcional del sistema mediante un caso de uso general, el modelo conceptual, los modelos de casos de uso específicos y la matriz de trazabilidad.

El capítulo 3, "Aspectos Relevantes del Diseño," se centra en la materialización de la solución. Aquí se define la arquitectura del software, incluyendo las características del recurso computacional y la arquitectura detallada del sistema. Se presenta el diseño de datos a través del modelo de clases, el modelo relacional y el diccionario de datos. Asimismo, se incluyen diagramas de secuencia extendidos para los casos de uso clave, el diseño de componentes y el diseño de la interfaz de usuario, incluyendo diagramas de navegación y prototipos de pantallas.

Finalmente, el documento cierra con las conclusiones, donde se sintetiza la experiencia adquirida y los principales resultados obtenidos a lo largo del desarrollo de este trabajo de título.

ÍNDICE

INTRODUCCIÓN	1
1 ASPECTOS RELEVANTES DEL PROYECTO Y SU GESTIÓN.....	3
1.1 Descripción de la Organización.....	3
1.1.1 Visión.....	3
1.1.2 Misión.....	3
1.1.3 Objetivos.....	4
1.1.4 Servicios	4
1.1.5 Departamentos afectados por el sistema a desarrollar.....	4
1.2 Objetivos y beneficios del proyecto	5
1.3 Descripción de la Situación sin Proyecto	6
1.4 Problemas detectados	7
1.5 Requerimientos y Requisitos del Usuario	9
1.6 Alternativas globales de solución	10
1.6.1 Alternativa 1: Desarrollo a medida.....	10
1.6.2 Alternativa 2: Software como Servicio Kalena	11
1.6.3 Alternativa 3: Software como Servicio Click Visitas.....	12
1.7 Evaluación de Alternativas.....	13
1.7.1 Criterios para evaluar las alternativas.....	13
1.7.2 Ponderación de Criterios y Escala de Evaluación	13
1.7.3 Evaluación	14
1.8 Alternativa seleccionada.....	16
1.8.1 Descripción.....	16
1.8.2 Beneficios	17
1.9 Planificación	18
1.9.1 Desarrollo del plan de personal.	18
1.9.2 Planificación temporal, y asignación de recursos (Carta Gantt)	20
1.10 Administración del riesgo.....	25
1.11 Estimación de costos	28
2 ASPECTOS RELEVANTES DEL ANÁLISIS.....	33
2.1 Descripción de la solución propuesta	33
2.2 Diagrama de actividades general UML (flujo administrativo).....	35

2.2.1	Asignación de estacionamientos a empresas inquilinas	36
2.2.2	Solicitud de estacionamientos	37
2.2.3	Solicitud de estacionamientos de visitas	38
2.2.4	Gestión de incidentes.....	39
2.3	Requerimientos del sistema (funcionales y no funcionales)	40
2.3.1	Requerimientos funcionales	40
2.3.2	Requerimientos no-funcionales	42
2.4	Estructura funcional del sistema (caso de uso general).....	43
2.5	Matriz de trazabilidad.....	46
2.6	Modelo conceptual	47
2.7	Modelo de casos de uso	48
2.7.1	Caso de uso narrativo CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina.....	48
2.7.2	Diagrama de secuencia CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina.....	52
2.7.3	Contratos CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina	52
2.7.4	Caso de uso narrativo CU-10 Controlar y registrar entradas/salidas de vehículos	53
2.7.5	Diagrama de secuencia CU-10 Controlar y registrar entradas/salidas de vehículos	55
2.7.6	Contratos CU-10 Controlar y registrar entradas/salidas de vehículos.....	56
2.7.7	Caso de uso narrativo CU-14 Gestionar reservas de estacionamiento de empresa inquilina	56
2.7.8	Diagrama de secuencia CU-14 Gestionar reservas de estacionamiento de empresa inquilina	60
2.7.9	Contratos CU-14 Gestionar reservas de estacionamiento de empresa inquilina ..	60
2.7.10	Caso de uso narrativo CU-15 Gestionar mis reservas de estacionamiento	61
2.7.11	Diagrama de secuencia CU-15 Gestionar mis reservas de estacionamiento	65
2.7.12	Contratos CU-15 Gestionar mis reservas de estacionamiento.....	66
2.7.13	Caso de uso narrativo CU-16 Gestionar mis solicitudes de estacionamiento de visitas	66
2.7.14	Diagrama de secuencia CU-16 Gestionar mis solicitudes de estacionamiento de visita	70
2.7.15	Contratos CU-16 Gestionar mis solicitudes de estacionamiento de visita	70

2.7.16	Caso de uso narrativo CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita	71
2.7.17	Diagrama de secuencia CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita	73
2.7.18	Contratos CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita	74
2.7.19	Caso de uso narrativo CU-19 Gestionar mis reportes de incidentes	74
2.7.20	Diagrama de secuencia CU-19 Gestionar mis reportes de incidentes	77
2.7.21	Contratos CU-19 Gestionar mis reportes de incidentes.....	77
2.7.22	Caso de uso narrativo CU-20 Gestionar incidentes operativos reportados	78
2.7.23	Diagrama de secuencia CU-20 Gestionar incidentes operativos reportados	80
2.7.24	Contratos CU-20 Gestionar incidentes operativos reportados	81
3	ASPECTOS RELEVANTES DEL DISEÑO	83
3.1	Arquitectura del software	83
3.1.1	Arquitectura lógica del sistema	85
3.1.2	Arquitectura de despliegue del sistema	87
3.1.3	Tecnologías seleccionadas.....	88
3.2	Diseño de Datos.....	89
3.2.1	Modelo de Clases.....	89
3.2.2	Modelo Relacional.....	91
3.2.3	Diccionario de Datos	93
3.3	Diagramas de secuencia extendido.....	105
3.3.1	Diagrama de secuencia extendido CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina	106
3.3.2	Diagrama de secuencia extendido CU-10 Controlar y registrar entradas/salidas de vehículos	107
3.3.3	Diagrama de secuencia extendido CU-14 Gestionar reservas de estacionamiento de empresa inquilina.....	108
	Diagrama de secuencia extendido CU-15 Gestionar mis reservas de estacionamiento	109
3.3.4	Diagrama de secuencia extendido CU-16 Gestionar mis solicitudes de estacionamiento de visitas	110
3.3.5	Diagrama de secuencia extendido CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita.....	111
3.3.6	Diagrama de secuencia extendido CU-19 Gestionar mis reportes de incidentes	112
3.3.7	Diagrama de secuencia extendido CU-20 Gestionar incidentes operativos reportados	113

3.4	Diseño de Interfaz.....	114
3.4.1	Diagrama de Navegación.....	114
3.4.2	Filosofía del diseño.....	116
CONCLUSIONES		123
BIBLIOGRAFÍA		125
ANEXOS.....		126

ÍNDICE DE TABLAS

Tabla 1-1. Criterios de evaluación.....	13
Tabla 1-2. Escala de evaluación	14
Tabla 1-3. Evaluación Alternativa 1: Desarrollo a medida	14
Tabla 1-4. Evaluación Alternativa 2: Software como servicio Kalena	15
Tabla 1-5. Evaluación Alternativa 3: Software como Servicio Click Visitas	15
Tabla 1-6. Evaluación de alternativas: Resumen.....	16
Tabla 1-7. Desarrollo del plan de personal: Recursos	20
Tabla 1-8. Carta Gantt del proyecto	20
Tabla 1-9. Administración de Riesgos	25
Tabla 1-10. Plan de mitigación.....	26
Tabla 1-11. Estructura de rentas mensuales	28
Tabla 1-12. Estructura de costos fijos	28
Tabla 1-13. Estructura de gastos de administración	29
Tabla 1-14. Matriz de esfuerzo por funcionalidad	29
Tabla 1-15. Costo del proyecto.....	31
Tabla 2-1. Requerimientos funcionales.....	40
Tabla 2-2. Requerimientos no funcionales	42
Tabla 2-3. Matriz de trazabilidad	46
Tabla 2-4. Caso de uso narrativo: CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina.....	48
Tabla 2-5. Contrato: Crear usuario de la empresa inquilina y sus métodos de acceso en el sistema	52
Tabla 2-6. Caso de uso narrativo: CU-10 Controlar y registrar entradas/salidas de vehículos	53
Tabla 2-7. Contrato: Registrar acceso al estacionamiento	56

Tabla 2-8. Caso de uso narrativo: CU-14 Gestionar reservas de estacionamiento de empresa inquilina.....	56
Tabla 2-9. Contrato: Crear reserva de estacionamiento en el sistema.....	60
Tabla 2-10. Caso de uso narrativo: CU-15 Gestionar mis reservas de estacionamiento.....	61
Tabla 2-11. Contrato: Crear reserva de estacionamiento a demanda en el sistema.....	66
Tabla 2-12. Caso de uso narrativo: CU-16 Gestionar mis solicitudes de estacionamiento de visitas.....	66
Tabla 2-13. Contrato: Solicitar estacionamiento de visita en el sistema.....	70
Tabla 2-14. Caso de uso narrativo: CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita.....	71
Tabla 2-15. Contrato: Aprobar/Rechazar solicitud de estacionamiento en el sistema.....	74
Tabla 2-16. Caso de uso narrativo: CU-19 Gestionar mis reportes de incidentes.....	74
Tabla 2-17. Contrato: Reportar incidente en el sistema.....	77
Tabla 2-18. Caso de uso narrativo: CU-20 Gestionar incidentes operativos reportados.....	78
Tabla 2-19. Contrato: Actualizar el estado de un incidente reportado en el sistema.....	81
Tabla 3-1. Arquitectura Hexagonal - tipos de clases.....	86
Tabla 3-2. Diccionario de datos: Asignación contractual.....	93
Tabla 3-3. Diccionario de datos: Asignación contractual – Plaza de estacionamiento.....	94
Tabla 3-4. Diccionario de datos: Asignación - Rol.....	94
Tabla 3-5. Diccionario de datos: Comentario de incidente.....	95
Tabla 3-6. Diccionario de datos: Controlador de acceso.....	95
Tabla 3-7. Diccionario de datos: Edificio.....	96
Tabla 3-8. Diccionario de datos: Empresa inquilina.....	96
Tabla 3-9. Diccionario de datos: Esquema de cobro.....	97
Tabla 3-10. Diccionario de datos: Evidencia de incidente.....	97
Tabla 3-11. Diccionario de datos: Incidente.....	98
Tabla 3-12. Diccionario de datos: Método de acceso.....	98
Tabla 3-13. Diccionario de datos: Plaza de estacionamiento.....	99
Tabla 3-14. Diccionario de datos: Registro de acceso.....	100
Tabla 3-15. Diccionario de datos: Registro de auditoría.....	101
Tabla 3-16. Diccionario de datos: Regla de reserva.....	101
Tabla 3-17. Diccionario de datos: Reserva de estacionamiento.....	102
Tabla 3-18. Diccionario de datos: Rol.....	102
Tabla 3-19. Diccionario de datos: Solicitud de estacionamiento de visita.....	103
Tabla 3-20. Diccionario de datos: Token de restablecimiento de clave.....	104
Tabla 3-21. Diccionario de datos: Usuario.....	104

Tabla 3-22. Diccionario de datos: Vehículo	105
--	-----

ÍNDICE DE FIGURAS

Figura 1-1. Organigrama empresa Arikok.....	5
Figura 2-1. Diagrama de actividad: Asignación de estacionamientos a empresas inquilinas	36
Figura 2-2. Diagrama de actividad: Solicitud de estacionamientos	37
Figura 2-3. Diagrama de actividad: Solicitud de estacionamiento de visitas	38
Figura 2-4. Diagrama de actividad: Gestión de incidentes.....	39
Figura 2-5. Diagrama de casos de uso general	45
Figura 2-6. Modelo conceptual.....	47
Figura 2-7. Diagrama de secuencia: CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina.....	52
Figura 2-8. Diagrama de secuencia: CU-10 Controlar y registrar entradas/salidas de vehículos	55
Figura 2-9. Diagrama de secuencia: CU-14 Gestionar reservas de estacionamiento de empresa inquilina.....	60
Figura 2-10. Diagrama de secuencia: CU-15 Gestionar mis reservas de estacionamiento ..	65
Figura 2-11. Diagrama de secuencia: CU-16 Gestionar mis solicitudes de estacionamiento de visita.....	70
Figura 2-12. Diagrama de secuencia: CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita.....	73
Figura 2-13. Diagrama de secuencia: CU-19 Gestionar mis reportes de incidentes	77
Figura 2-14. Diagrama de secuencia: CU-20 Gestionar incidentes operativos reportados ..	80
Figura 3-1. Diagrama de arquitectura del sistema – vista general.....	83
Figura 3-2. Diagrama de arquitectura del sistema - vista lógica	85
Figura 3-3. Diagrama de arquitectura del sistema - vista de despliegue	87
Figura 3-4. Diagrama de clases	90
Figura 3-5. Diagrama de Entidad-Relación	92
Figura 3-6. Diagrama de secuencia extendido: CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina	106
Figura 3-7. Diagrama de secuencia extendido: CU-10 Controlar y registrar entradas/salidas de vehículos	107

Figura 3-8. Diagrama de secuencia extendido: CU-14 Gestionar reservas de estacionamiento de empresa inquilina.....	108
Figura 3-9. Diagrama de secuencia extendido: CU-15 Gestionar mis reservas de estacionamiento	109
Figura 3-10. Diagrama de secuencia extendido: CU-16 Gestionar mis solicitudes de estacionamiento de visitas	110
Figura 3-11. Diagrama de secuencia extendido CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita.....	111
Figura 3-12. Diagrama de secuencia extendido: CU-19 Gestionar mis reportes de incidentes	112
Figura 3-13. Diagrama de secuencia extendido: CU-20 Gestionar incidentes operativos reportados	113
Figura 3-14. Mapa del sitio.....	115
Figura 3-15. Wireframe: Patrón Asistente.....	116
Figura 3-16. Wireframe: Patrón Retroalimentación de Acción.....	117
Figura 3-17. Wireframe: Patrón Confirmación Explícita.....	118
Figura 3-18. Wireframe: Patrón Tabla de Datos Interactiva	118
Figura 3-19. Wireframe: Patrón Acción Principal Destacada	119
Figura 3-20. Wireframe: Patrón Navegación por Migas de Pan	119
Figura 3-21. Wireframe: Patrón Etiquetas Claras y Alineadas	120
Figura 3-22. Wireframe: Patrón Valores por Defecto Inteligentes	121
Figura 3-23. Wireframe: Patrón Validación en Línea y Mensajes de Error Constructivos	122

SIGLAS Y SÍMBOLOS

A continuación, se presentan las siglas y símbolos utilizados en el presente informe.

SIGLAS:

UML	:	Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language).
CU	:	Caso de Uso.
SaaS	:	Software como Servicio (SaaS, por sus siglas en inglés, Software as a Service).
OpenUP	:	Proceso Unificado Abierto (OpenUP, por sus siglas en inglés, Open Unified Process).
QA	:	Aseguramiento de la Calidad (QA, por sus siglas en inglés, Quality Assurance).
UX	:	Experiencia de Usuario (UX, por sus siglas en inglés, User Experience).
UF	:	Unidad de Fomento.
API	:	Interfaz de Programación de Aplicaciones (API, por sus siglas en inglés, Application Programming Interface).
GUI	:	Interfaz Gráfica de Usuario (GUI, por sus siglas en inglés, Graphical User Interface).
DTO	:	Objeto de Transferencia de Datos (DTO, por sus siglas en inglés, Data Transfer Object).
HTTP	:	Protocolo de Transferencia de Hipertexto (HTTP, por sus siglas en inglés, Hypertext Transfer Protocol).
HTTPS	:	Protocolo Seguro de Transferencia de Hipertexto (HTTPS, por sus siglas en inglés, Hypertext Transfer Protocol Secure).
JDBC	:	Conectividad de Bases de Datos de Java (JDBC, por sus siglas en inglés, Java Database Connectivity).
JPA	:	API de Persistencia de Java (JPA, por sus siglas en inglés, Java Persistence API).
JSON	:	Notación de Objetos de JavaScript (JSON, por sus siglas en inglés, JavaScript Object Notation).
LPR	:	Reconocimiento de Patentes (LPR, por sus siglas en inglés, License Plate Recognition).
RFID	:	Identificación por Radiofrecuencia (RFID, por sus siglas en inglés, Radio-Frequency Identification).
RUN	:	Rol Único Nacional.
SMTP	:	Protocolo para Transferencia Simple de Correo (SMTP, por sus siglas en inglés, Simple Mail Transfer Protocol).
SPA	:	Aplicación de Página Única (SPA, por sus siglas en inglés, Single Page Application).

- CRUD** : Crear, Leer, Actualizar y Eliminar (CRUD, por sus siglas en inglés, Create, Read, Update, Delete).
- ADR** : Architecture Decision Record (ADR, por sus siglas en inglés, Architecture Decision Record) corresponde a un registro formal utilizado para documentar decisiones de arquitectura y el razonamiento que las sustenta.

INTRODUCCIÓN

La gestión de edificios comerciales destinados al arriendo de oficinas es un proceso complejo que busca garantizar el funcionamiento eficiente del inmueble y maximizar su valor en el tiempo. Esta labor abarca desde el mantenimiento de las instalaciones hasta la relación con las empresas inquilinas, procurando ofrecer un entorno seguro, confortable y funcional.

Dentro de este ámbito, la administración de estacionamientos es un aspecto crítico, pues impacta en la experiencia de los usuarios, la eficiencia operativa y la percepción del valor de la propiedad. Una gestión inadecuada de estos espacios puede derivar en problemas como congestión vehicular, demoras en los accesos, mayores costos operativos y una imagen de desorganización, lo que afecta la satisfacción de los usuarios y la capacidad del edificio para retener a sus arrendatarios.

El uso de tecnologías de automatización ofrece alternativas para enfrentar estas dificultades. Sistemas de control de acceso, plataformas digitales de asignación y reserva de espacios, junto con el monitoreo en tiempo real de la disponibilidad, permiten optimizar el uso de los estacionamientos, mejorar la seguridad y facilitar la implementación de políticas de uso más flexibles, contribuyendo a una operación más eficiente y una experiencia organizada para todos los involucrados.

En este contexto, el presente trabajo se centra en la empresa Arikok, dedicada a la administración integral de edificios comerciales, la cual enfrenta el desafío de gestionar de manera manual y poco eficiente los estacionamientos en sus inmuebles. Esta situación genera demoras en los accesos, es propensa a errores humanos y carece de un control automatizado, afectando tanto la experiencia de los usuarios como la eficiencia operativa.

El propósito principal del documento es detallar el proceso de análisis y diseño de una solución informática orientada a optimizar la gestión de estacionamientos. El alcance comprende desde el análisis del contexto de negocio y la definición de la problemática, hasta la especificación de los requerimientos funcionales y no funcionales que debe cumplir el sistema. Además, se incluye el modelado conceptual de la solución, con sus casos de uso, y el diseño técnico, abarcando la arquitectura de software, la estructura de datos y la interfaz de usuario, como base para su futura implementación.

Para presentar este proceso de manera clara, el trabajo se organiza en tres capítulos. El Capítulo 1 describe la empresa Arikok, la situación actual, los objetivos y requerimientos del proyecto, además de la planificación, gestión de riesgos y costos. El Capítulo 2 expone la solución propuesta, detallando los requerimientos, el modelo conceptual y los casos de uso. Finalmente, el Capítulo 3 aborda el diseño de la solución, incluyendo la arquitectura de software, el modelado de datos, los diagramas de secuencia y la interfaz de usuario.

CAPÍTULO 1. ASPECTOS RELEVANTES DEL PROYECTO Y SU GESTIÓN

1 ASPECTOS RELEVANTES DEL PROYECTO Y SU GESTIÓN

Este capítulo se enfoca en describir la situación actual y el proceso que se busca optimizar, proporcionando un marco claro para entender el problema que se desea resolver. A lo largo del capítulo, se identificarán las principales deficiencias observadas en el estado actual, y se explorarán tres alternativas de solución. Cada una de estas alternativas será evaluada en función de criterios específicos que permitirán determinar su factibilidad. Finalmente, se seleccionará la opción más viable, respaldando la decisión mediante un análisis comparativo entre las propuestas.

1.1 Descripción de la Organización

Arikok es una empresa chilena con más de 2 décadas de trayectoria, especializada en la administración integral de edificios comerciales destinados al arriendo de oficinas. Fue fundada en el año 2000, la empresa tiene su casa matriz en Santiago de Chile, desde donde gestiona y supervisa sus operaciones a nivel nacional, con un enfoque en la optimización y valorización de las propiedades comerciales.

Arikok se dedica al rubro de la administración integral de edificios comerciales para oficinas, enfocándose en garantizar una operación eficiente, un mantenimiento preventivo y correctivo de las instalaciones, y una sólida gestión de las relaciones con las empresas inquilinas. Su propósito principal es maximizar la valorización a largo plazo de las propiedades que administra, asegurando el funcionamiento continuo y la satisfacción tanto de los propietarios como de las empresas inquilinas.

1.1.1 Visión

La visión de Arikok es “convertirse en el líder en la administración de edificios comerciales para oficinas, ofreciendo servicios de alta calidad que aseguren la valorización continua de las propiedades, así como la satisfacción a largo plazo tanto de los propietarios como de las empresas inquilinas”.

1.1.2 Misión

La misión de Arikok es “garantizar la operación óptima de los edificios comerciales para oficinas a través de una gestión integral que abarque desde el mantenimiento y operación de las instalaciones hasta la relación con las empresas inquilinas. La empresa busca proporcionar un entorno seguro, eficiente y adecuado para maximizar la satisfacción de todos los usuarios de sus edificios”.

1.1.3 Objetivos

Los objetivos estratégicos de la empresa son:

- Proveer de una operación eficiente de alto nivel en los distintos edificios que administra.
- Optimizar los costos operativos para maximizar la rentabilidad para los propietarios.
- Proveer soluciones de mantenimiento preventivo y correctivo que aseguren la continuidad operativa de todos los sistemas dentro del edificio.
- Fortalecer las relaciones tanto con los propietarios como con las empresas inquilinas, asegurando un servicio continuo y de calidad.

1.1.4 Servicios

Arikok ofrece una amplia gama de servicios relacionados con la administración de edificios. Entre sus servicios más importantes se encuentran:

- **Gestión de contratos de arriendo:** Administra las relaciones contractuales con las empresas inquilinas.
- **Mantenimiento preventivo y correctivo:** Asegura la continuidad operativa de todos los sistemas del edificio, desde la infraestructura hasta la tecnología.
- **Control de acceso y seguridad:** Implementa soluciones para el control de acceso a las instalaciones, incluyendo áreas como los estacionamientos del inmueble y espacios comunes.
- **Optimización de espacios:** Evalúa constantemente el uso de los espacios para mejorar la eficiencia y satisfacción de las empresas inquilinas.

1.1.5 Departamentos afectados por el sistema a desarrollar

Actualmente, Arikok gestiona los estacionamientos a través de un sistema manual en el que la administración del edificio realiza el registro, la asignación y el control de los espacios mediante planillas Excel, mientras que la comunicación con las empresas inquilinas se lleva a cabo principalmente mediante correos electrónicos. Este método requiere un alto nivel de intervención humana y genera una carga de trabajo significativa, especialmente en la Gerencia de Operaciones, donde el personal de seguridad verifica presencialmente a los

usuarios en las barreras de acceso físicas, asegurando que solo personas autorizadas ingresen a los estacionamientos.

La nueva solución de gestión y control de estacionamientos reforzará los servicios provistos por Arikok, beneficiando principalmente a la Gerencia Comercial y Gerencia de Operaciones que son áreas clave dentro de la estructura organizacional, cómo se visualiza en el organigrama de la empresa.

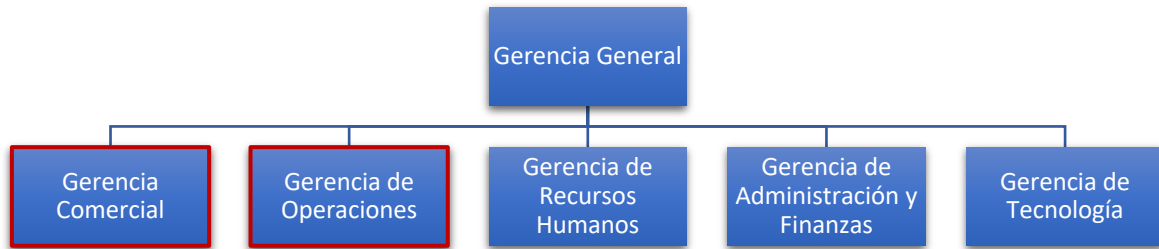


Figura 1-1. Organigrama empresa Arikok

Fuente: Elaboración propia

La implementación del sistema generará beneficios específicos y medibles para las dos áreas involucradas:

- **Para la Gerencia Comercial:** La solución optimizará la gestión de los estacionamientos asignados a las empresas inquilinas. Esto se traduce en un servicio más ágil y de mayor calidad, lo que es fundamental para fortalecer las relaciones comerciales y la satisfacción del cliente.
- **Para la Gerencia de Operaciones:** La principal ventaja será la automatización de los procesos manuales, lo que disminuirá significativamente la carga de trabajo del personal. Además, mejorará la seguridad en los accesos y permitirá una administración más eficiente de los espacios disponibles, gracias al control y acceso a la información en tiempo real.

1.2 Objetivos y beneficios del proyecto

El objetivo general es optimizar la gestión y el control de acceso a los estacionamientos de un edificio mediante una solución informática para la empresa Arikok. Con este fin, se establecieron los siguientes objetivos específicos.

1. **Optimizar la gestión operativa de los estacionamientos, incluyendo la asignación de espacios y la resolución de incidentes:** Reemplazar la gestión manual basada en correos y planillas por una plataforma centralizada que agilice tanto la asignación de espacios a empresas como el proceso de registro, seguimiento y resolución de incidentes operativos.
2. **Reducir el tiempo de acceso al estacionamiento mediante la automatización:** Minimizar las demoras en horario punta, producto de la verificación manual, mediante la automatización del control de las entradas y salidas. El sistema validará en tiempo real los permisos de acceso de cada usuario, integrándose con el hardware físico del estacionamiento.
3. **Mejorar la seguridad en el control de acceso al estacionamiento:** Mitigar el riesgo de accesos no autorizados, producto de un control manual propenso a errores. La plataforma garantizará que el ingreso sea permitido únicamente a usuarios con una asignación o reserva válida y vigente en el sistema, asegurando que solo personas autorizadas utilicen los espacios.
4. **Facilitar la comunicación, transparencia y autogestión para las empresas inquilinas:** Empoderar a las empresas inquilinas con herramientas de autogestión para que administren de forma autónoma sus estacionamientos, reservas y solicitudes de visita.
5. **Proveer información oportuna y relevante que apoye la toma de decisiones estratégicas y la gestión operativa:** Generar reportes detallados sobre la operación del estacionamiento. Esta información permitirá tanto a la administración como a las empresas inquilinas tomar decisiones estratégicas y operativas basadas en datos concretos, optimizando el uso de los recursos.

1.3 Descripción de la Situación sin Proyecto

Actualmente, la gestión y control de los estacionamientos en los edificios se lleva a cabo de manera manual, utilizando principalmente el correo electrónico como medio de comunicación entre la administración del edificio y las empresas inquilinas. Junto con el arriendo de las oficinas, existe la opción de incluir estacionamientos para los inquilinos. Sin embargo, el proceso de asignación y control de estos espacios es engorroso y presenta varias limitaciones operativas.

La gestión de los estacionamientos se realiza de forma manual. La administración del edificio se comunica con las empresas inquilinas a través de correos electrónicos para informar sobre la ubicación o número de los estacionamientos asignados. A su vez, las empresas deben enviar un correo electrónico a la administración con los nombres de las personas que harán uso de los estacionamientos asignados, lo que implica una alta dependencia entre ambas partes para coordinar este proceso.

Además, si una empresa inquilina necesita solicitar un estacionamiento para visitas, debe hacerlo enviando un correo con al menos 24 horas de anticipación. La administración del edificio revisa la disponibilidad de estacionamientos y, en caso de que haya espacio, asigna un estacionamiento temporal, comunicando su decisión nuevamente a través de correo electrónico.

El control de acceso a los estacionamientos también es manual y está a cargo del personal de la administración. Este personal verifica la tarjeta de identificación de los usuarios para garantizar que solo las personas autorizadas utilicen los estacionamientos asignados. A su vez, la administración mantiene un registro manual de las asignaciones de los estacionamientos, lo que dificulta tener un control preciso y actualizado sobre la ocupación de los espacios.

La gestión de incidencias por ocupación indebida también es manual. Al recibir un aviso, el personal verifica asignaciones en registros manuales, intenta identificar y contactar al infractor (proceso lento y no siempre exitoso) y busca una solución temporal para el afectado. Todo este proceso depende de la comunicación directa y la intervención del personal, generando demoras, consumiendo recursos administrativos y potencialmente causando insatisfacción en los usuarios afectados.

En definitiva, este sistema manual de gestión de estacionamientos es intrínsecamente lento y susceptible a errores humanos, lo cual compromete la eficiencia en el uso de los estacionamientos y deteriora la experiencia general de los usuarios del edificio.

1.4 Problemas detectados

En base a la situación actual, donde la gestión de los estacionamientos se realiza de manera manual, se han encontrado los siguientes problemas.

1. **Demoras en el acceso al estacionamiento del edificio en horario punta por la verificación manual realizada por el personal del estacionamiento:** Este problema afecta directamente la experiencia de los usuarios, generando largas esperas y un flujo ineficiente de vehículos.
2. **El control en el acceso a los estacionamientos es propenso a errores:** La falta de precisión en la verificación manual genera un riesgo significativo de que personas no autorizadas accedan a los estacionamientos.

3. **Falta de un control automatizado para asegurar que solo las personas autorizadas utilicen los estacionamientos asignados:** La ausencia de un sistema automatizado, como lectores de patentes o de tarjetas de verificación, genera un riesgo de uso indebido de los espacios asignados, ya que el personal encargado debe verificar manualmente el acceso.
4. **Elevada complejidad operativa en la gestión de incidencias:** La resolución manual de casos donde un estacionamiento asignado es ocupado indebidamente demanda un esfuerzo significativo por parte de la administración, que debe invertir tiempo significativo en identificar vehículos, contactar a los involucrados, buscar soluciones temporales y registrar cada incidente.
5. **Mayor dificultad para llevar un registro detallado y actualizado de la ocupación de los estacionamientos:** La falta de un registro en línea impide una correcta gestión de los espacios, lo que puede llevar a una ocupación ineficiente y pérdida de ingresos por no aprovechar al máximo los estacionamientos disponibles.
6. **Existe una mayor probabilidad de errores humanos en la asignación de los estacionamientos a las empresas inquilinas:** Los errores en la asignación de estacionamientos pueden generar conflictos entre la administración y las empresas inquilinas, afectando la relación comercial entre ambos.
7. **Menor eficiencia en la comunicación y coordinación entre la administración del edificio y las empresas inquilinas:** El uso del correo electrónico para coordinar la asignación de estacionamientos es un proceso lento y poco eficiente, lo que afecta la experiencia de las empresas inquilinas y la agilidad operativa.
8. **Limitaciones para que las empresas inquilinas puedan realizar una gestión interna de los estacionamientos:** El actual proceso manual impide a las empresas inquilinas gestionar eficientemente la asignación de sus espacios a colaboradores y aplicar políticas internas (como el cobro por uso), limitando severamente su autonomía operativa y la adaptación a sus necesidades específicas.
9. **Falta de transparencia en el proceso de solicitud y asignación de estacionamientos de visitas:** La solicitud de estacionamientos para visitas puede generar inconvenientes y malentendidos, afectando la confianza entre las empresas inquilinas y la administración.

1.5 Requerimientos y Requisitos del Usuario

A continuación, se describen las expectativas y necesidades que el usuario tienen con respecto al sistema. Estos requisitos establecen las funcionalidades y características que deben ser implementadas para asegurar que el sistema cumpla con los objetivos de negocio y operativos.

1. El usuario requiere de un sistema que reduzca las demoras en el acceso al estacionamiento, especialmente durante las horas punta, para mejorar la experiencia de los usuarios del estacionamiento.
2. El sistema debe garantizar que solo las personas autorizadas tengan acceso a los estacionamientos, proporcionando un control de acceso seguro y automatizado.
3. El usuario requiere la capacidad de gestionar usuarios dentro del sistema.
4. El usuario necesita la capacidad de registrar información de los edificios, incluyendo nombre, ubicación y características, además de asignar usuarios.
5. El usuario necesita la capacidad de gestionar los estacionamientos de cada edificio.
6. El usuario requiere la capacidad de gestionar empresas inquilinas dentro del sistema.
7. El usuario requiere la capacidad de gestionar las asignaciones de los estacionamientos tanto para las empresas inquilinas como para las visitas.
8. El usuario requiere que las empresas inquilinas puedan realizar solicitudes de estacionamiento para visitas a través del sistema.
9. El sistema debe permitir a la administración del edificio aprobar o rechazar las solicitudes de estacionamiento de visitas.
10. El usuario requiere que las reglas para reservar estacionamientos sean ajustables, para adaptarse a las necesidades cambiantes de la administración y las empresas inquilinas.
11. El usuario requiere que las empresas inquilinas tengan la capacidad de gestionar sus propias reservas de estacionamiento de forma sencilla a través del sistema.
12. El usuario requiere que las empresas inquilinas puedan configurar y administrar esquemas de cobro a sus empleados por el uso de estacionamientos directamente en el sistema.

13. El usuario requiere la capacidad de gestionar incidencias operativas de los estacionamientos a través del sistema.
14. El usuario requiere contar con funcionalidades de comunicación integradas en el sistema para facilitar la interacción entre la administración y los usuarios como parte del proceso de gestión de incidencias.
15. El sistema debe ofrecer la generación de reportes de uso y cobros para que las empresas inquilinas apliquen los respectivos cargos a sus empleados.
16. El sistema debe ofrecer la generación de reportes detallados que apoyen la gestión operativa y la toma de decisiones estratégicas, proporcionando una visión clara del uso y la ocupación de los estacionamientos.

1.6 Alternativas globales de solución

A continuación, se presentan las principales alternativas globales de solución para responder a necesidades y expectativas que el usuario tiene con respecto al sistema. Durante el proceso de investigación, se revisaron las soluciones disponibles en el mercado, encontrándose únicamente alternativas del tipo Software como Servicio (SaaS) que cumplen en mayor o menor medida con los requerimientos planteados.

1.6.1 Alternativa 1: Desarrollo a medida

El desarrollo a medida permite crear una solución específica para los requerimientos del usuario. Esta alternativa ofrece la flexibilidad de diseñar funcionalidades personalizadas que optimizan la gestión y el control de acceso a los estacionamientos, adaptándose a los procesos operativos de la empresa Arikok y los edificios comerciales que administra. Al no depender de soluciones comerciales genéricas, el desarrollo a medida garantiza una experiencia de usuario fluida, simplificando la gestión y la interacción de los usuarios sin experiencia previa.

Además, esta solución facilita la mantenibilidad del sistema, permitiendo incorporar futuras necesidades o ajustes en las reglas de asignación y/o control de acceso a medida que los requisitos de operación cambien. La arquitectura del software puede diseñarse para soportar un crecimiento gradual en usuarios y complejidad operativa, asegurando que el sistema siga siendo eficiente y confiable a lo largo del tiempo. En términos de seguridad, se puede implementar una solución que cumpla con los estándares y normativas de protección de datos, con controles de acceso y medidas de cifrado que aseguren la integridad y confidencialidad de la información para asegurar el cumplimiento de la ley de protección de datos chilena.

Aunque el desarrollo a medida implica un mayor costo inicial en comparación con soluciones comerciales genéricas, ofrece un control total sobre el código y la propiedad intelectual del sistema, eliminando la dependencia de terceros y reduciendo costos recurrentes como el de las suscripciones. Esta autonomía en la gestión del ciclo de vida del software facilita modificaciones futuras y garantiza que la empresa mantenga el control sobre las mejoras del sistema, alineándolo siempre con sus objetivos estratégicos y operativos.

1.6.2 Alternativa 2: Software como Servicio Kalena

La plataforma Kalena ofrece una solución estándar y configurable para la gestión de estacionamientos, proporcionando funcionalidades que permiten la administración eficiente de los espacios. Esta alternativa permite implementar rápidamente un sistema de gestión de estacionamientos, optimizando la reserva de espacios y mejorando la experiencia del usuario sin requerir desarrollos desde cero. Al tratarse de una plataforma ya establecida, facilita un uso para usuarios sin experiencia previa, con interfaces diseñadas para simplificar la navegación y operación del sistema.

Kalena también permite cierta adaptabilidad, pudiendo ajustar ciertas configuraciones para adaptarse a las necesidades de Arikok. Sin embargo, las posibilidades de personalización y expansión están limitadas a las características que la plataforma ofrece en sus distintas versiones o a través de actualizaciones específicas, lo que puede implicar ajustes en los procesos de Arikok para adaptarse a las capacidades de la plataforma.

En cuanto a la seguridad, Kalena incorpora medidas de protección de datos propias de su solución y sigue estándares internacionales en la protección de la información. No obstante, para asegurar el cumplimiento de la normativa local, es necesario validar que las configuraciones de Kalena se alineen con los requisitos de la ley de protección de datos chilena.

La implementación de Kalena presenta costos iniciales y recurrentes de suscripción, que, aunque pueden ser menores a la inversión inicial de un desarrollo a medida, representan una dependencia continua con el proveedor. Además, la propiedad intelectual de la plataforma es del proveedor, lo que limita la posibilidad de realizar modificaciones específicas sin la intervención de ellos, restringiendo la autonomía de Arikok sobre la tecnología.

Por otro lado, el modelo comercial de Kalena, basado en la cantidad de recursos administrados, implica que los costos aumentan a medida que crece el uso del sistema. El plan Enterprise de Kalena introduce costos recurrentes adicionales debido a los servicios personalizados, como la integración de usuarios corporativos, la gestión de reglas de reservas y la integración con los sistemas de control de acceso. Esto agrega una variable dinámica a la planificación financiera del proyecto, haciendo que el presupuesto dependa no solo de la implementación inicial, sino también de la proyección de crecimiento y de las necesidades operativas de la organización.

Para obtener más detalles de la alternativa de solución se puede revisar en su sitio web <https://kalena.es>.

1.6.3 Alternativa 3: Software como Servicio Click Visitas

La plataforma Click Visitas ofrece una solución eficiente y simplificada para la gestión de accesos y visitas en instalaciones corporativas, permitiendo a Arikok implementar rápidamente un sistema sin necesidad de desarrollar software desde cero. Esta alternativa proporciona una interfaz intuitiva y amigable, ideal para usuarios sin experiencia previa en sistemas de control de acceso, lo cual facilita la adopción por parte del personal y reduce la curva de aprendizaje.

Si bien Click Visitas permite ciertos ajustes en sus configuraciones, sus opciones de personalización están limitadas a las funcionalidades estándar del sistema. Esto significa que Arikok deberá adaptar algunos de sus procesos a las capacidades de la plataforma, ya que la incorporación de características adicionales requiere la intervención del proveedor. Estas limitaciones pueden restringir la flexibilidad del sistema en el futuro, especialmente si las necesidades de la organización evolucionan.

La seguridad es un aspecto importante de Click Visitas, que incluye medidas de protección de datos para asegurar la confidencialidad y el control de la información de los visitantes. Sin embargo, la plataforma no cuenta con certificaciones internacionales de protección de datos, lo que podría limitar la garantía de cumplimiento de algunos estándares globales. Por esta razón, se debe validar que las configuraciones de seguridad de Click Visitas cumplan con la normativa chilena de protección de datos, asegurando así el cumplimiento regulatorio local.

La implementación de Click Visitas implica tanto costos iniciales como recurrentes, destinados al soporte y mantenimiento de la plataforma. Como parte de los costos iniciales, también se debe considerar el desarrollo e integración de funcionalidades adicionales que no están incluidas en las opciones estándar de la plataforma. Aunque estos costos pueden ser menores a la inversión inicial de un desarrollo a medida, representan una dependencia continua con el proveedor, lo cual limita la autonomía tecnológica de Arikok, dado que la propiedad intelectual de la plataforma pertenece a la empresa ClickBi.

Además, cualquier integración o adecuación específica que Arikok requiera deberá ser evaluada y cotizada por el proveedor, lo que introduce una variable adicional en la planificación financiera del proyecto. Este modelo comercial implica que el presupuesto debe contemplar no solo los costos iniciales de implementación y los gastos recurrentes de soporte y mantenimiento, sino también cualquier costo adicional que surja de futuras necesidades operativas de la organización.

Para obtener más detalles de la alternativa de solución se puede revisar en su sitio web <https://clickbi.cl/Click-Visitas.html>.

1.7 Evaluación de Alternativas

A continuación, se describe el método de evaluación utilizado para analizar las alternativas mencionadas anteriormente. Para ello, se ha establecido una escala basada en criterios ponderados, los cuales permiten una valoración objetiva de cada alternativa. Estos criterios han sido seleccionados para reflejar la relevancia de los distintos aspectos que se consideran clave en la decisión final. Con este enfoque, se pretende asegurar una evaluación precisa y alineada con los objetivos del proyecto.

1.7.1 Criterios para evaluar las alternativas.

Los criterios seleccionados para la evaluación de las alternativas son los siguientes.

1. **Funcionalidad:** Evalúa la capacidad de la solución para satisfacer los requerimientos y requisitos específicos del negocio.
2. **Usabilidad:** Considera la facilidad de uso y la experiencia del usuario no experto.
3. **Mantenibilidad:** Mide la capacidad del sistema para evolucionar y adaptarse a futuros cambios o nuevas necesidades del negocio.
4. **Seguridad de la información:** Evalúa las garantías que ofrece la solución en cuanto a protección de datos y cumplimiento de normativas de seguridad.
5. **Costo:** Considera los costos iniciales y recurrentes de la solución en un horizonte de 3 años.
6. **Propiedad intelectual:** Analiza la propiedad y control sobre el código y/o tecnología de la solución.

1.7.2 Ponderación de Criterios y Escala de Evaluación

Este apartado presenta la ponderación de cada criterio y la escala de evaluación utilizada para calificar objetivamente cada alternativa, facilitando una selección fundamentada.

Tabla 1-1. Criterios de evaluación

CRITERIO	PONDERACIÓN
Funcionalidad	25%
Usabilidad	10%
Mantenibilidad	15%

Seguridad de la información	15%
Costo	20%
Propiedad intelectual	15%
TOTAL	100%

Fuente: Elaboración propia

Tabla 1-2. Escala de evaluación

CALIFICACIÓN	CONCEPTO
1	Muy malo
2	Malo
3	Regular
4	Bueno
5	Muy bueno

Fuente: Elaboración propia

1.7.3 Evaluación

A continuación, se presentan las evaluaciones de cada alternativa según los criterios definidos, incluyendo una justificación por cada criterio evaluado.

Tabla 1-3. Evaluación Alternativa 1: Desarrollo a medida

Alternativa 1: Desarrollo a medida		
Criterio	Calificación	Justificación
Funcionalidad	5	Permite la personalización completa de la solución, satisfaciendo todos los requerimientos.
Usabilidad	5	Interfaces diseñadas específicamente para los usuarios del sistema, facilitando la navegación y usabilidad.
Mantenibilidad	5	Permite ampliar o agregar nuevas funcionalidades para asegurar la utilidad del sistema a largo plazo.
Seguridad de la información	5	Permite la implementación de controles específicos de seguridad según la normativa de protección de datos chilena.
Costo	2	Alto costo inicial en comparación con soluciones SaaS, pero sin costos recurrentes significativos.
Propiedad intelectual	5	Arikok mantiene el control total sobre el sistema y su propiedad intelectual.

Fuente: Elaboración propia

Tabla 1-4. Evaluación Alternativa 2: Software como servicio Kalena

Alternativa 2: Software como servicio Kalena		
Criterio	Calificación	Justificación
Funcionalidad	4	Cubre los requerimientos en su mayoría, aunque con limitaciones en la personalización.
Usabilidad	5	Interfaz amigable y diseñada para facilitar la adopción por parte de los usuarios.
Mantenibilidad	3	Limitada a las configuraciones disponibles en el plan contratado.
Seguridad de la información	4	Incluye medidas de seguridad de estándar internacional, pero requiere validación adicional para la normativa chilena.
Costo	3	Costos iniciales y recurrentes a escala, pero las opciones de integración con sistemas externos o personalización tienen un costo adicional.
Propiedad intelectual	2	Dependencia del proveedor con restricciones para personalización y mejoras específicas.

Fuente: Elaboración propia

Tabla 1-5. Evaluación Alternativa 3: Software como Servicio Click Visitas

Alternativa 3: Software como servicio Click Visitas		
Criterio	Calificación	Justificación
Funcionalidad	3	Funcionalidades limitadas a la gestión de accesos y visitas.
Usabilidad	5	Interfaz simplificada y accesible para usuarios sin experiencia.
Mantenibilidad	3	Limitada, dependiente del proveedor.
Seguridad de la información	3	Incluye medidas de protección de datos, pero sin certificaciones internacionales de seguridad.
Costo	4	Costos iniciales y recurrentes fijos, lo cual elimina el riesgo de sobrecostos por mayor uso o crecimiento.
Propiedad intelectual	2	Dependencia significativa del proveedor con poca autonomía en personalización.

Fuente: Elaboración propia

A continuación, la siguiente tabla consolida los resultados de la evaluación, presentando una comparación cuantitativa de las tres alternativas analizadas. Su propósito es obtener una calificación final y objetiva para cada opción, facilitando así la selección de la solución más adecuada para Arikok.

En la tabla se desglosa, para cada criterio, los siguientes valores:

- **Ponderación:** El peso porcentual asignado a cada criterio según su importancia estratégica para el proyecto.
- **Calificación:** La puntuación de 1 a 5 que recibió cada alternativa (A1: Desarrollo a medida, A2: Kalena, A3: Click Visitas) en las evaluaciones previas.
- **Resultado Ponderado:** El valor obtenido al multiplicar la calificación de la alternativa por la ponderación del criterio.

Finalmente, la fila “TOTAL” suma estas puntuaciones ponderadas. Como se observa en los resultados, la alternativa de Desarrollo a medida (A1) obtiene la puntuación final más alta, posicionándose como la solución más idónea para los requerimientos del proyecto.

Tabla 1-6. Evaluación de alternativas: Resumen

CRITERIO	PONDERACIÓN	EVALUACIONES					
		CALIFICACIÓN			PONDERACIÓN		
		A1	A2	A3	A1	A2	A3
Funcionalidad	25%	5	4	3	1,25	1	0,75
Usabilidad	10%	5	5	5	0,5	0,5	0,5
Mantenibilidad	15%	5	3	3	0,75	0,45	0,45
Seguridad de la información	15%	5	4	3	0,75	0,6	0,45
Costo	20%	2	3	4	0,4	0,6	0,8
Propiedad Intelectual	15%	5	2	2	0,75	0,3	0,3
TOTAL	100%	27	21	20	4,4	3,45	3,25

Fuente: Elaboración propia

1.8 Alternativa seleccionada

A continuación, se presenta la alternativa seleccionada, detallando los aspectos que la convierten en la mejor opción para satisfacer las necesidades de Arikok. Este apartado incluye una descripción de la solución elegida y sus principales beneficios.

1.8.1 Descripción.

La alternativa de desarrollo a medida es la opción más adecuada para Arikok debido a su capacidad para satisfacer de manera integral las necesidades específicas de la empresa, ofreciendo flexibilidad, mantenibilidad, seguridad y control total sobre la solución. Esta opción permite diseñar un sistema completamente adaptado a los requerimientos particulares de la gestión de estacionamientos, asegurando que cada funcionalidad esté alineada con los

procesos internos de Arikok y optimizando la eficiencia operativa sin las limitaciones de un software estándar.

Una ventaja clave del desarrollo a medida es su mantenibilidad. La solución puede crecer junto con la empresa, permitiendo la ampliación de funcionalidades y ajustes en las reglas de operación conforme evolucionan las necesidades de Arikok, sin estar restringida por las configuraciones predefinidas de un proveedor. En cuanto a seguridad, el desarrollo a medida permite implementar los más altos estándares de protección de datos, asegurando que se cumplan todas las normativas requeridas, incluidas las regulaciones chilenas, para garantizar la integridad de la información de la empresa y de sus usuarios.

Otra razón decisiva es que Arikok retiene el control total sobre la propiedad intelectual del sistema, eliminando la dependencia de un proveedor externo. Esto le otorga una autonomía completa para realizar cualquier modificación o mejora futura, ajustando el sistema sin incurrir en costos adicionales de personalización o limitaciones impuestas por un tercero.

Aunque el costo inicial es alto en comparación con las otras alternativas, el desarrollo a medida resulta ser una inversión estratégica a largo plazo. Al eliminar pagos recurrentes a proveedores y posibles cargos por personalizaciones, esta opción permite un manejo más predecible y controlado del presupuesto a lo largo del tiempo.

En conjunto, estas características convierten al desarrollo a medida en la mejor opción para Arikok, ya que ofrece una solución adaptada, segura, escalable y autónoma, alineada plenamente con los objetivos estratégicos y operativos de la empresa y con una estructura de costos más optimizada a largo plazo.

1.8.2 Beneficios

A continuación, se presentan los principales beneficios que se obtendrán con la implementación de la alternativa seleccionada.

1. **Optimización de la gestión operativa:** El sistema permitirá una asignación más eficiente de los espacios, mejorando el aprovechamiento de los estacionamientos disponibles y agilizando la gestión y resolución de incidencias operativas.
2. **Reducción en los tiempos de acceso:** Al automatizar el control de acceso, se disminuirán las demoras especialmente en los horarios de mayor demanda, mejorando la fluidez vehicular y experiencia de los usuarios.
3. **Mejora en la seguridad del acceso:** La implementación de controles de acceso automatizados garantizará que solo las personas autorizadas utilicen los espacios asignados, disminuyendo el riesgo de accesos no autorizados.

4. **Mayor eficiencia en la administración:** Al reemplazar la gestión manual por una solución informática, se disminuirán los errores humanos y la carga de trabajo para el personal administrativo, permitiendo una administración más eficiente.
5. **Facilitación de la comunicación:** El sistema permitirá una comunicación más fluida y transparente entre la administración y las empresas inquilinas, tanto para la gestión diaria como para la comunicación específica durante la resolución de incidencias.
6. **Mayor autonomía y flexibilidad para las empresas inquilinas:** Otorgar a las empresas inquilinas la capacidad para autogestionar la asignación interna de sus espacios de estacionamiento, manejar reservar y aplicar sus propias políticas de uso, incluyendo la configuración y gestión de esquemas de cobro a sus empleados.
7. **Mejora en la gestión de estacionamientos para visitas:** El sistema permitirá una asignación más rápida y ordenada de los estacionamientos para visitas, facilitando su reserva anticipada y el control de su uso. Permitiendo a la administración contar con un registro claro y accesible de todas las solicitudes realizadas.
8. **Información valiosa para la gestión:** El sistema permitirá la generación de reportes detallados que apoyen la toma de decisiones estratégicas sobre la ocupación, el uso de los estacionamientos, patrones de acceso, incidencias recurrentes y cobros gestionados por las empresas inquilinas, que apoyen tanto la toma de decisiones como la optimización operativa continua.

1.9 Planificación

A continuación, se presentan los recursos humanos necesarios para el proyecto, detallando los roles y funciones requeridos para cada uno de ellos. Asimismo, se incluye la planificación temporal, especificando las tareas a ejecutar, su duración y los tiempos programados.

1.9.1 Desarrollo del plan de personal.

El equipo de trabajo está integrado por profesionales de gran experiencia, cada uno con funciones y responsabilidades específicas, orientadas a asegurar el cumplimiento de los objetivos del proyecto.

- **Jefe de proyecto (1):** Responsable de planificar, dirigir y supervisar todas las actividades del proyecto, asegurando el cumplimiento de los plazos, el presupuesto y

los objetivos establecidos, facilitando la comunicación entre los miembros del equipo y las partes interesadas, gestionando riesgos y resolviendo problemas, coordinando las fases del proyecto, y asegurando la calidad del producto final y la satisfacción del cliente.

- **Analista de sistemas (1):** Responsable de recolectar y documentar los requisitos del sistema y del usuario, analizar la viabilidad técnica y funcional de los requerimientos, crear diagramas de casos de uso y modelos conceptuales, especificar funcionalidades y colaborar estrechamente con el arquitecto de software y los desarrolladores para asegurar que el diseño cumpla con los requisitos.
- **Arquitecto de software (1):** Responsable de diseñar la arquitectura del sistema definiendo componentes, tecnologías e integraciones necesarias, seleccionando patrones de diseño y buenas prácticas para garantizar escalabilidad, seguridad y eficiencia, supervisa la implementación técnica para asegurar la adherencia a la arquitectura y documenta las decisiones de diseño para referencia futura.
- **Desarrolladores fullstack (2):** implementan tanto el backend como el frontend de la aplicación según las especificaciones técnicas y funcionales, escriben y mantienen código limpio y documentado, realizan pruebas unitarias e integraciones para validar la funcionalidad del sistema, y resuelven errores optimizando el rendimiento de la aplicación.
- **Tester QA (1):** Responsable de diseñar y ejecutar los planes de prueba para validar la funcionalidad del sistema, realiza pruebas funcionales, de regresión y de carga, reporta errores y trabaja con los desarrolladores para solucionarlos, asegurando que el producto final cumpla con los estándares de calidad y los requisitos del usuario.
- **Diseñador UX (1):** Responsable de diseñar interfaces de usuario enfocadas en mejorar la experiencia del usuario final, realiza investigaciones y pruebas de usabilidad para iterar y optimizar los diseños, crea prototipos y wireframes de alta fidelidad, y colabora con los desarrolladores para garantizar la correcta implementación de las interfaces.

Tabla 1-7. Desarrollo del plan de personal: Recursos

RECURSO	CANTIDAD	INICIAL
Jefe de proyecto	1	JP
Analista de sistemas	1	AN
Arquitecto de software	1	ARQ
Desarrollador Fullstack	2	DEV
Tester QA	1	QA
Diseñador UX	1	UX

Fuente: Elaboración propia

1.9.2 Planificación temporal, y asignación de recursos (Carta Gantt)

La presente carta Gantt detalla la planificación del proyecto, proporcionando una visión general de las actividades, hitos clave y recursos en cada etapa. Este documento tiene como objetivo facilitar el seguimiento del progreso, coordinar esfuerzos y asegurar el cumplimiento de los plazos establecidos. Además, se utilizará un enfoque iterativo incremental para garantizar adaptabilidad y mejora continua en cada fase del proyecto. Las actividades se organizan en iteraciones dentro de un ciclo de vida estructurado basado en OpenUP. Esta metodología, que es una versión ligera y esencial del Proceso Unificado, prioriza la colaboración, la mitigación de riesgos y la entrega continua de software funcional.

Tabla 1-8. Carta Gantt del proyecto

ID	NOMBRE DE TAREA	DURACIÓN	PRED.	RECURSOS
1	PROYECTO "GESTIÓN Y CONTROL DE ESTACIONAMIENTOS EN EDIFICIOS CORP."	103 días		
2	Inicio	15 días		
3	Iteración I1	15 días		
4	Gestión de Proyecto	15 días		
5	Kickoff	1 día		JP;AN[0,5];ARQ[0,5]
6	Planificación iteración	0,5 días	5	JP;AN;ARQ
7	Elaboración del documento de visión	5 días	11	JP[0,5]
8	Revisión iteración	0,5 días	7	JP;AN;ARQ
9	Requisitos	13 días		
10	Levantamiento de requisitos funcionales	5 días	6	JP[0,5];AN
11	Levantamiento de requisitos no funcionales	3 días	10	JP[0,5];AN;ARQ
12	Creación de casos de uso de alto nivel	5 días	11	JP[0,5];AN
13	Hito: Visión del Ciclo de Vida Acordada	0 días	8	
14	Elaboración	22 días		
15	Iteración E1	11 días		
16	Gestión de Proyecto	11 días		
17	Planificación iteración	0,5 días	13	JP;AN;ARQ

18	Seguimiento y gestión de riesgos	11 días	13	JP[0,5]
19	Revisión iteración	0,5 días	23	JP;AN;ARQ
20	Requisitos	10 días		
21	Especificación de casos de uso	5 días	17	AN
22	Validación de los CU con las partes interesadas	2 días	21	JP[0,5];AN;ARQ[0,5]
23	Documentación de las especificaciones de requisitos	3 días	22	AN
24	Iteración E2	11 días		
25	Gestión de Proyecto	11 días		
26	Planificación iteración	0,5 días	19	JP;ARQ
27	Seguimiento y gestión de riesgos	11 días	19	JP[0,5]
28	Revisión iteración	0,5 días	41	JP;ARQ;DEV;QA
29	Arquitectura	5 días		
30	Diseño de los diagramas UML del sistema	5 días	26	ARQ
31	Diseño de las interfaces de usuario	5 días	26	UX
32	Entorno	1 día		
33	Configuración del entorno de desarrollo	1 día	30	ARQ[0,5];DEV
34	Configuración del entorno de pruebas	1 día	30	ARQ[0,5];DEV
35	Implementación	2,5 días		
36	Implementación Prototipo Arquitectónico	2,5 días	34;33	ARQ;DEV[2]
37	Despliegue	0,5 días		
38	Instalación del prototipo en el entorno de pruebas	0,5 días	36	DEV[2]
39	Pruebas	5 días		
40	Creación del plan de pruebas	4 días	30;31	QA
41	Pruebas iniciales del prototipo	1 día	38	QA;DEV[2]
42	Hito: Arquitectura base estable	0 días	29	
43	Construcción	50 días		
44	Iteración C1 - Gestión de seguridad y usuarios	10 días		
45	Gestión de Proyecto	10 días		
46	Planificación iteración	0,5 días	42	JP;ARQ;DEV
47	Seguimiento y gestión de riesgos	10 días	42	JP[0,5]
48	Revisión iteración	0,5 días	59	JP;ARQ;DEV
49	Arquitectura	0,5 días		
50	Refinar diseño de los módulos de la iteración	0,5 días	46	ARQ;DEV[2]
51	Refinar diseño de los módulos de la iteración	0,5 días	46	UX
52	Implementación	5,5 días		
53	Implementación del modelo de datos	1 día	50	ARQ[0,5];DEV[2]
54	Desarrollo del módulo de seguridad	4,5 días	53	DEV
55	Desarrollo del módulo de usuarios	4,5 días	53	DEV

56	Despliegue	0,5 días		
57	Instalación del incremento en el entorno de pruebas	0,5 días	59	DEV[2]
58	Pruebas	2,5 días		
59	Ejecución de pruebas unitarias	0,5 días	54;55	DEV[2]
60	Pruebas de integración de los módulos	2 días	57	QA;DEV
61	Pruebas de usabilidad de los módulos	1 día	57	UX
62	Iteración C2 - Gestión de estacionamientos y reservas	10 días		
63	Gestión de Proyecto	10 días		
64	Planificación iteración	0,5 días	48	JP;ARQ;DEV
65	Seguimiento y gestión de riesgos	10 días	48	JP[0,5]
66	Revisión iteración	0,5 días	78	JP;ARQ;DEV
67	Arquitectura	0,5 días		
68	Refinar diseño de los módulos de la iteración	0,5 días	64	ARQ;DEV[2]
69	Refinar diseño de los módulos de la iteración	0,5 días	64	UX
70	Implementación	5,5 días		
71	Implementación del modelo de datos	1 día	68	ARQ[0,5];DEV[2]
72	Desarrollo del módulo de estacionamientos	4,5 días	71	DEV
73	Desarrollo del módulo de reservas	4,5 días	71	DEV
74	Despliegue	0,5 días		
75	Instalación del incremento en el entorno de pruebas	0,5 días	77	DEV[2]
76	Pruebas	2,5 días		
77	Ejecución de pruebas unitarias	0,5 días	72;73	DEV[2]
78	Pruebas de integración de los módulos	2 días	75	QA;DEV
79	Pruebas de usabilidad de los módulos	1 día	75	UX
80	Iteración C3 - Gestión de visitas e incidentes	10 días		
81	Gestión de Proyecto	10 días		
82	Planificación iteración	0,5 días	66	JP;ARQ;DEV
83	Seguimiento y gestión de riesgos	10 días	66	JP[0,5]
84	Revisión iteración	0,5 días	96	JP;ARQ;DEV
85	Arquitectura	0,5 días		
86	Refinar diseño de los módulos de la iteración	0,5 días	82	ARQ;DEV[2]
87	Refinar diseño de los módulos de la iteración	0,5 días	82	UX
88	Implementación	5,5 días		
89	Implementación del modelo de datos	1 día	86	ARQ[0,5];DEV[2]
90	Desarrollo del módulo de visitas	4,5 días	89	DEV
91	Desarrollo del módulo de incidentes	4,5 días	89	DEV
92	Despliegue	0,5 días		

93	Instalación del incremento en el entorno de pruebas	0,5 días	95	DEV[2]
94	Pruebas	2,5 días		
95	Ejecución de pruebas unitarias	0,5 días	90;91	DEV[2]
96	Pruebas de integración de los módulos	2 días	93	QA;DEV
97	Pruebas de usabilidad de los módulos	1 día	93	UX
98	Iteración C4 – Gestión de acceso físico	10 días		
99	Gestión de Proyecto	10 días		
100	Planificación iteración	0,5 días	84	JP;ARQ;DEV
101	Seguimiento y gestión de riesgos	10 días	84	JP[0,5]
102	Revisión iteración	0,5 días	113	JP;ARQ;DEV
103	Arquitectura	0,5 días		
104	Refinar diseño del módulo de la iteración	0,5 días	100	DEV[2];ARQ
105	Refinar diseño de los módulos de la iteración	0,5 días	100	UX
106	Implementación	5,5 días		
107	Implementación del modelo de datos	1 día	104	ARQ[0,5];DEV[2]
108	Desarrollo del módulo de acceso físico	4,5 días	107	DEV[2]
109	Despliegue	0,5 días		
110	Instalación del incremento en el entorno de pruebas	0,5 días	112	DEV[2]
111	Pruebas	2,5 días		
112	Ejecución de pruebas unitarias	0,5 días	108	DEV[2]
113	Pruebas de integración del módulo	2 días	110	QA;ARQ[0,5];DEV
114	Pruebas de usabilidad del módulo	1 día	110	UX
115	Iteración C5 - Gestión de cobros y reportes	10 días		
116	Gestión de Proyecto	10 días		
117	Planificación iteración	0,5 días	102	JP;ARQ;DEV
118	Seguimiento y gestión de riesgos	10 días	102	JP[0,5]
119	Revisión iteración	0,5 días	131	JP;ARQ;DEV
120	Arquitectura	0,5 días		
121	Refinar diseño de los módulos de la iteración	0,5 días	117	ARQ;DEV[2]
122	Refinar diseño de los módulos de la iteración	0,5 días	117	UX
123	Implementación	5,5 días		
124	Implementación del modelo de datos	1 día	121	ARQ[0,5];DEV[2]
125	Desarrollo del módulo de cobros	4,5 días	124	DEV
126	Desarrollo del módulo de reportes	4,5 días	124	DEV
127	Despliegue	0,5 días		
128	Instalación del incremento en el entorno de pruebas	0,5 días	130	DEV[2]
129	Pruebas	2,5 días		

130	Ejecución de pruebas unitarias	0,5 días	125;126	DEV[2]
131	Pruebas de integración de los módulos	2 días	128	QA;DEV
132	Pruebas de usabilidad de los módulos	1 día	128	UX
133	Hito: Capacidad Operacional Inicial	0 días	119	
134	Transición	16 días		
135	Iteración T1 - Verificación	7 días		
136	Gestión de Proyecto	7 días		
137	Planificación iteración	0,5 días	133	JP;ARQ;DEV;QA
138	Seguimiento y gestión de riesgos	7 días	133	JP[0,5]
139	Revisión iteración	0,5 días	147	JP;ARQ;DEV;QA
140	Entorno	2 días		
141	Configuración del entorno de producción	2 días	137	ARQ;DEV
142	Despliegue	5 días		
143	Instalación del sistema en el entorno de pruebas	1 día	137	ARQ;DEV
144	Documentación del sistema	3 días	141	DEV
145	Pruebas	5 días		
146	Ejecución del plan de pruebas	3 días	143	QA
147	Ejecución de pruebas de aceptación del usuario	3 días	146	QA[0,5];UX[0,5]
148	Implementación	5 días		
149	Resolución de errores y optimización del sistema	5 días	143	DEV[0,5]
150	Iteración T2 - Producción	9 días		
151	Gestión de Proyecto	9 días		
152	Planificación iteración	0,5 días	139	JP;ARQ;DEV
153	Seguimiento y gestión de riesgos	7 días	139	JP[0,5]
154	Revisión iteración	0,5 días	155;159	JP;ARQ;DEV
155	Transición al equipo de operaciones y mantenimiento	5 días	158	ARQ[0,5]; DEV[0,5]
156	Documentación de cierre administrativo	2 días	154	JP
157	Despliegue	6 días		
158	Instalación del sistema en el entorno de producción	1 día	152	ARQ;DEV
159	Capacitación a los usuarios del sistema	5 días	158	UX
160	Hito: Lanzamiento del producto	0 días	154	

Fuente: Elaboración propia

1.10 Administración del riesgo

A continuación, se presenta la gestión de riesgos, cuyo objetivo es ofrecer una visión clara y estructurada de los posibles desafíos que podrían afectar el alcance, el tiempo, los costos o la calidad del proyecto.

La prioridad de los riesgos se determina a partir de la siguiente fórmula:

$$\text{Prioridad} = (11 - \text{probabilidad}) \times (11 - \text{impacto}) \times \text{costo de retiro}$$

Este cálculo considera la probabilidad de ocurrencia, el impacto potencial y el costo de mitigación, de modo que los riesgos con menor puntaje resultante se interpretan como de mayor prioridad en la gestión del proyecto.

Cada riesgo ha sido evaluado con estos criterios y se han definido estrategias específicas para su mitigación o resolución, asegurando una administración efectiva y alineada con los objetivos del trabajo.

Tabla 1-9. Administración de Riesgos

CATEGORÍA	RIESGO	POSIBILIDAD	IMPACTO	COSTO	PRIORIDAD
Proyecto	Retrasos significativos en el cronograma del proyecto.	5	8	5	90
Producto	Problemas para definir los requerimientos.	5	7	4	96
Tecnología	Dependencia de sistemas físicos de control de acceso.	7	8	6	72
Proyecto	Inadecuada validación funcional del sistema.	4	9	5	70
Producto	Insuficiente experiencia técnica del equipo.	2	7	3	108
Organización	Problemas en las relaciones del equipo.	3	6	2	80
Proyecto	Resistencia de los usuarios a adoptar el sistema.	5	8	5	90
Producto	Interfaz poco intuitiva para los usuarios finales.	4	7	4	112
Organización	Falta de personal clave debido a enfermedades.	3	6	4	160
Proyecto	Falta de compromiso y apoyo de la alta dirección al proyecto.	3	6	2	80

Fuente: Elaboración propia

A continuación, se presenta el plan de mitigación de riesgos, estos se encuentran ordenados por orden de prioridad y se detallan las estrategias y acciones necesarias para prevenir o minimizar el impacto de los riesgos identificados, asegurando la continuidad y el éxito del proyecto.

Tabla 1-10. Plan de mitigación

RIESGO	PRIORIDAD	SEÑAL DE ALERTA	PLAN DE MITIGACIÓN PREVENTIVA	PLAN DE MITIGACIÓN CORRECTIVA	RESPONSABLE
Inadecuada validación funcional del sistema.	70	Alto número de errores reportados en pruebas iniciales.	Diseño exhaustivo de casos de prueba y pruebas continuas.	Extensión del ciclo de pruebas con foco en áreas críticas.	Tester QA
Dependencia de sistemas físicos de control de acceso.	72	Incompatibilidad o fallos en la integración de sistemas de control de acceso.	Pruebas de integración tempranas con sistemas externos.	Desarrollo de adaptadores o soluciones alternativas.	Arquitecto de software
Problemas en las relaciones del equipo.	80	Conflictos frecuentes o falta de colaboración entre miembros.	Sesiones regulares de team building y comunicación efectiva.	Mediación inmediata para resolver conflictos y reforzar objetivos comunes.	Jefe de proyecto
Falta de compromiso y apoyo de la alta dirección al proyecto.	80	Decisiones críticas retrasadas debido a la ausencia de la alta dirección en las reuniones clave del proyecto.	Identificar un sponsor comprometido desde etapas tempranas, que facilite la comunicación con la alta dirección.	Convocar reunión urgente con la alta dirección para reforzar el valor estratégico del proyecto y obtener compromisos explícitos.	Jefe de proyecto
Retrasos significativos en el cronograma del proyecto.	90	Cumplimiento parcial o tardío de hitos.	Reunión de seguimiento semanal del cronograma.	Reasignación de tareas y/o recursos y actualización del cronograma.	Jefe de proyecto

Resistencia de los usuarios a adoptar el sistema.	90	Baja adherencia de los usuarios a las capacitaciones ofrecidas.	Implementación de sesiones de capacitación recurrentes.	Organizar talleres emergentes de refuerzo en puntos críticos.	Diseñador UX
Problemas para definir los requerimientos.	96	Ambigüedades en los documentos de requerimientos.	Reuniones frecuentes con stakeholders para clarificar requerimientos.	Sesiones intensivas de recopilación y validación de requerimientos.	Jefe de proyecto y Analista de sistemas
Insuficiente experiencia técnica del equipo.	108	Baja productividad o errores frecuentes del equipo.	Plan de formación continua y materiales de apoyo.	Talleres intensivos de capacitación en áreas específicas.	Jefe de proyecto
Interfaz poco intuitiva para los usuarios finales.	112	Desempeño bajo en pruebas de usabilidad con usuarios finales.	Realización temprana de pruebas de usabilidad iterativas.	Reiteración de pruebas y ajustes inmediatos en la interfaz.	Diseñador UX
Falta de personal clave debido a enfermedades.	160	Falta de reemplazos planificados ante ausencias críticas.	Definir un protocolo claro para gestionar ausencias críticas.	Redistribuir tareas y roles para cubrir ausencias inesperadas.	Jefe de proyecto

Fuente: Elaboración propia

1.11 Estimación de costos

A continuación, se presenta un análisis detallado de la estimación de costos para la ejecución de este proyecto. Esta proyección financiera es fundamental para determinar la inversión requerida, se basa en el esfuerzo estimado del equipo profesional y desglosa los distintos costos, desde las rentas del personal hasta los gatos operativos. Las tablas siguientes ofrecerán una visión estructurada de cada componente del presupuesto. Como consideración fundamental, se asume que Arikok ejecuta sólo este proyecto y no tiene costos variables.

La tabla estructura de rentas detalla la renta mensual asignada a cada uno de los cargos profesionales que participarán en el proyecto expresada en Unidades de Fomento. Los valores fueron obtenidos desde la guía salarial TI 2024 -2025 de la empresa Chilena IT Hunters, especialista en profesionales TI del mercado local.

Tabla 1-11. Estructura de rentas mensuales

CARGO	SENIORITY	RENTA BRUTA
Jefe de proyectos	Senior	UF 123
Arquitecto de software	Senior	UF 142
Analista de sistemas	Semi Senior	UF 89
Desarrollador Fullstack 1	Semi Senior	UF 89
Desarrollador Fullstack 2	Semi Senior	UF 89
Tester QA	Semi Senior	UF 79
Diseñador UX	Semi Senior	UF 63
TOTAL		UF 674

Fuente: 12° Guía Salarial TI 2024 - 2025 IT HUNTERS

A continuación, se presentan los costos fijos y gastos de administración. Los costos fijos incluyen las rentas mensuales del equipo de proyecto y otros gastos mensuales constantes. Los gastos de administración corresponden a los proporcionales de contabilidad y recursos humanos, que representan el soporte organizacional requerido específicamente para la gestión de este proyecto.

Tabla 1-12. Estructura de costos fijos

CONCEPTO	VALOR
Rentas	UF 674
Otros	UF 100
TOTAL	UF 774

Fuente: Elaboración propia

Tabla 1-13. Estructura de gastos de administración

CONCEPTO	VALOR
Contabilidad	UF 35
Recursos Humanos	UF 60
TOTAL	UF 95

Fuente: Elaboración propia

A continuación, se presenta la tabla matriz de esfuerzo por funcionalidad que cuantifica en días la dedicación de cada recurso del proyecto. Dicha distribución del esfuerzo se detalla por cada fase o funcionalidad, lo que permite visualizar con precisión cómo se asignan las horas de trabajo a lo largo del ciclo de vida del proyecto.

Tabla 1-14. Matriz de esfuerzo por funcionalidad

FUNCIONALIDAD	RECURSO	GESTIÓN DE PROYECTO	REQUISITOS	ARQUITECTURA	IMPLEMENTACIÓN	PRUEBAS	ENTORNO	DESPLIEGUE	TOTAL (días)
Inicio	JP	4,5	6,5						11
	AN	1,5	13						14,5
	ARQ	1,5	3						4,5
Elaboración	JP	13	1						14
	AN	1	10						11
	ARQ	2	1	5	2,5		1		11,5
	DEV	0,5			5	2	2	1	10,5
	QA	0,5				5			5,5
	UX			5					5
Gestión de seguridad y usuarios	JP	6							6
	ARQ	1		0,5	0,5				2
	DEV	1		1	12	3			17
	QA					2			2
	UX			0,5		1			1,5
Gestión de estacionamientos y reservas	JP	6							6
	ARQ	1		0,5	0,5				2
	DEV	1		1	12	3			17
	QA					2			2
	UX			0,5		1			1,5
Gestión de visitas e incidentes	JP	6							6
	ARQ	1		0,5	0,5				2
	DEV	1		1	12	3			17

	QA					2			2
	UX			0,5		1			1,5
Gestión de acceso físico	JP	6							6
	ARQ	1		0,5	0,5	1			3
	DEV	1		1	12	3			17
	QA					2			2
	UX			0,5		1			1,5
Gestión de cobros y reportes	JP	6							6
	ARQ	1		0,5	0,5				2
	DEV	1		1	12	3			17
	QA					2			2
	UX			0,5		1			1,5
Transición	JP	11							11
	ARQ	4,5					2	1	7,5
	DEV	4,5			2,5		2	5	14
	QA	1				4			5
	UX					1		5	6
TOTALES (días)		85,5	34,5	20	72,5	43	7	12	274,5

Fuente: Elaboración propia

La tabla costo del proyecto consolida y calcula el costo total. Para cada recurso humano, se muestra su participación porcentual en las rentas, se calcula su participación en el costo de la empresa que corresponde a la suma de los costos fijos y gastos de administración, el costo por día y los días asignados al proyecto. Al multiplicar el costo diario por los días asignados se obtiene el costo por recurso, cuyos valores se suman para el subtotal del proyecto. A este monto se agrega un 5% por imprevistos, obteniéndose así el costo final estimado en UF.

Tabla 1-15. Costo del proyecto

RRHH	PART. RENTAS	PART. COSTO EMPRESA	COSTO POR DÍA	DÍAS PROY.	COSTO PARA EMPRESA	COSTO PROYECTO
JP	18,32%	UF 159,14	UF 7,96	66	UF 525,15	
ARQ	13,15%	UF 114,25	UF 5,71	34,5	UF 197,09	
AN	21,14%	UF 183,62	UF 9,18	25,5	UF 234,12	
DEV1	13,15%	UF 114,25	UF 5,71	57,25	UF 327,05	
DEV2	13,15%	UF 114,25	UF 5,71	57,25	UF 327,05	
QA	11,74%	UF 102,01	UF 5,10	20,5	UF 104,56	
UX	9,35%	UF 81,23	UF 4,06	18,5	UF 75,14	
SUBTOTAL					UF 1.790,15	UF 1.790,15
IMPREVISTOS (5%)						UF 89,51
COSTO FINAL						UF 1.879,66

Fuente: Elaboración propia

**CAPÍTULO 2. ASPECTOS RELEVANTES DEL ANÁLISIS PARA LA
ALTERNATIVA SELECCIONADA**

2 ASPECTOS RELEVANTES DEL ANÁLISIS

Este capítulo se enfoca en los aspectos relevantes del análisis para el sistema de gestión y control de estacionamientos. A lo largo de este capítulo, se presentará una descripción detallada de la solución, ofreciendo una visión clara de su alcance y funcionalidades. Esta perspectiva se complementará con el diagrama de actividades, la definición de los requerimientos del sistema y descripción de su estructura funcional. Adicionalmente, el capítulo incluirá la matriz de trazabilidad, el modelo conceptual y el modelo de casos de uso detallado.

2.1 Descripción de la solución propuesta

La solución propuesta para optimizar la gestión y el control de estacionamientos en los edificios comerciales consiste en el desarrollo a medida de un sistema informático. Este sistema se materializará como una aplicación web responsiva, diseñada para ser accesible desde diversos dispositivos, facilitando así su uso por parte de la administración del edificio y las empresas inquilinas.

El desarrollo del sistema se llevará a cabo siguiendo un enfoque ágil, iterativo e incremental, utilizando la metodología OpenUP. Esta aproximación permite entregar valor de forma temprana y continua a través de incrementos, adaptando el sistema a las necesidades que puedan surgir durante el proyecto.

La aplicación será modular, lo que facilitará su desarrollo, mantenimiento y futuras actualizaciones. Los módulos principales que compondrán el sistema son los siguientes:

1. **Gestión de seguridad:** Este módulo transversal garantizará la integridad y confidencialidad del sistema mediante la gestión de tres pilares fundamentales: la autenticación de usuarios, la autorización de acceso a funcionalidad según roles y perfiles, y el registro de auditoría de todas las acciones críticas que se realicen en la plataforma.
2. **Gestión de usuarios:** Centralizará el ciclo de vida de las cuentas de usuario. Facilitará la administración de los distintos roles del sistema y la gestión de sus respectivos métodos de acceso, tanto lógicos (contraseñas) como físicos (identificadores para vehículos).
3. **Gestión de Estacionamientos:** Constituirá el núcleo del modelo de dominio, permitiendo registrar y administrar la información estructural de los activos físicos y contractuales. Abarcará la gestión de los edificios junto con sus estacionamientos, las empresas inquilinas y el control de las asignaciones contractuales.

4. **Gestión de Reservas:** Orquestrará la asignación y el uso de los estacionamientos. Ofrecerá a las empresas inquilinas herramientas de autogestión para administrar las plazas asignadas, permitiendo tanto la asignación directa a empleados como la reserva a demanda por parte de los usuarios, de acuerdo con las políticas configuradas.
5. **Gestión de Visitas:** Formalizará el flujo de solicitud y autorización de estacionamientos para visitas. Canalizará las peticiones de los empleados y proveerá a la administración del edificio una interfaz centralizada para aprobar o rechazar dichas solicitudes, mejorando así la trazabilidad, eficiencia y transparencia del proceso.
6. **Gestión de Incidentes:** Establecerá un canal formal para el reporte y seguimiento de incidencias operativas. Proveerá una plataforma centralizada para registrar, clasificar y gestionar eventos como ocupaciones indebidas o problemas de acceso, facilitando la comunicación entre el usuario afectado y el personal de operaciones hasta su resolución.
7. **Gestión de acceso físico:** Materializará el control de acceso automatizado, registrando las entradas y salidas de los vehículos en tiempo real para optimizar el flujo vehicular y reforzar la seguridad. El módulo contemplará la capacidad de interactuar con hardware externo, integrándose con sistemas físicos como barreras vehiculares o lectores de patentes para validar los permisos de acceso de forma automática.
8. **Gestión de cobros:** Otorgará autonomía a las empresas inquilinas para implementar políticas de monetización interna. Permitirá configurar y administrar esquemas de cobros flexibles (por minuto, hora o día) aplicables a sus empleados por el uso de los estacionamientos, generando los datos necesarios para su gestión administrativa.
9. **Generación de reportes:** Ofrecerá la capacidad de generar informes y visualizaciones detalladas sobre métricas clave como la ocupación, patrones de uso, cobros e incidentes, apoyando con ello tanto la gestión estratégica de Arikok como la administración interna de las empresas inquilinas.

En conjunto, esta solución informática busca reemplazar los procesos manuales actuales, optimizando la eficiencia operativa, mejorando la experiencia de los usuarios y proveyendo herramientas robustas para la gestión integral de los estacionamientos por parte de Arikok.

2.2 Diagrama de actividades general UML (flujo administrativo)

El presente apartado tiene como finalidad presentar la propuesta de solución mediante los principales flujos administrativos que el sistema implementará. Para ello, se utilizarán diagramas de actividades UML que integran explícitamente al sistema a desarrollar como uno de los actores del flujo, permitiendo representar de forma clara de la secuencia de acciones, los puntos de decisión y las responsabilidades tanto de los usuarios como del propio sistema dentro de cada proceso clave.

2.2.1 Asignación de estacionamientos a empresas inquilinas

Este diagrama de actividad detalla el proceso mediante el cual el Administrador del edificio registra una nueva empresa inquilina y le asigna los estacionamientos correspondientes dentro del sistema.

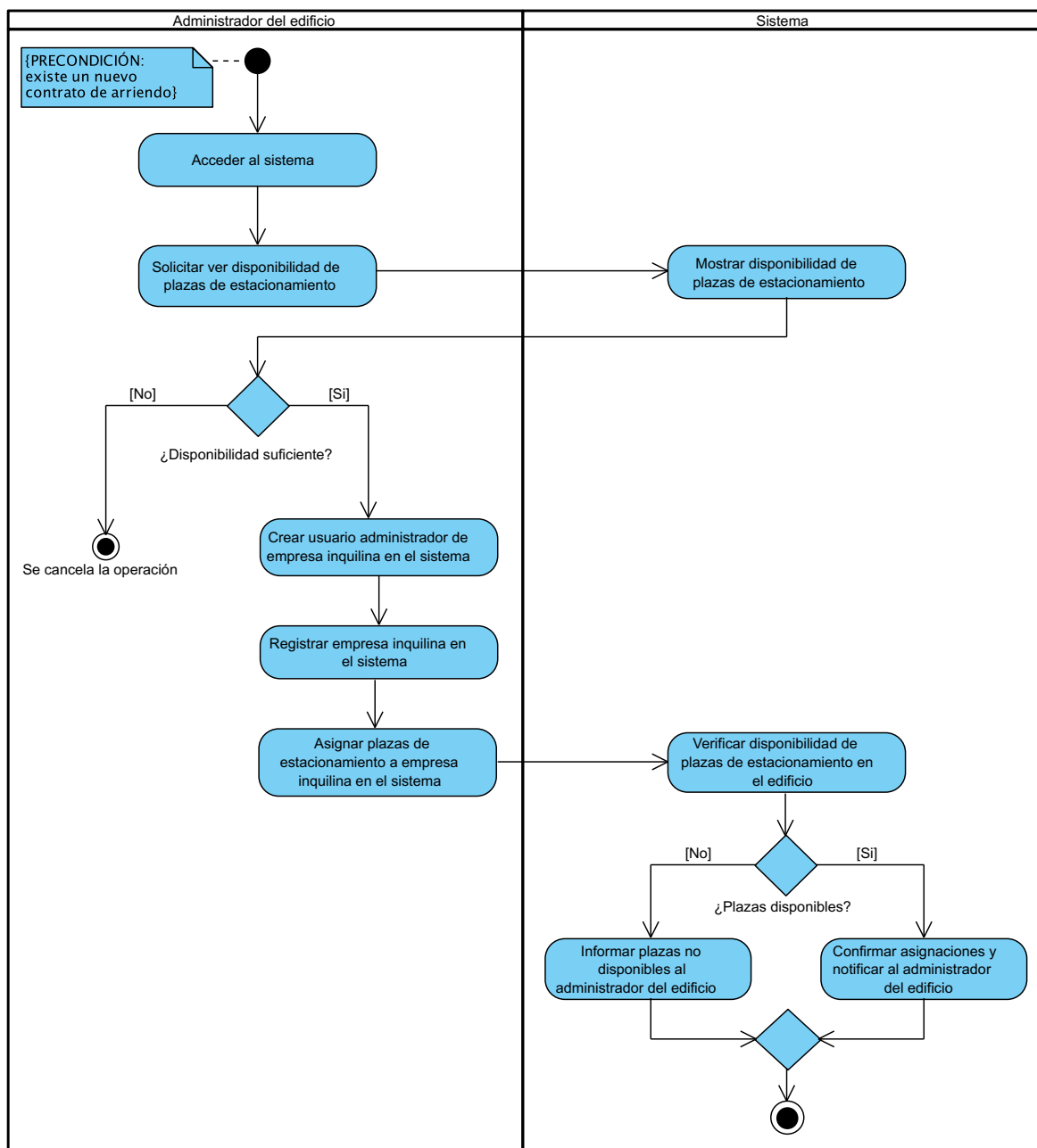


Figura 2-1. Diagrama de actividad: Asignación de estacionamientos a empresas inquilinas

Fuente: Elaboración propia

2.2.2 Solicitud de estacionamientos

Este diagrama de actividad detalla el proceso que sigue un Empleado de la empresa inquilina para solicitar un estacionamiento.

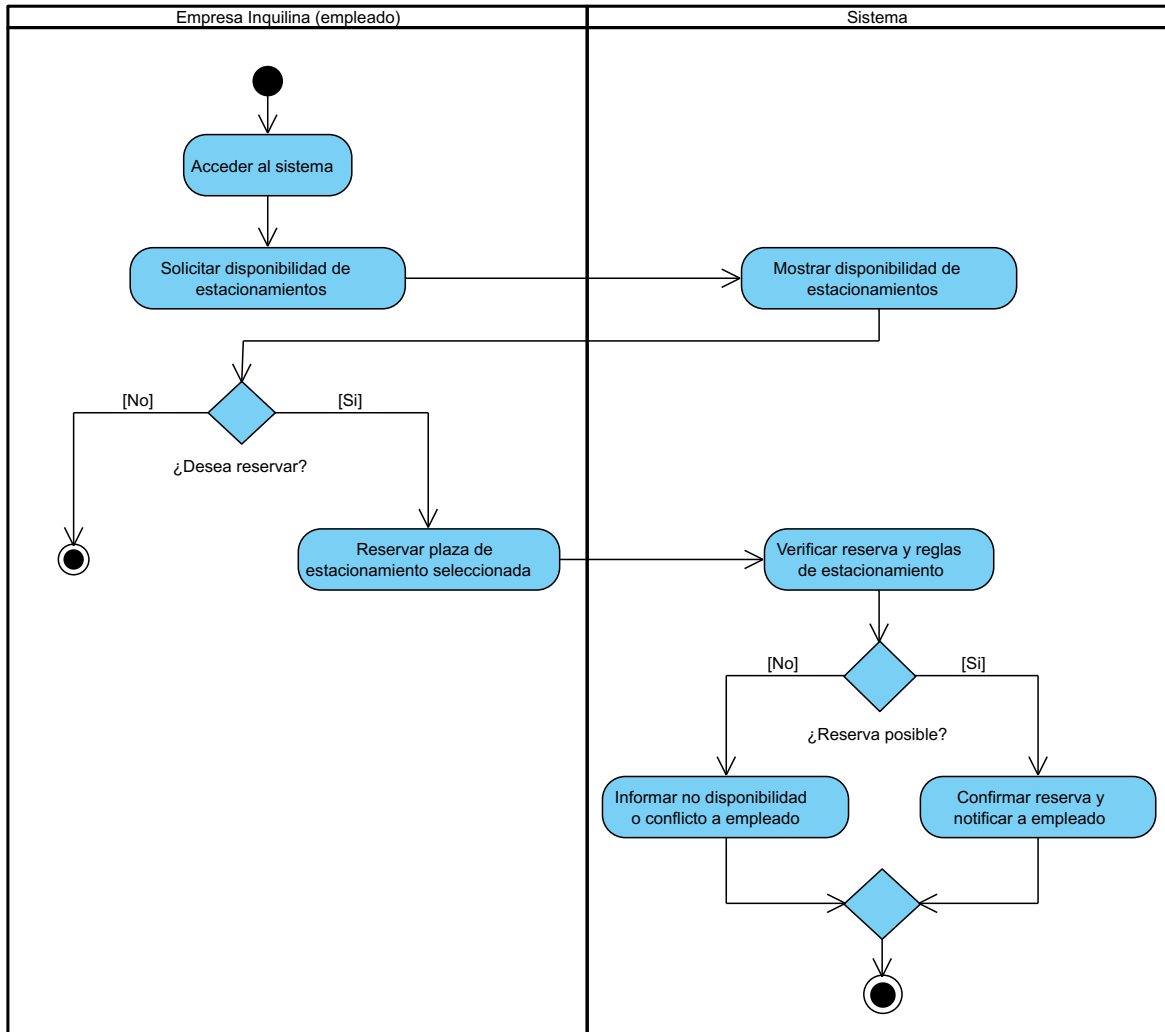


Figura 2-2. Diagrama de actividad: Solicitud de estacionamientos

Fuente: Elaboración propia

2.2.3 Solicitud de estacionamientos de visitas

Este diagrama de actividad detalla el proceso completo y asíncrono para la gestión de solicitudes de estacionamientos destinados a visitas.

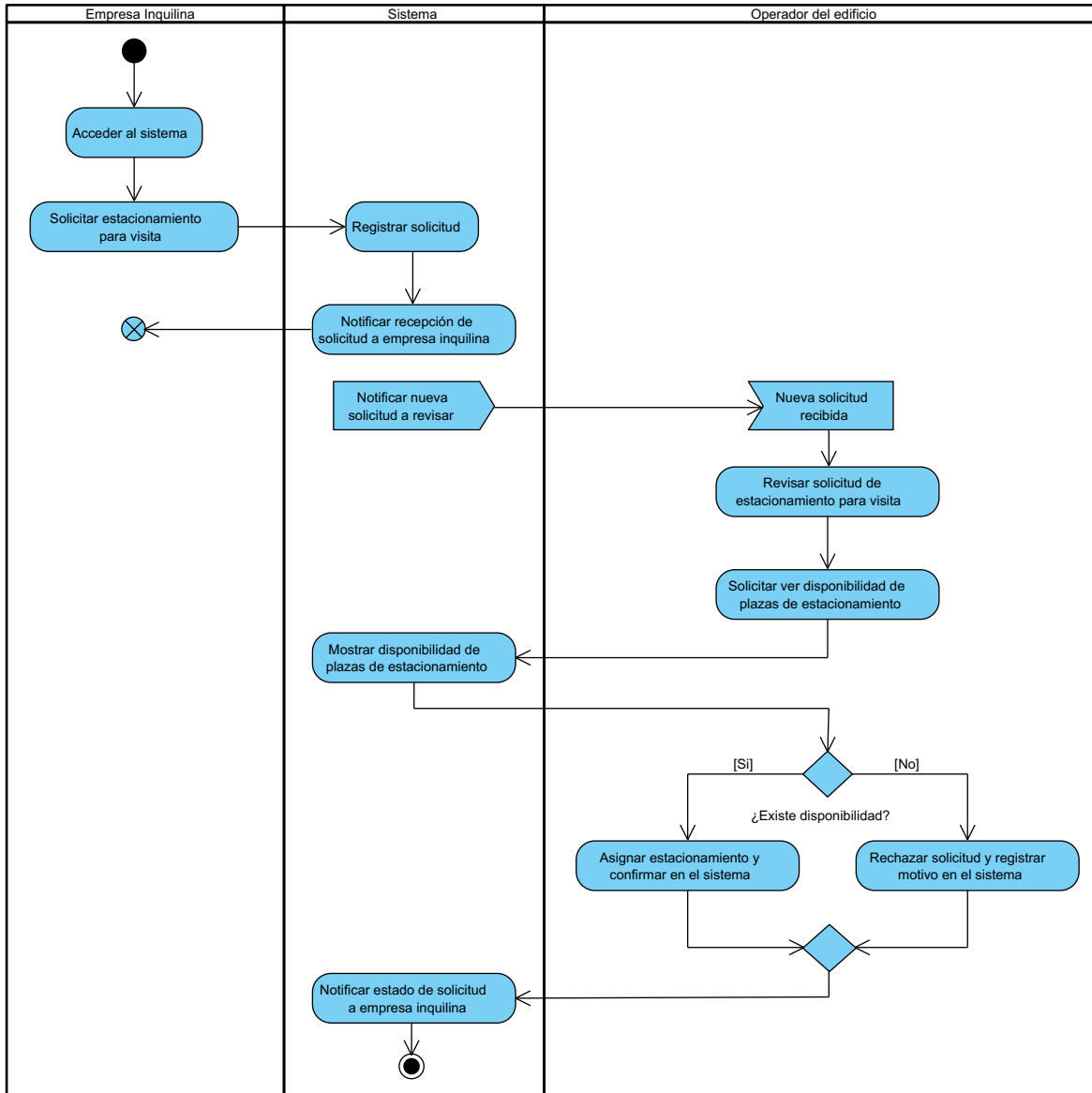


Figura 2-3. Diagrama de actividad: Solicitud de estacionamiento de visitas

Fuente: Elaboración propia

2.2.4 Gestión de incidentes

Este diagrama de actividad detalla el proceso completo y asíncrono para la gestión de incidentes relacionados con el uso de los estacionamientos.

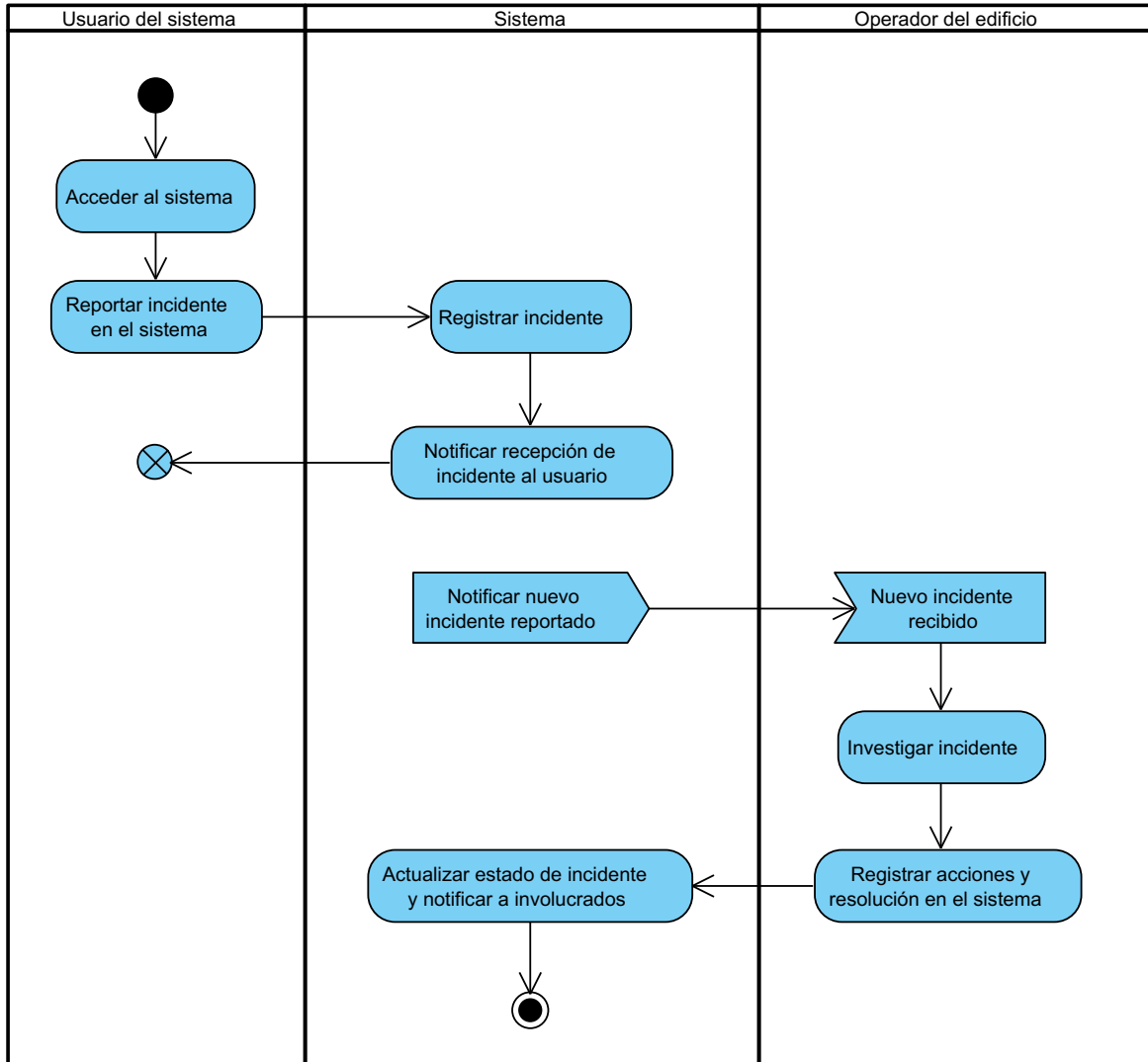


Figura 2-4. Diagrama de actividad: Gestión de incidentes

Fuente: Elaboración propia

2.3 Requerimientos del sistema (funcionales y no funcionales)

A continuación, se detallan los requerimientos que el sistema debe cumplir, divididos en funcionales, que describen las tareas específicas que el sistema realizará, y no funcionales, que definen las restricciones y atributos de calidad de un sistema. Estos requerimientos son cruciales para guiar el desarrollo y asegurar que el producto final satisfaga las necesidades de Arikok y sus usuarios.

2.3.1 Requerimientos funcionales

Los requerimientos funcionales están descritos en la siguiente tabla.

Tabla 2-1. Requerimientos funcionales

ID	DESCRIPCIÓN
RF-01	El sistema debe permitir al Administrador del sistema crear, visualizar, actualizar y eliminar lógicamente cuentas de usuarios para los roles Administrador del sistema y Administrador del edificio.
RF-02	El sistema debe permitir a los usuarios registrados autenticarse mediante su identificador de usuario y contraseña para acceder a las funcionalidades del sistema.
RF-03	El sistema debe permitir a un usuario autenticado cambiar su contraseña actual por una nueva.
RF-04	El sistema debe proveer un mecanismo para que los usuarios registrados que hayan olvidado su contraseña puedan restablecerla de forma segura.
RF-05	El sistema debe permitir al Administrador del sistema crear, visualizar, actualizar y eliminar lógicamente edificios en el sistema.
RF-06	El sistema debe permitir al Administrador del sistema asignar usuarios con el rol de “administrador del edificio” para que gestionen uno o más edificios registrados en el sistema.
RF-07	El sistema debe permitir al Administrador del edificio crear, visualizar, actualizar y eliminar lógicamente empresas inquilinas en los edificios que administra.
RF-08	El sistema debe permitir al Administrador del edificio crear, visualizar, actualizar y eliminar lógicamente cuentas de usuario para los operadores del edificio y administradores de las empresas inquilinas asociados a los edificios que gestiona.
RF-09	El sistema debe permitir al Administrador de la empresa inquilina gestionar integralmente la información de los empleados de su empresa, lo que incluye la capacidad para crear, visualizar, actualizar y eliminar lógicamente tanto las cuentas de usuario como sus vehículos y métodos de acceso asociados.
RF-10	El sistema debe permitir al Administrador de la empresa inquilina asignar roles de administrador de empresa, analista de empresa y usuario de empresa a los empleados de su empresa que determinarán sus permisos sobre las funcionalidades del sistema.
RF-11	El sistema debe permitir al Administrador del edificio crear, visualizar, actualizar y eliminar lógicamente plazas de estacionamiento.

RF-12	El sistema debe permitir al Administrador del edificio crear, visualizar, actualizar y eliminar lógicamente las asignaciones contractuales de plazas de estacionamiento a las empresas inquilinas.
RF-13	El sistema de forma automatizada o con intervención del Operador del edificio debe controlar y registrar las entradas y salidas de vehículos para reducir demoras y mejorar la seguridad.
RF-14	El sistema debe tener la capacidad de integrarse con controles de acceso físico existentes, tales como lectores de RFID y de patentes para automatizar el control de acceso.
RF-15	El sistema debe permitir al Administrador de la empresa inquilina configurar y ajustar las reglas para la reserva de estacionamientos, tales como antelación, duración y cantidad de reservas por unidad de tiempo.
RF-16	El sistema debe permitir al Administrador de la empresa inquilina crear, visualizar, actualizar y eliminar lógicamente reservas de estacionamientos de su empresa.
RF-17	El sistema debe permitir a los Empleados de una empresa inquilina crear reservas de estacionamiento o visualizar y eliminar lógicamente sus reservas de estacionamiento.
RF-18	El sistema debe permitir a los Empleados de una empresa inquilina crear solicitudes de estacionamiento de visita, visualizar y eliminar lógicamente sus propias solicitudes de estacionamiento de visitas.
RF-19	El sistema debe permitir al Operador del edificio visualizar, aprobar o rechazar las solicitudes de estacionamiento de visitas. El proceso de aprobación deberá incluir la asignación de un estacionamiento de visitas disponible.
RF-20	El sistema debe permitir al Administrador de la empresa inquilina configurar y administrar esquemas de cobros a los empleados de su empresa por el uso de estacionamientos.
RF-21	El sistema debe permitir a los usuarios del sistema registrar incidentes operativos relacionados con los estacionamientos, tales como ocupación indebida, problemas de acceso, daños y robos.
RF-22	El sistema debe permitir a los usuarios del sistema visualizar sus incidentes operativos registrados y añadir comentarios a los mismos.
RF-23	El sistema debe permitir al Operador del edificio realizar el seguimiento, actualizar el estado, añadir comentarios y registrar la resolución de los incidentes operativos reportados en los edificios que están asignados.
RF-24	El sistema debe permitir al Analista de la empresa inquilina generar y visualizar reportes del uso de los estacionamientos y de los cobros aplicados a sus empleados.
RF-25	El sistema debe permitir al Administrador del edificio generar y visualizar reportes detallados sobre la asignación y uso de los estacionamientos, patrones de acceso e incidentes para apoyar la gestión operativa y toma de decisiones estratégicas.

Fuente: Elaboración propia

2.3.2 Requerimientos no-funcionales

Los requerimientos no funcionales están descritos en la siguiente tabla.

Tabla 2-2. Requerimientos no funcionales

ID	DESCRIPCIÓN	FORMA DE VERIFICACIÓN
RNF-01	Trazabilidad: El sistema generará logs de auditoría para toda operación de Creación, Actualización o Eliminación (CUD) de datos en cualquier entidad del sistema que sea mutable. Cada entrada de log registrará la identificación del usuario, fecha/hora, tipo de modificación, el nombre de la entidad, el identificador único de la entidad y una copia completa de todos sus datos (estado posterior para creaciones o actualizaciones; estado anterior para eliminaciones). Estos logs serán protegidos contra alteraciones y se conservarán por un mínimo de 12 meses.	Se ejecutarán pruebas funcionales con operaciones de CUD que abarquen todas las entidades de datos modificables del sistema. Posteriormente, se inspeccionarán los logs de auditoría para: 1. Verificar la correcta generación de una entrada de log por cada operación CUD y su contenido completo (incluyendo usuario, fecha/hora, tipo de modificación, identificador y todos los datos de la entidad afectada según la operación). 2. Evaluar los mecanismos de seguridad implementados para los logs (como protección contra modificaciones no autorizadas y controles de acceso). 3. Comprobar la configuración y aplicación de la política de retención de logs.
RNF-02	Compatibilidad: La aplicación web responsiva deberá ser compatible y funcional en las últimas dos versiones de los navegadores web Google Chrome y Safari.	Pruebas de compatibilidad en Chrome y Safari (<i>2 últimas versiones major estables al momento de la verificación</i>). Se valida visualización correcta, ejecución exitosa de flujos críticos y ausencia de errores bloqueantes en la consola del navegador.
RNF-03	Usabilidad: Un usuario nuevo, con el perfil adecuado y tras una capacitación de máximo 1 hora, deberá ser capaz de completar las tareas principales asociadas a su rol (ej. realizar una reserva, registrar una visita, gestionar usuarios de su empresa) de forma intuitiva.	Pruebas de usabilidad con usuarios representativos de cada rol, observando la facilidad para completar tareas, midiendo la cantidad de errores cometidos y recogiendo feedback. Los errores no deben exceder a 1 por funcionalidad asociada a su rol.
RNF-04	Rendimiento: El sistema deberá cargar las funcionalidades de inicio de sesión, reserva de estacionamientos y confirmación de reserva de	Pruebas de carga simulando el número especificado de usuarios concurrentes y midiendo los tiempos de respuesta de las funcionalidades clave.

	estacionamiento en menos de 3 segundos con una carga concurrente de hasta 100 usuarios y con un ancho de banda mínimo de 10Mbps.	
RNF-05	Disponibilidad: El sistema deberá tener una disponibilidad del 99.5% durante el horario operativo definido de lunes a viernes de 07:00 a 21:00 hrs), excluyendo mantenimientos programados.	Monitoreo del uptime del sistema durante un período de 1 mes. Cálculo del porcentaje de disponibilidad.
RNF-06	Mantenibilidad: La arquitectura modular del sistema y la documentación del código deberán permitir que un nuevo desarrollador con la experiencia adecuada pueda entender y realizar modificaciones en un módulo específico en un plazo no superior a 5 días hábiles.	Revisión de la documentación del código y la arquitectura. Prueba práctica asignando una tarea de modificación simple a un desarrollador no familiarizado.
RNF-07	Seguridad - Confidencialidad e Integridad de Datos: Toda la información sensible (ej. contraseñas, datos personales) deberá almacenarse cifrada. El sistema debe cumplir con los principios de la ley de protección de datos chilena.	Revisión de la configuración de la base de datos, algoritmos de cifrado utilizados y auditoría de cumplimiento normativo. Pruebas de penetración (pentesting) sobre el sistema.
RNF-08	Seguridad - Control de Acceso: El sistema deberá asegurar que los usuarios solo puedan acceder a las funcionalidades y datos permitidos según su rol asignado.	Pruebas funcionales específicas para cada rol, intentando acceder a funcionalidades no permitidas.
RNF-09	Interoperabilidad: El sistema deberá poder integrarse de forma efectiva con los sistemas de control de acceso físico existentes (ej. barreras, lectores de patentes), asegurando una latencia en la comunicación no mayor a 5 segundos.	Pruebas de integración con los sistemas físicos especificados, verificando el intercambio de datos y la respuesta en tiempo real.

Fuente: Elaboración propia

2.4 Estructura funcional del sistema (caso de uso general)

Para ilustrar la interacción entre los distintos usuarios y las funcionalidades clave del sistema, a continuación, se presenta la estructura funcional mediante un diagrama de casos de uso general.

En este diagrama, los casos de uso se formulan con el verbo “gestionar” seguido de la entidad correspondiente, entendiendo “gestionar” como una referencia genérica a operaciones de administración sobre dicha entidad.

Cabe destacar que, según las reglas de negocio y los permisos asociados a cada actor, no todos los casos de uso de tipo “gestionar” contemplan necesariamente el conjunto de operaciones CRUD; en algunos casos, el alcance se restringe a un subconjunto de estas acciones, lo cual se detalla en la especificación de cada caso de uso.

Los actores que participarán del sistema son los siguientes:

1. **Administrador del sistema:** Responsable de la configuración y gestión global del sistema.
2. **Administrador de edificio:** Responsable de la gestión de uno o más edificios específicos.
3. **Operador de edificio:** Personal operativo que maneja el día a día los accesos vehiculares e incidentes en un edificio.
4. **Administrador de la empresa inquilina:** Representante de una empresa que arrienda espacios en el edificio, gestiona empleados y los recursos de estacionamiento para su empresa.
5. **Empleado de la empresa inquilina:** Empleado de una empresa inquilina que utiliza los servicios de estacionamiento provistos en el edificio.
6. **Analista de la empresa inquilina:** Empleado de una empresa inquilina enfocado en la visualización de los reportes de uso y cobro de estacionamientos para su empresa.
7. **Usuario del sistema:** Cualquier usuario que interactúa con el sistema independiente de su rol. Este actor no representa un tipo de usuario, sino una generalización que agrupa todos los usuarios del sistema.
8. **Sistema de control de acceso físico:** Hardware/Software externo para el control de barreras, lectores de patentes, etc.
9. **Servicio de email:** Sistema externo para enviar las notificaciones mediante correo electrónico.

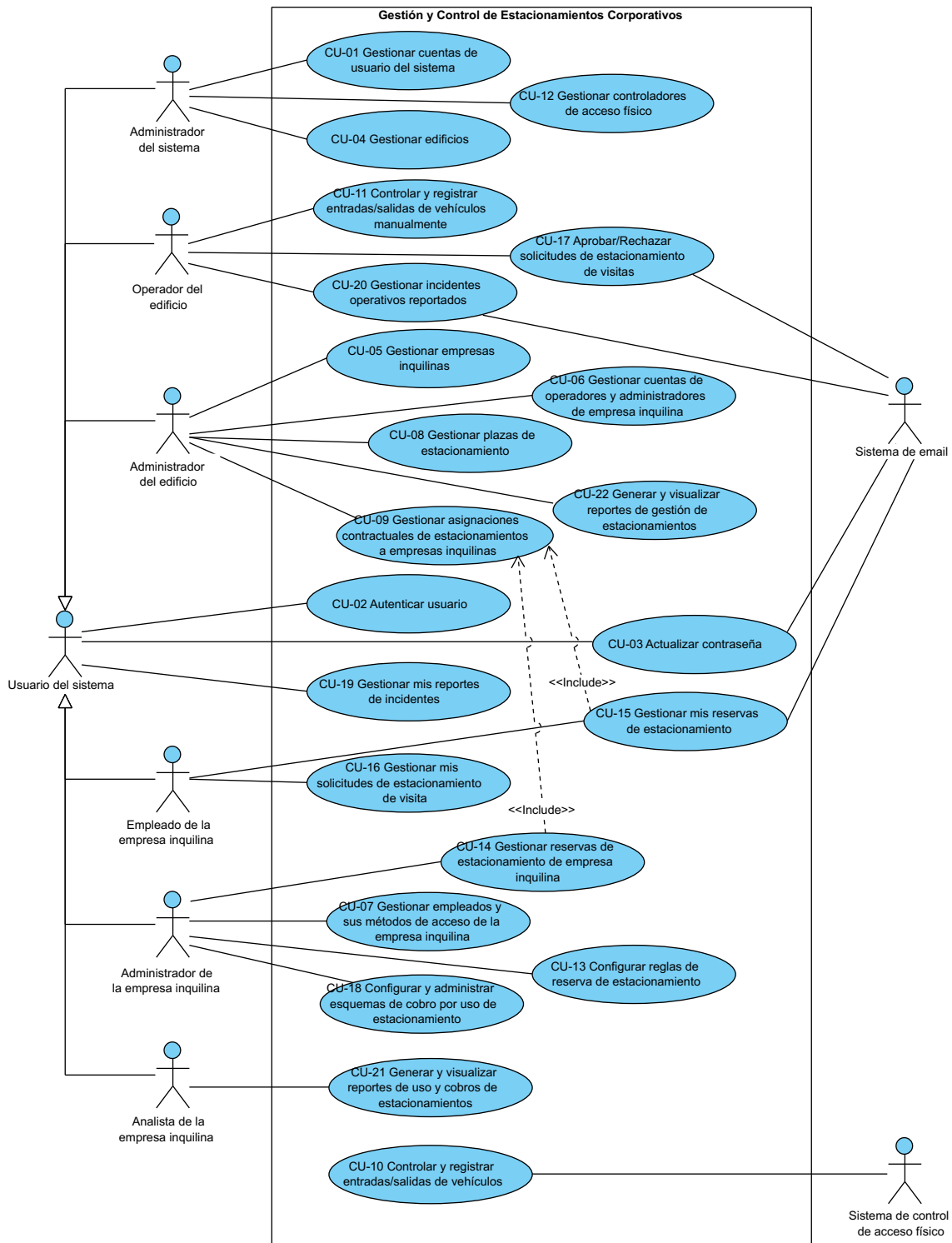


Figura 2-5. Diagrama de casos de uso general

Fuente: Elaboración propia

2.5 Matriz de trazabilidad

A continuación, se presenta la matriz de trazabilidad, una herramienta fundamental para asegurar que todos los requerimientos funcionales identificados previamente se aborden adecuadamente mediante los casos de uso del sistema. Esta matriz facilita el seguimiento y la verificación de que cada funcionalidad requerida tiene una correspondencia en el diseño del sistema.

Tabla 2-3. Matriz de trazabilidad

	NÚMERO DEL CASO DE USO DEL SISTEMA																					
	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22
RF-01	X																					
RF-02		X																				
RF-03			X																			
RF-04			X																			
RF-05				X																		
RF-06				X																		
RF-07					X																	
RF-08						X																
RF-09							X															
RF-10							X															
RF-11								X														
RF-12									X													
RF-13										X	X											
RF-14												X										
RF-15													X									
RF-16														X								
RF-17															X							
RF-18																X						
RF-19																	X					
RF-20																		X				
RF-21																			X			
RF-22																			X			
RF-23																				X		
RF-24																					X	
RF-25																						X

Fuente: Elaboración propia

2.6 Modelo conceptual

A continuación, se presenta el modelo conceptual que establece la estructura fundamental del sistema, identificando las entidades de negocio clave y las relaciones que existen entre ellas. Este diagrama sirve como el pilar sobre el cual se construirá toda la funcionalidad descrita en los casos de uso.

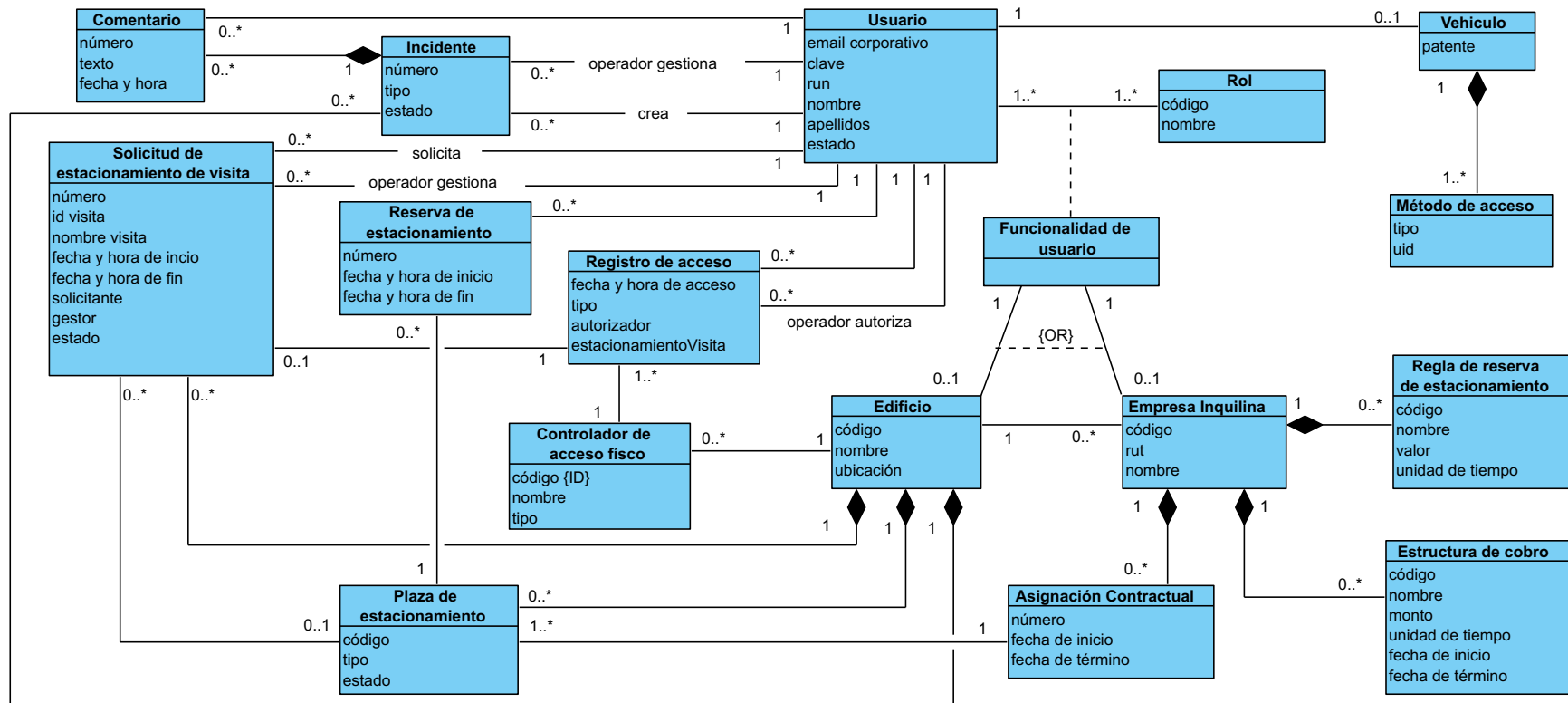


Figura 2-6. Modelo conceptual

Fuente: Elaboración propia

2.7 Modelo de casos de uso

Esta sección descompone cada funcionalidad en un modelo de casos de uso detallado y para mantener el enfoque en los aspectos de mayor impacto, se especifican solo los casos de uso más importantes del sistema.

Cada caso se presenta en un formato narrativo estructurado que describe el flujo principal de eventos, sus alternativas, y las precondiciones y postcondiciones asociadas. Este análisis servirá como la especificación formal del comportamiento del sistema para guiar la fase de diseño.

2.7.1 Caso de uso narrativo CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina

Tabla 2-4. Caso de uso narrativo: CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina

Caso de uso	Gestionar empleados y sus métodos de acceso de la empresa inquilina
Código	CU-07
Resumen	El CU permite al Administrador de la empresa inquilina realizar la gestión integral de las cuentas de usuario para los roles de Administrador de la empresa inquilina, Analista de la empresa inquilina y Empleado de la empresa inquilina, incluyendo la creación de nuevas cuentas, su visualización, modificación y eliminación respectivamente, así como la gestión de sus vehículos y métodos de acceso asociados.
Tipo	Evidente
Referencias cruzadas	No tiene
Actor	Administrador de la empresa inquilina
Precondiciones	El administrador de la empresa inquilina debe estar autenticado en la plataforma tecnológica y debe estar en el contexto de la empresa inquilina donde realizará las gestiones de este CU.
FLUJO NORMAL	
Actor	Sistema
1. El Administrador de la empresa inquilina selecciona la opción “Gestionar cuentas de usuario de la empresa inquilina” en el menú principal.	2. El sistema presenta la interfaz de gestión de cuentas de usuario de la empresa inquilina, la cual contiene: - Una tabla de las cuentas de usuario existentes, con la información de nombre, email, roles y estado. - Opciones de búsqueda para localizar cuentas específicas.

	- Opciones asociadas a cada cuenta en la tabla para visualizar detalles, modificar y eliminar. - Una opción para “Crear Nueva Cuenta”.
3. El Administrador de la empresa inquilina selecciona la opción “Crear Nueva Cuenta”.	4. El sistema muestra un formulario para el ingreso de los datos de la nueva cuenta de usuario. Los datos a ingresar son: RUN, nombre, apellidos, email corporativo, clave inicial, celular y roles (administrador/analista/empleado). Si se selecciona el rol de Empleado, se mostrarán campos adicionales para registrar el vehículo (marca, modelo, color y patente) y los identificadores de acceso asociados a este (tipo y valor).
5. El Administrador de la empresa inquilina completa todos los datos requeridos con la información del nuevo usuario y envía el formulario para crear una nueva cuenta de usuario.	6. El sistema valida los datos ingresados según las reglas definidas: - Todos los datos son obligatorios. - El formato del RUN, email, celular y patente del vehículo deben ser válidos. - La complejidad de la clave inicial debe contener números, letras, símbolos y ser mayor a 8 dígitos. - El rol debe ser válido. - El email corporativo debe ser único en la base de datos. - El valor de los identificadores de acceso debe ser único en el edificio.
	7. El sistema crea la nueva cuenta de usuario en la base de datos y estableciendo el estado inicial (nueva).
	8. El sistema registra la acción de creación de la cuenta en el log de auditoría.
	9. El sistema muestra un mensaje de confirmación indicando que la cuenta de usuario ha sido creada exitosamente.
Postcondición	- Se creó exitosamente en el sistema una nueva instancia de Usuario. - Si uno de los roles del usuario es "empleado", también se crearon las instancias de Vehículo y Métodos de acceso, asociándose correctamente al usuario. - La operación de creación se registró en el log de auditoría del sistema.
FLUJOS ALTERNATIVOS	

Flujo Alternativo		Visualización del detalle específico de una cuenta de usuario existente.	
Actor		Sistema	
2.a.1. El Administrador de la empresa inquilina selecciona la opción “Ver detalles” para una cuenta de usuario específica.		2.a.2. El sistema recupera y muestra toda la información de la cuenta de usuario seleccionada en modo de solo lectura, incluyendo el vehículo e identificadores de acceso si cuenta con el rol de Empleado, junto con la opción para volver a la tabla de gestión de cuentas de usuario.	
2.a.3. El Administrador de la empresa inquilina visualiza la información de la cuenta específica.			
2.a.4. El Administrador de la empresa inquilina selecciona la opción para volver a la tabla de gestión de cuentas de usuario.		2.a.5. El sistema vuelve al paso 2 del flujo normal.	
Flujo Alternativo		Modificación de una cuenta de usuario existente.	
Actor		Sistema	
2.b.1. El Administrador de la empresa inquilina selecciona la opción “Editar” para una cuenta de usuario específica.		2.b.2. El sistema recupera y muestra toda la información de la cuenta de usuario seleccionada en modo de edición para todos los datos (exceptuando el RUN) junto con las opciones para cancelar y actualizar la cuenta de usuario. Si se selecciona el rol de Empleado, se mostrarán campos adicionales para registrar el vehículo (marca, modelo, color y patente) y los identificadores de acceso asociados a este (tipo y valor).	
2.b.3. El Administrador de la empresa inquilina modifica los datos deseados y envía el formulario.		2.b.4. El sistema valida los datos ingresados según las reglas definidas: - Todos los datos son obligatorios. - El formato del email, celular y patente del vehículo deben ser válidos. - La complejidad de la clave debe contener números, letras, símbolos y ser mayor a 8 dígitos. - El rol debe ser válido. - El email corporativo debe ser único en la base de datos. - El valor de los identificadores de acceso debe ser único en el edificio. Si la validación detecta errores: El sistema muestra mensajes descriptivos indicando los	

	errores y permite al Administrador de la empresa inquilina volver al paso 2.b.3. del flujo alternativo.
	2.b.5. El sistema actualiza la información de la cuenta de usuario en la base de datos.
	2.b.6. El sistema registra la acción de modificación de la cuenta en el log de auditoría.
	2.b.7. El sistema muestra un mensaje de confirmación indicando que la cuenta de usuario ha sido actualizada exitosamente.
	2.b.8. El sistema vuelve al paso 2 del flujo normal.
Flujo Alternativo	Eliminación de una cuenta de usuario existente.
Actor	Sistema
2.c.1. El Administrador de la empresa inquilina selecciona la opción “Eliminar” para una cuenta de usuario específica.	2.c.2. El sistema solicita la confirmación al Administrador de la empresa inquilina antes de proceder con la eliminación.
2.c.3. El Administrador de la empresa inquilina confirma la eliminación.	2.c.4. El sistema elimina lógicamente la cuenta de usuario de la base de datos.
	2.c.5. El sistema registra la acción de eliminación de la cuenta en el log de auditoría.
	2.c.6. El sistema muestra un mensaje de confirmación indicando que la cuenta ha sido eliminada exitosamente.
2.c.7. (Alternativa a 2.c.3) El Administrador de la empresa inquilina cancela la eliminación.	2.c.8. (Alternativa a 2.c.3) El sistema cierra el diálogo de confirmación y vuelve al paso 2 del flujo normal.
Flujo Alternativo	El Administrador de la empresa inquilina desiste de la creación de una nueva cuenta.
Actor	Sistema
4.a.1. El Administrador de la empresa inquilina selecciona la opción “Cancelar”.	4.a.2. El sistema descarta cualquier dato ingresado en el formulario y vuelve al paso 2 del flujo normal.
Flujo Alternativo	El sistema detecta que los datos de creación son inválidos
Actor	Sistema
	6.a.1. El sistema muestra mensajes de error específicos indicando qué datos son incorrectos o faltantes.
	6.a.2. El sistema vuelve al punto 4 del flujo normal con los datos ingresados previamente.

Fuente: Elaboración propia

2.7.2 Diagrama de secuencia CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina

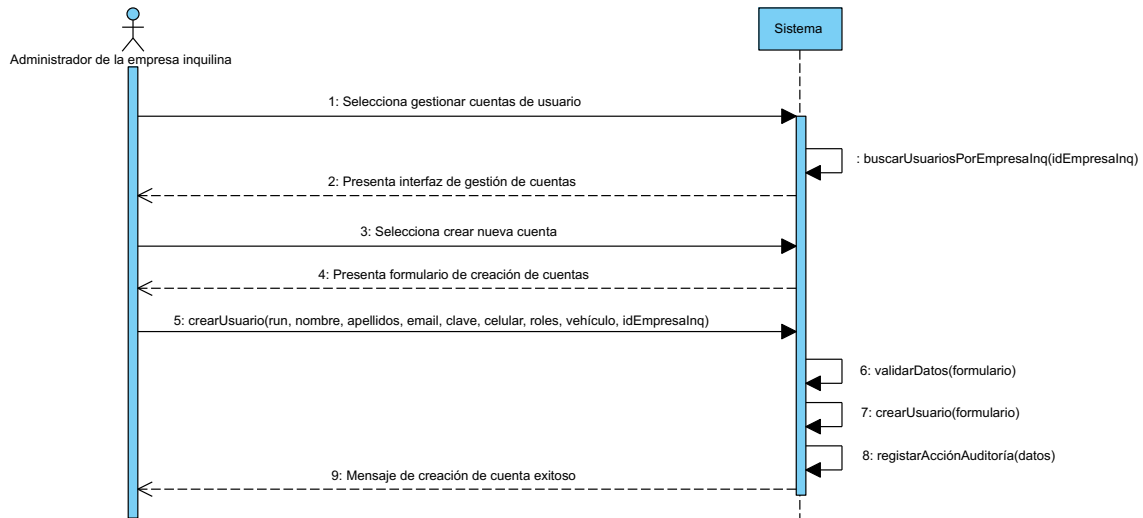


Figura 2-7. Diagrama de secuencia: CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina

Fuente: Elaboración propia

2.7.3 Contratos CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina

Tabla 2-5. Contrato: Crear usuario de la empresa inquilina y sus métodos de acceso en el sistema

Operación	crearUsuario(formulario)
Referencias cruzadas	CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina
Precondiciones	Los datos de la nueva cuenta de usuario son válidos.
Postcondiciones	- Se creó una nueva instancia de la entidad Usuario, Vehículo y Método de acceso junto con sus asociaciones si corresponde.

Fuente: Elaboración propia

2.7.4 Caso de uso narrativo CU-10 Controlar y registrar entradas/salidas de vehículos

Tabla 2-6. Caso de uso narrativo: CU-10 Controlar y registrar entradas/salidas de vehículos

Caso de uso	Controlar y registrar entradas/salidas de vehículos
Código	CU-10
Resumen	El CU describe cómo un sistema de control de acceso físico interactúa con la plataforma mediante una API para autorizar el ingreso o salida de un vehículo, basándose en las reservas existentes.
Tipo	Evidente
Referencias cruzadas	No tiene
Actor	Sistema de control de acceso físico.
Precondiciones	El controlador de acceso físico debe estar previamente registrado en el sistema.
FLUJO NORMAL	
Actor	Sistema
	1. El sistema recibe una solicitud de acceso y verifica que contenga toda la información requerida para procesar la solicitud. La información requerida es: - ID edificio: Identificador del edificio. - APIKEY: Clave única del edificio para acceder al API. - UID dispositivo: Identificador del controlador. - UID método de acceso: Identificador del método de acceso del usuario (ej. patente, código RFID). - Fecha y hora: fecha y hora de la transacción en formato ISO 8601.
	2. El sistema verifica que la APIKEY del edificio sea válida.
	3. El sistema valida que el identificador del dispositivo esté registrado en el edificio correspondiente para asegurar que la petición es legítima y proviene de un controlador registrado en el edificio.
	4. El sistema busca al usuario asociado al identificador del método de acceso.

	5. El sistema verifica si el usuario encontrado tiene una reserva de estacionamiento válida para la fecha y hora de la transacción.
	6. El sistema verifica si la reserva está asociada al edificio correspondiente.
	7. El sistema registra un nuevo evento en el registro de accesos, asociándolo al usuario y controlador de acceso físico.
	8. El sistema responde con un código de respuesta HTTP 200 y el cuerpo de la solicitud en formato JSON indicando que el acceso ha sido autorizado.
Postcondición	- Se registró un nuevo evento en el registro de accesos del sistema. - Se asoció el nuevo registro de acceso con el método de acceso del usuario, la reserva y el controlador de acceso del edificio en el sistema. - El vehículo ingresa o sale del estacionamiento.
FLUJOS ALTERNATIVOS	
Flujo Alternativo	Solicitud incorrecta
Actor	Sistema
	1.a.1. El sistema detecta que la solicitud no contiene toda la información requerida.
	1.a.2. El sistema responde a la solicitud con un código HTTP 400 Bad Request.
Flujo Alternativo	API Key inválida.
Actor	Sistema
	2.a.1. El sistema detecta que la APIKEY del edificio no es válida.
	2.a.2. El sistema responde a la solicitud con un código HTTP 401 Unauthorized.
Flujo Alternativo	Controlador de acceso no encontrado.
Actor	Sistema
	3.a.1. El sistema no encuentra el controlador de acceso en el edificio correspondiente.
	3.a.2. El sistema responde a la solicitud con un código HTTP 404 Not Found.
Flujo Alternativo	Usuario no encontrado.
Actor	Sistema
	4.a.1. El sistema no encuentra un usuario asociado al identificador del método de acceso provisto.
	4.a.2. El sistema responde a la solicitud con un código HTTP 403 Forbidden.
Flujo Alternativo	No existe una reserva válida.
Actor	Sistema

	5.a.1. El sistema no encuentra una reserva de estacionamiento para el usuario con la fecha y hora de la transacción.
	5.a.2. El sistema responde a la solicitud con un código HTTP 403 Forbidden.

Fuente: Elaboración propia

2.7.5 Diagrama de secuencia CU-10 Controlar y registrar entradas/salidas de vehículos

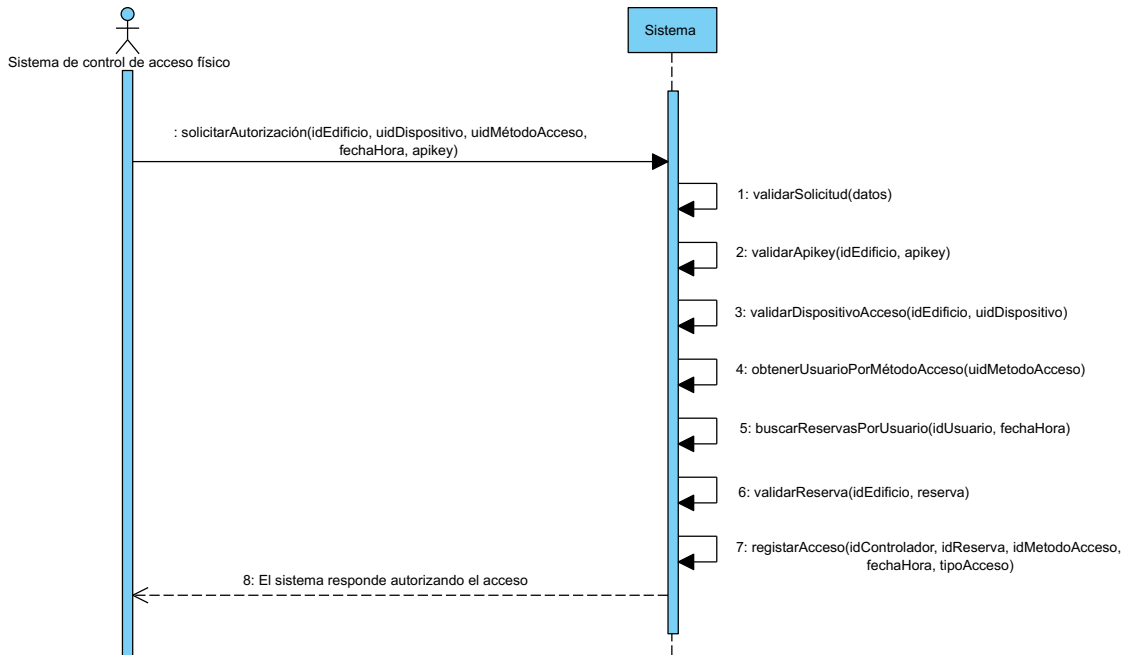


Figura 2-8. Diagrama de secuencia: CU-10 Controlar y registrar entradas/salidas de vehículos

Fuente: Elaboración propia

2.7.6 Contratos CU-10 Controlar y registrar entradas/salidas de vehículos

Tabla 2-7. Contrato: Registrar acceso al estacionamiento

Operación	registrarAcceso(idControlador, idReserva, idMetodoAcceso, fechaHora, tipoAcceso)
Referencias cruzadas	CU-10 Controlar y registrar entradas/salidas de vehículos
Precondiciones	Los datos proporcionados por el sistema de control de acceso son válidos. El usuario del identificador del método de acceso tiene una reserva de estacionamiento válida para la fecha y hora de la transacción.
Postcondiciones	Se creó una nueva instancia de la entidad Registro de acceso y se asoció a las entidades de Controlador de acceso, Reserva de estacionamiento y Método de acceso respectivas.

Fuente: Elaboración propia

2.7.7 Caso de uso narrativo CU-14 Gestionar reservas de estacionamiento de empresa inquilina

Tabla 2-8. Caso de uso narrativo: CU-14 Gestionar reservas de estacionamiento de empresa inquilina

Caso de uso	Gestionar reservas de estacionamiento de empresa inquilina
Código	CU-14
Resumen	El CU permite al Administrador de la empresa inquilina realizar la asignación directa de estacionamientos a empleados específicos (ej. gerentes). Adicionalmente, le permite gestionar (visualizar, editar y eliminar) todas las reservas de estacionamiento activas de la empresa inquilina.
Tipo	Evidente
Referencias cruzadas	CU-09 Gestionar asignaciones contractuales de estacionamientos a empresas inquilinas
Actor	Administrador de la empresa inquilina
Precondiciones	El administrador de la empresa inquilina debe estar autenticado en la plataforma tecnológica y debe estar en el contexto de la empresa inquilina donde realizará las gestiones de este CU. Deben existir plazas de estacionamiento disponibles en el sistema para la empresa inquilina del contexto.
FLUJO NORMAL	
Actor	Sistema
1. El Administrador de la empresa inquilina selecciona la opción “Gestionar reservas de estacionamiento” en el menú principal.	2. El sistema presenta la interfaz de gestión de reservas de estacionamiento, la cual contiene: - Una tabla con todas las reservas vigentes de la empresa inquilina (tanto directas como a

	demanda), con la información del nombre del empleado, email corporativo, código de plaza de estacionamiento, fecha y hora de inicio y fecha y hora de fin para la reserva de estacionamiento. - Opciones de búsqueda para localizar las reservas específicas. - Opciones asociadas a cada reserva en la tabla para visualizar detalles, modificar y finalizar. - Una opción para “Realizar Asignación Directa”.
3. El Administrador del edificio selecciona la opción “Realizar Asignación Directa”.	4. El sistema muestra un formulario para el ingreso de los datos de la reserva de estacionamiento. Los datos a ingresar son: email corporativo del empleado, la plaza de estacionamiento, fecha y hora de inicio y fecha y hora de fin de la reserva de estacionamiento. El sistema obtiene las plazas de estacionamientos asignadas a la empresa inquilina mediante el “CU-09 Gestionar asignaciones contractuales de estacionamientos a empresas inquilinas”.
5. El Administrador del edificio completa todos los datos requeridos y envía el formulario para crear la nueva reserva de estacionamiento.	6. El sistema valida los datos ingresados según las reglas definidas: - Todos los datos son obligatorios. - El empleado debe existir en la empresa inquilina del contexto. - La plaza de estacionamiento seleccionada debe estar disponible durante el periodo de fechas especificado. - La fecha y hora de inicio no puede ser posterior a la fecha y hora de fin.
	7. El sistema crea la nueva reserva de estacionamiento en la base de datos.
	8. El sistema registra la acción de creación de la reserva de estacionamiento de en el log de auditoría.
	9. El sistema notifica mediante email al empleado correspondiente con los datos de la reserva de estacionamiento.
	10. El sistema muestra un mensaje de confirmación indicando que la reserva de

		estacionamiento ha sido creada exitosamente.
Postcondición	- Se creó exitosamente en el sistema una nueva instancia de Reserva de Estacionamiento asociada a un Usuario específico. - La operación se registró en el log de auditoría del sistema.	
FLUJOS ALTERNATIVOS		
Flujo Alternativo	Visualización del detalle específico de una reserva de estacionamiento existente.	
Actor		Sistema
2.a.1. El Administrador de la empresa inquilina selecciona la opción “Ver detalles” para una reserva de estacionamiento específica.		2.a.2. El sistema recupera y muestra toda la información de la reserva de estacionamiento seleccionada en modo de solo lectura junto con la opción para volver a la tabla de gestión de reservas de estacionamiento.
2.a.3. El Administrador de la empresa inquilina visualiza la información de la reserva de estacionamiento específica.		
2.a.4. El Administrador de la empresa inquilina selecciona la opción para volver a la tabla de gestión de reservas de estacionamiento.		2.a.5. El sistema vuelve al paso 2 del flujo normal.
Flujo Alternativo	Modificación de una reserva de estacionamiento existente.	
Actor		Sistema
2.b.1. El Administrador de la empresa inquilina selecciona la opción “Editar” para una reserva de estacionamiento específica.		2.b.2. El sistema recupera y muestra toda la información de la asignación contractual seleccionada en modo de edición solamente para el campo “fecha y hora de fin” junto con las opciones para cancelar y actualizar la reserva de estacionamiento.
2.b.3. El Administrador de la empresa inquilina modifica la fecha y hora de fin y envía el formulario.		2.b.4. El sistema valida los datos ingresados según las reglas definidas: - Todos los datos son obligatorios. - La plaza de estacionamiento seleccionada debe estar disponible durante el periodo de fechas especificado. - La fecha y hora de inicio no puede ser posterior a la fecha y hora de fin. Si la validación detecta errores: El sistema muestra mensajes descriptivos indicando los errores y permite al Administrador de la empresa inquilina volver al paso 2.b.3. del flujo alternativo.
		2.b.5. El sistema actualiza la información de la reserva de estacionamiento en la base de datos.

	2.b.6. El sistema registra la acción de modificación de la reserva de estacionamiento en el log de auditoría.
	2.b.7. El sistema muestra un mensaje de confirmación indicando que la reserva de estacionamiento ha sido actualizada exitosamente y notifica mediante email al empleado correspondiente.
	2.b.8. El sistema vuelve al paso 2 del flujo normal.
Flujo Alternativo	Eliminación de una reserva de estacionamiento existente.
Actor	Sistema
2.c.1. El Administrador de la empresa inquilina selecciona la opción “Eliminar” para una reserva de estacionamiento específica.	2.c.2. El sistema solicita la confirmación al Administrador de la empresa inquilina antes de proceder con la eliminación.
2.c.3. El Administrador de la empresa inquilina confirma la eliminación.	2.c.4. El sistema elimina lógicamente la reserva de estacionamiento de la base de datos.
	2.c.5. El sistema registra la acción de eliminación de la reserva de estacionamiento en el log de auditoría.
	2.c.6. El sistema muestra un mensaje de confirmación indicando que la reserva de estacionamiento ha sido eliminada exitosamente.
2.c.7. (Alternativa a 2.c.3) El Administrador de la empresa inquilina cancela la eliminación.	2.c.8. (Alternativa a 2.c.3) El sistema cierra el diálogo de confirmación y vuelve al paso 2 del flujo normal.
Flujo Alternativo	El Administrador de la empresa inquilina desiste de la creación de una nueva reserva de estacionamiento.
Actor	Sistema
4.a.1. El Administrador de la empresa inquilina selecciona la opción “Cancelar”.	4.a.2. El sistema descarta cualquier dato ingresado en el formulario y vuelve al paso 2 del flujo normal.
Flujo Alternativo	El sistema detecta que los datos de creación son inválidos
Actor	Sistema
	6.a.1. El sistema muestra mensajes de error específicos indicando qué datos son incorrectos o faltantes.
	6.a.2. El sistema vuelve al punto 4 del flujo normal con los datos ingresados previamente.

Fuente: Elaboración propia

2.7.8 Diagrama de secuencia CU-14 Gestionar reservas de estacionamiento de empresa inquilina

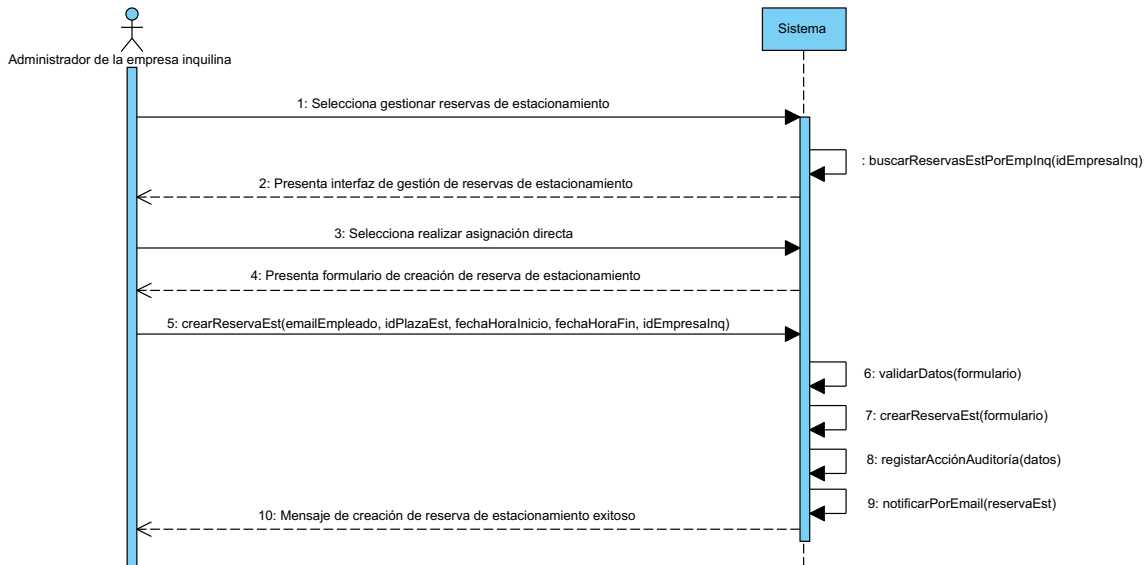


Figura 2-9. Diagrama de secuencia: CU-14 Gestionar reservas de estacionamiento de empresa inquilina

Fuente: Elaboración propia

2.7.9 Contratos CU-14 Gestionar reservas de estacionamiento de empresa inquilina

Tabla 2-9. Contrato: Crear reserva de estacionamiento en el sistema

Operación	crearReservaEst(formulario)
Referencias cruzadas	CU-14 Gestionar reservas de estacionamiento de empresa inquilinas.
Precondiciones	Los datos de la reserva de estacionamiento son válidos.
Postcondiciones	- Se creó una nueva instancia de la entidad Reserva de Estacionamiento asociada a una Plaza de estacionamiento y Usuario respectivamente.

Fuente: Elaboración propia

2.7.10 Caso de uso narrativo CU-15 Gestionar mis reservas de estacionamiento

Tabla 2-10. Caso de uso narrativo: CU-15 Gestionar mis reservas de estacionamiento

Caso de uso	Gestionar mis reservas de estacionamiento
Código	CU-15
Resumen	El CU permite al Empleado de la empresa inquilina realizar la gestión de sus reservas de estacionamiento. Esto incluye la creación de nuevas reservas, cumpliendo con las reglas de negocio establecidas por su empresa, así como la visualización y eliminación de sus reservas existentes.
Tipo	Evidente
Referencias cruzadas	- CU-09 Gestionar asignaciones contractuales de estacionamientos a empresas inquilinas.
Actor	Empleado de la empresa inquilina
Precondiciones	El empleado de la empresa inquilina debe estar autenticado en la plataforma tecnológica.
FLUJO NORMAL	
Actor	Sistema
1. El Empleado de la empresa inquilina selecciona la opción “Gestionar mis reservas de estacionamiento” en el menú principal.	2. El sistema presenta la interfaz de gestión de mis reservas de estacionamiento, la cual contiene: - Una tabla con todas las reservas vigentes del empleado, con la información de código de plaza de estacionamiento, fecha y hora de inicio, y fecha y hora de fin para la reserva de estacionamiento. - Opciones de búsqueda para localizar las reservas específicas. - Opciones asociadas a cada reserva en la tabla para visualizar detalles y eliminar. - Una opción para “Realizar Reserva de Estacionamiento”.
3. El Empleado de la empresa inquilina selecciona la opción “Realizar Reserva de Estacionamiento”.	4. El sistema presenta un formulario solicitando la fecha y hora de inicio de la reserva.
5. El Empleado de la empresa inquilina ingresa la fecha y hora de inicio y envía el formulario.	6. El sistema valida que la fecha y hora de inicio cumpla con la regla de antelación para la reserva de estacionamiento configurada para la empresa inquilina.
	7. El sistema obtiene las plazas de estacionamientos asignadas a la empresa inquilina del Empleado mediante el “ CU-09 Gestionar asignaciones contractuales de estacionamientos a empresas inquilinas ”.

	8. El sistema muestra una lista de las plazas de estacionamiento disponibles. La disponibilidad se calcula considerando la fecha y hora de inicio ingresada más la duración de reserva de estacionamiento configurada para la empresa inquilina (calculando así la fecha y hora de término).
9. El empleado de la empresa inquilina selecciona una de las plazas de estacionamiento disponibles y envía la solicitud para crear la reserva.	10. El sistema valida que la plaza de estacionamiento seleccionada esté disponible en el horario solicitado.
	11. El sistema valida que la solicitud de reserva cumpla con todas las reglas de reserva de estacionamiento definidas para la empresa inquilina (ej. límite de reservas por período, etc.).
	12. El sistema crea la nueva reserva de estacionamiento en la base de datos.
	13. El sistema registra la acción de creación de la reserva de estacionamiento en el log de auditoría.
	14. El sistema notifica el empleado mediante email con los detalles de la reserva.
	15. El sistema muestra un mensaje de confirmación indicando que la reserva ha sido creada exitosamente.
Postcondición	- Se creó y almacenó exitosamente en el sistema una nueva instancia de Reserva de Estacionamiento. - La operación de creación fue registrada en el log de auditoría del sistema. - Se notificó al usuario por email la Reserva de Estacionamiento registrada en el sistema.
FLUJOS ALTERNATIVOS	
Flujo Alternativo	Visualización del detalle específico de una reserva de estacionamiento existente.
Actor	Sistema
2.a.1. El Empleado de la empresa inquilina selecciona la opción “Ver detalles” para una reserva de estacionamiento específica.	2.a.2. El sistema recupera y muestra toda la información de la reserva de estacionamiento seleccionada en modo de solo lectura junto con la opción para volver a la tabla de gestión de mis reservas de estacionamiento.
2.a.3. El Empleado de la empresa inquilina visualiza la información de la reserva de estacionamiento específica.	

2.a.4. El Empleado de la empresa inquilina selecciona la opción para volver a la tabla de gestión de mis reservas de estacionamiento.	2.a.5. El sistema vuelve al paso 2 del flujo normal.
Flujo Alternativo	Eliminación de una reserva de estacionamiento existente.
Actor	Sistema
2.b.1. El Empleado de la empresa inquilina selecciona la opción “Eliminar” para una reserva de estacionamiento específica.	2.b.2. El sistema solicita la confirmación al Empleado de la empresa inquilina antes de proceder con la eliminación.
2.b.3. El Empleado de la empresa inquilina confirma la eliminación.	2.b.4. El sistema elimina lógicamente la reserva de estacionamiento de la base de datos.
	2.b.5. El sistema registra la acción de eliminación de la reserva de estacionamiento en el log de auditoría.
	2.b.6. El sistema muestra un mensaje de confirmación indicando que la reserva de estacionamiento ha sido eliminada exitosamente.
2.b.7. (Alternativa a 2.b.3) El Empleado de la empresa inquilina cancela la eliminación.	2.b.8. (Alternativa a 2.b.3) El sistema cierra el diálogo de confirmación y vuelve al paso 2 del flujo normal.
Flujo Alternativo	Regla de antelación de reserva de estacionamiento no configurada.
Actor	Sistema
	6.a.1. Al momento de validar, el sistema detecta que la regla de antelación no existe para la empresa inquilina
	6.a.2. El sistema omite esta validación específica y vuelve al paso 7 del flujo normal.
Flujo Alternativo	La fecha y hora de inicio no cumple con la regla de antelación.
Actor	Sistema
	6.b.1. El sistema detecta que la fecha y hora de inicio no cumple con la antelación mínima requerida.
	6.b.2. El sistema muestra un mensaje de error indicando la regla infringida y no permite continuar.
	6.b.3. El sistema vuelve al paso 4 del flujo normal.
Flujo Alternativo	La empresa inquilina no tiene plazas de estacionamientos asignadas
Actor	Sistema
	7.a.1. El sistema no encuentra plazas de estacionamiento asignadas a la empresa inquilina.

	7.a.2. El sistema muestra un mensaje indicando que la empresa inquilina no tiene plazas de estacionamiento asignadas.
	7.a.3. El caso de uso finaliza.
Flujo Alternativo	Regla de duración de reserva de estacionamiento no configurada
Actor	Sistema
	8.a.1. El sistema verifica si existe la regla de duración de reserva de estacionamiento para la empresa inquilina.
	8.a.2. Si la regla de duración no existe, el sistema no continúa el proceso.
	8.a.3. El sistema muestra un mensaje informando que la funcionalidad no está disponible por falta de configuración y que debe contactar al administrador de su empresa.
	8.a.4. El caso de uso finaliza.
Flujo Alternativo	No hay plazas de estacionamiento disponibles
Actor	Sistema
	8.b.1. El sistema no encuentra plazas de estacionamiento disponibles para el periodo solicitado.
	8.b.2. El sistema muestra un mensaje indicando que no hay disponibilidad de plazas de estacionamiento para el periodo solicitado.
	8.b.3. El sistema vuelve al paso 4 del flujo normal.
Flujo Alternativo	El Empleado de la empresa inquilina desiste de la creación de una nueva reserva de estacionamiento.
Actor	Sistema
9.a.1. El Empleado de la empresa inquilina selecciona la opción “Cancelar”.	9.a.2. El sistema descarta cualquier dato ingresado en el formulario y vuelve al paso 2 del flujo normal.
Flujo Alternativo	La plaza de estacionamiento seleccionada no se encuentra disponible
Actor	Sistema
	10.a.1. El sistema detecta que la plaza de estacionamiento seleccionada no se encuentra disponible.
	10.a.2. El sistema muestra un mensaje de error indicando que la plaza de estacionamiento seleccionada no se encuentra disponible.
	10.a.3. El sistema vuelve al paso 7 del flujo normal.

Flujo Alternativo	La solicitud final no cumple con las reglas de reserva de estacionamiento.
Actor	Sistema
	11.a.1. El sistema detecta que la solicitud infringe una o más reglas de reserva de estacionamiento de la empresa inquilina.
	11.a.2. El sistema muestra un mensaje de error indicando que regla se ha incumplido (ej. “Ha superado el límite de reservas diarias”).
	11.a.3. El sistema vuelve al paso 4 del flujo normal.

Fuente: Elaboración propia

2.7.11 Diagrama de secuencia CU-15 Gestionar mis reservas de estacionamiento

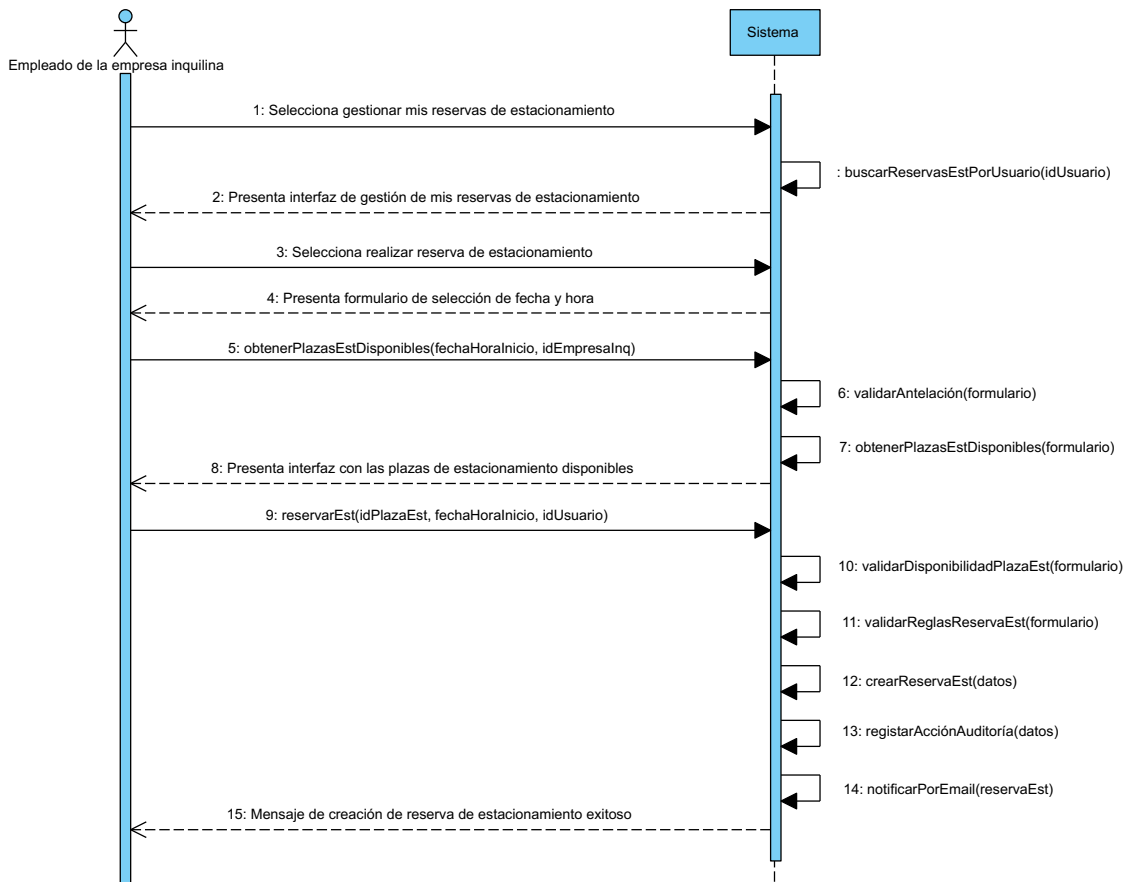


Figura 2-10. Diagrama de secuencia: CU-15 Gestionar mis reservas de estacionamiento

Fuente: Elaboración propia

2.7.12 Contratos CU-15 Gestionar mis reservas de estacionamiento

Tabla 2-11. Contrato: Crear reserva de estacionamiento a demanda en el sistema

Operación	crearReservaEst(datos)
Referencias cruzadas	CU-15 Gestionar mis reservas de estacionamiento.
Precondiciones	Los datos de la reserva de estacionamiento son válidos y cumplen con las reglas de reserva de estacionamiento configuradas por la empresa inquilina.
Postcondiciones	- Se creó una instancia de Reserva de Estacionamiento. - Se asoció la instancia de Reserva de Estacionamiento con la plaza de estacionamiento seleccionada y con el usuario que solicitó la reserva de estacionamiento.

Fuente: Elaboración propia

2.7.13 Caso de uso narrativo CU-16 Gestionar mis solicitudes de estacionamiento de visitas

Tabla 2-12. Caso de uso narrativo: CU-16 Gestionar mis solicitudes de estacionamiento de visitas

Caso de uso	Gestionar mis solicitudes de estacionamiento de visita
Código	CU-16
Resumen	El CU permite al Empleado de la empresa inquilina realizar la gestión de sus solicitudes de estacionamiento de visita. Esto incluye la creación de nuevas solicitudes, así como la visualización y eliminación de sus solicitudes existentes.
Tipo	Evidente
Referencias cruzadas	No tiene
Actor	Empleado de la empresa inquilina
Precondiciones	El empleado de la empresa inquilina debe estar autenticado en la plataforma tecnológica.

FLUJO NORMAL

Actor	Sistema
1. El Empleado de la empresa inquilina selecciona la opción “Gestionar mis solicitudes de estacionamiento de visita” en el menú principal.	2. El sistema presenta la interfaz de gestión de solicitudes de estacionamiento de visita, la cual contiene: - Una tabla con todas las solicitudes vigentes del empleado, con la información de número de solicitud, fecha y hora de inicio, hora de fin y estado para la solicitud. - Opciones de búsqueda para localizar las solicitudes específicas. - Opciones asociadas a cada solicitud en la tabla para visualizar detalles y eliminar.

	- Una opción para “Solicitar Estacionamiento de Visita”.
3. El Empleado de la empresa inquilina selecciona la opción “Solicitar Estacionamiento de Visita”.	4. El sistema presenta un formulario solicitando un identificador oficial (RUN, Pasaporte, otro), motivo visita (reunión, proveedor, evento, otro), nombre, apellidos, email, fecha y hora desde, hora hasta, patente del vehículo y comentarios.
5. El Empleado de la empresa inquilina completa todos los datos requeridos con la información del visitante y envía el formulario para crear una nueva solicitud de estacionamiento de visita.	6. El sistema valida los datos ingresados según las reglas definidas: - Todos los datos son obligatorios a excepción del campo comentarios. - El formato del identificador, patente del vehículo y email deben ser válidos. - La fecha y hora desde no puede ser anterior a la fecha actual.
	7. El sistema crea la nueva solicitud de estacionamiento de visita en la base de datos con el estado “En revisión” vinculada al usuario solicitante y al edificio al cual pertenece la empresa inquilina del usuario solicitante.
	8. El sistema registra la acción de creación de la solicitud de estacionamiento de visita en el log de auditoría.
	9. El sistema notifica a los operadores del edificio mediante email con los detalles de la solicitud.
	10. El sistema muestra un mensaje de confirmación indicando que la solicitud de estacionamiento de visita ha sido realizada exitosamente.
Postcondición	- Se creó exitosamente en el sistema una nueva instancia de Solicitud de estacionamiento de visita asociada al Usuario solicitante y al Edificio respectivo. - La operación se registró en el log de auditoría del sistema. - Se notificó a los operadores del edificio correspondiente.
FLUJOS ALTERNATIVOS	
Flujo Alternativo	Visualización del detalle específico de una solicitud de estacionamiento de visita existente.
Actor	Sistema
2.a.1. El Empleado de la empresa inquilina selecciona la opción “Ver detalles” para una solicitud de estacionamiento de visita específica.	2.a.2. El sistema recupera y muestra toda la información de la solicitud de estacionamiento de visita seleccionada en modo de solo lectura junto con la opción para

	volver a la tabla de gestión de mis solicitudes de estacionamiento de visita.
2.a.3. El Empleado de la empresa inquilina visualiza la información de la solicitud de estacionamiento de visita específica.	
2.a.4. El Empleado de la empresa inquilina selecciona la opción para volver a la tabla de gestión de mis solicitudes de estacionamiento de visita.	2.a.5. El sistema vuelve al paso 2 del flujo normal.
Flujo Alternativo	Eliminación de una solicitud de estacionamiento de visita existente con estado “En revisión” o “Rechazada”.
Actor	Sistema
2.b.1. El Empleado de la empresa inquilina selecciona la opción “Eliminar” para una solicitud de estacionamiento de visita específica.	2.b.2. El sistema solicita la confirmación al Empleado de la empresa inquilina antes de proceder con la eliminación.
2.b.3. El Empleado de la empresa inquilina confirma la eliminación.	2.b.4. El sistema elimina lógicamente la solicitud de estacionamiento de visita de la base de datos.
	2.b.5. El sistema registra la acción de eliminación de la solicitud de estacionamiento de visita en el log de auditoría.
	2.b.6. El sistema muestra un mensaje de confirmación indicando que la solicitud de estacionamiento de visita ha sido eliminada exitosamente.
2.b.7. (Alternativa a 2.b.3) El Empleado de la empresa inquilina cancela la eliminación.	2.b.8. (Alternativa a 2.b.3) El sistema cierra el diálogo de confirmación y vuelve al paso 2 del flujo normal.
Flujo Alternativo	Eliminación de una solicitud de estacionamiento de visita existente con estado “Aprobada”.
Actor	Sistema
2.c.1. El Empleado de la empresa inquilina selecciona la opción “Eliminar” para una solicitud de estacionamiento de visita específica.	2.c.2. El sistema solicita la confirmación al Empleado de la empresa inquilina antes de proceder con la eliminación.
2.c.3. El Empleado de la empresa inquilina confirma la eliminación.	2.c.4. El sistema elimina lógicamente la solicitud de estacionamiento de visita de la base de datos.
	2.c.5. El sistema registra la acción de eliminación de la solicitud de estacionamiento de visita en el log de auditoría.

	2.c.6. El sistema notifica mediante email al visitante que su reserva de estacionamiento de visita ha sido eliminada por el anfitrión.
	2.c.7. El sistema muestra un mensaje de confirmación indicando que la solicitud de estacionamiento de visita ha sido eliminada exitosamente.
2.c.8. (Alternativa a 2.c.3) El Empleado de la empresa inquilina cancela la eliminación.	2.c.9. (Alternativa a 2.c.3) El sistema cierra el diálogo de confirmación y vuelve al paso 2 del flujo normal.
Flujo Alternativo	El Empleado de la empresa inquilina desiste de la creación de una nueva solicitud de estacionamiento de visita.
Actor	Sistema
5.a.1. El Empleado de la empresa inquilina selecciona la opción “Cancelar”.	5.a.2. El sistema descarta cualquier dato ingresado en el formulario y vuelve al paso 2 del flujo normal.
Flujo Alternativo	El sistema detecta que los datos de la solicitud son inválidos
Actor	Sistema
	6.a.1. El sistema muestra mensajes de error específicos indicando qué datos son incorrectos o faltantes.
	6.a.2. El sistema vuelve al punto 4 del flujo normal con los datos ingresados previamente.

Fuente: Elaboración propia

2.7.14 Diagrama de secuencia CU-16 Gestionar mis solicitudes de estacionamiento de visita

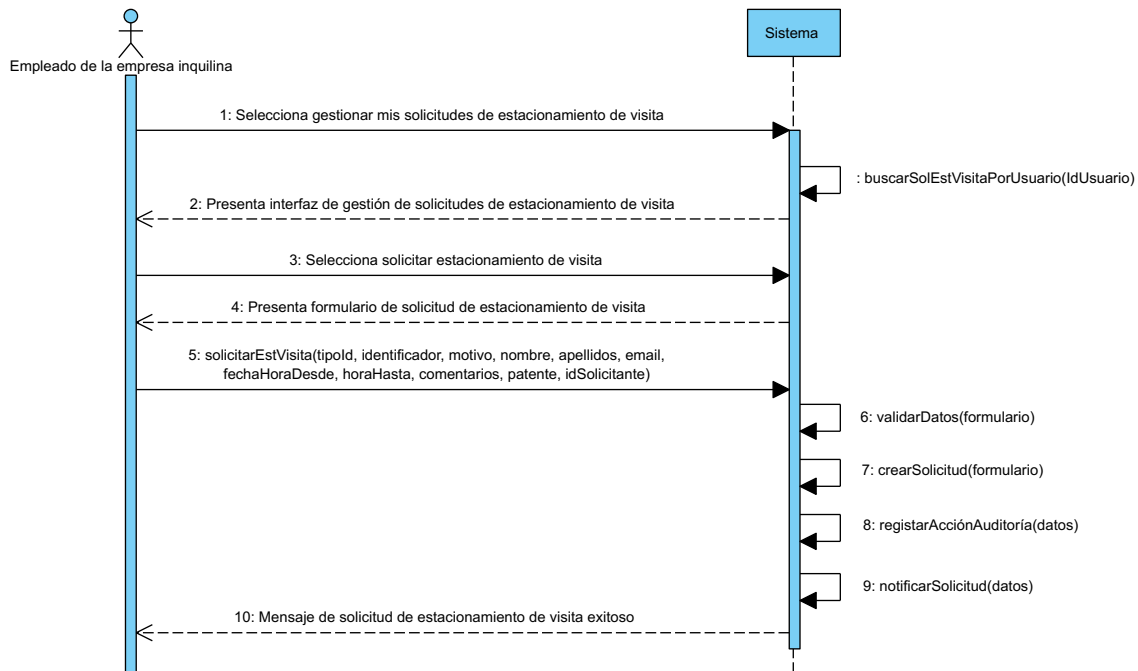


Figura 2-11. Diagrama de secuencia: CU-16 Gestionar mis solicitudes de estacionamiento de visita

Fuente: Elaboración propia

2.7.15 Contratos CU-16 Gestionar mis solicitudes de estacionamiento de visita

Tabla 2-13. Contrato: Solicitar estacionamiento de visita en el sistema

Operación	crearSolicitud(formulario)
Referencias cruzadas	CU-16 Gestionar mis solicitudes de estacionamiento de visita
Precondiciones	Los datos de la nueva solicitud de estacionamiento de visita son válidos.
Postcondiciones	- Se creó una nueva instancia de la entidad Solicitud de estacionamiento de visita asociada al Usuario solicitante y al Edificio respectivo.

Fuente: Elaboración propia

2.7.16 Caso de uso narrativo CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita

Tabla 2-14. Caso de uso narrativo: CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita

Caso de uso	Aprobar/Rechazar solicitudes de estacionamiento de visita.	
Código	CU-17	
Resumen	El CU permite al Operador del edificio gestionar las solicitudes de estacionamiento de visita que se encuentran en estado "En revisión". El operador puede aprobar una solicitud asignando una plaza de estacionamiento disponible, rechazarla indicando un motivo, o mantenerla en revisión para una gestión posterior.	
Tipo	Evidente	
Referencias cruzadas		
Actor	Operador del edificio	
Precondiciones	- El Operador del edificio debe estar autenticado en la plataforma tecnológica y debe estar en el contexto del edificio donde realizará las gestiones de este CU. - Debe existir una solicitud de reserva de estacionamiento de visita vigente en estado “En revisión”.	
FLUJO NORMAL		
Actor	Sistema	
1. El Operador del edificio selecciona la opción “Gestionar solicitudes de estacionamiento de visita” en el menú principal.	2. El sistema presenta la interfaz de gestión de solicitudes de estacionamiento de visita, la cual contiene: - Una tabla con todas las solicitudes vigentes del edificio, con la información de número de solicitud, nombre del solicitante, fecha y hora de inicio y fin solicitadas y estado para la solicitud. - Opciones de búsqueda para localizar las solicitudes específicas. - Opciones asociadas a cada solicitud en la tabla para gestionarla.	
3. El Operador del edificio selecciona una solicitud específica para gestionarla.	4. El sistema muestra los detalles completo de la solicitud y, basándose en el horario solicitado, busca y muestra una lista de las plazas de estacionamiento para visitas que se encuentran disponibles en ese rango de tiempo.	
5. El Operador del edificio selecciona una de las plazas de estacionamiento disponible y elige la opción “Aprobar Solicitud”.	6. El sistema valida que la plaza de estacionamiento seleccionada esté disponible en el rango horario de la solicitud.	

	7. El sistema actualiza la solicitud de estacionamiento de visita asociándola a la plaza de estacionamiento seleccionada y cambiando el estado a “Aprobada”.
	8. El sistema registra la acción de actualización en el log de auditoría.
	9. El sistema envía una notificación mediante correo electrónico tanto al usuario solicitante como al visitante, informando la aprobación y los detalles de la plaza de estacionamiento asignada.
	10. El sistema muestra un mensaje de confirmación indicando que la solicitud ha sido aprobada exitosamente y vuelve a la interfaz de gestión de solicitudes.
Postcondición	- El estado de la solicitud se actualizó a “Aprobada”. - La solicitud quedó asociada a una plaza de estacionamiento para visitas. - Se notificó al usuario solicitante y visitante de la aprobación de la solicitud. - La operación se registró en el log de auditoría del sistema.
FLUJOS ALTERNATIVOS	
Flujo Alternativo	El Operador del edificio rechaza la solicitud de estacionamiento de visita.
Actor	Sistema
5.a.1. El Operador del edificio, tras revisar la solicitud, elige la opción “Rechazar Solicitud” (ej. no hay plazas de estacionamiento disponibles).	5.a.2. El sistema muestra un campo para que el Operador ingrese el motivo del rechazo.
5.a.3. El Operador del edificio ingresa el motivo y confirma el rechazo.	5.a.4. El sistema actualiza el estado de la solicitud a “Rechazada”.
	5.a.5. El sistema registra la acción actualización en el log de auditoría.
	5.a.6. El sistema envía una notificación por correo electrónico al usuario solicitante, informando del rechazo de la solicitud y el motivo.
	5.a.7. El sistema muestra un mensaje de confirmación indicando que la solicitud ha sido rechazada exitosamente y vuelve a la interfaz de gestión de solicitudes.
Flujo Alternativo	El Operador del edificio opta por mantener la solicitud en el estado “En revisión”.
Actor	Sistema
5.b.1. El Operador del edificio, tras revisar la solicitud, elige la opción “Volver”.	5.b.2. El sistema vuelve a la interfaz de gestión de solicitudes.

Flujo Alternativo	La plaza de estacionamiento seleccionada no se encuentra disponible
Actor	Sistema
	6.a.1. El sistema detecta que la plaza de estacionamiento seleccionada no se encuentra disponible.
	6.a.2. El sistema muestra un mensaje indicando que la plaza de estacionamiento seleccionada no se encuentra disponible y vuelve al paso 4 del flujo normal.

Fuente: Elaboración propia

2.7.17 Diagrama de secuencia CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita

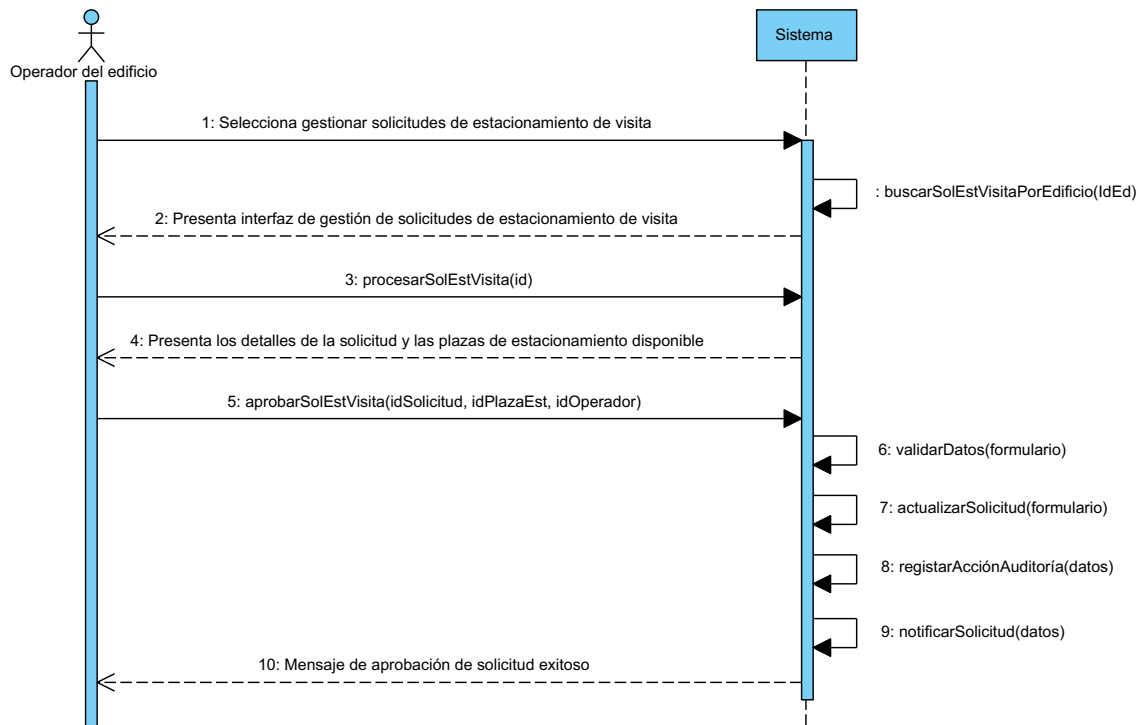


Figura 2-12. Diagrama de secuencia: CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita

Fuente: Elaboración propia

2.7.18 Contratos CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita

Tabla 2-15. Contrato: Aprobar/Rechazar solicitud de estacionamiento en el sistema

Operación	actualizarSolicitud(formulario)
Referencias cruzadas	CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita
Precondiciones	La plaza de estacionamiento seleccionada está disponible.
Postcondiciones	- Se actualizó el estado de la instancia de la entidad Solicitud de estacionamiento de visita a “Aprobada”. - Se asoció la Solicitud de estacionamiento de visita con la Plaza de estacionamiento seleccionada.

Fuente: Elaboración propia

2.7.19 Caso de uso narrativo CU-19 Gestionar mis reportes de incidentes

Tabla 2-16. Caso de uso narrativo: CU-19 Gestionar mis reportes de incidentes

Caso de uso	Gestionar mis reportes de incidentes
Código	CU-19
Resumen	El CU permite al Usuario del sistema registrar nuevos incidentes operativos relacionados con el uso de los estacionamientos. Adicionalmente, le permite visualizar el estado y detalle de todos los incidentes que ha reportado y añadir comentarios a los mismos para facilitar la comunicación con el Operador del edificio.
Tipo	Evidente
Referencias cruzadas	No tiene
Actor	Usuario del sistema
Precondiciones	El Usuario del sistema debe estar autenticado en la plataforma tecnológica.
FLUJO NORMAL	
Actor	Sistema
1. El Usuario del sistema selecciona la opción “Gestionar mis reportes de incidentes” en el menú principal.	2. El sistema presenta la interfaz de gestión de mis reportes de incidentes, la cual contiene: - Una tabla con todos los incidentes reportados por el usuario, con la información de número de incidente, tipo, fecha y estado. - Opciones para localizar incidentes específicos. - Opciones asociadas a cada incidente en la tabla para visualizar el detalle. - Una opción para “Reportar Nuevo Incidente”.

3. El Usuario del sistema selecciona la opción “Reportar Nuevo Incidente”.	4. El sistema muestra un formulario para el ingreso de los datos del nuevo incidente. Los datos a ingresar son: tipo de incidente (ej. ocupación indebida, problemas de acceso, daños, robos), descripción detallada del incidente, y la opción de adjuntar evidencia (fotos).
5. El Usuario del sistema completa todos los datos solicitados con la información del incidente y envía el formulario para crear un nuevo reporte.	6. El sistema valida los datos ingresados según las reglas definidas: - El tipo de incidente y la descripción es obligatoria. - La evidencia en caso de adjuntarse debe cumplir con lo siguiente: a. Los formatos admitidos son jpg, jpeg y png. b. El tamaño de cada archivo no puede superar los 3 MB de tamaño.
	7. El sistema crea el nuevo incidente en la base de datos, asociándolo al usuario y al edificio correspondiente, con un estado inicial de “Reportado”.
	8. El sistema registra la acción de creación del incidente en el log de auditoría.
	9. El sistema notifica mediante correo electrónico a los operadores del edificio sobre el nuevo incidente reportado.
	10. El sistema muestra un mensaje de confirmación indicando que el incidente ha sido reportado exitosamente junto con el número asignado al incidente.
Postcondición	- Se creó exitosamente en el sistema una nueva instancia de Incidente asociada al Usuario y Edificio respectivo. - La operación se registró en el log de auditoría del sistema. - Se notificó a los operadores del edificio de un nuevo Incidente.
FLUJOS ALTERNATIVOS	
Flujo Alternativo	Visualización del detalle específico de un incidente reportado.
Actor	Sistema
2.a.1. El Usuario del sistema selecciona la opción “Ver detalles” para un incidente específico.	2.a.2. El sistema recupera y muestra toda la información del incidente seleccionado en modo de solo lectura, incluyendo los comentarios junto con las opciones para volver a la tabla de gestión de mis reportes de incidentes y añadir un nuevo comentario.
2.a.3. El Usuario del sistema visualiza la información del reporte de incidente específico.	

2.a.4. El Usuario del sistema selecciona la opción para volver a la tabla de gestión de mis reportes de incidentes.		2.a.5. El sistema vuelve al paso 2 del flujo normal.	
Flujo Alternativo		Añadir un comentario a un incidente específico.	
Actor		Sistema	
2.b.1. El Usuario del sistema selecciona la opción “Ver detalles” para un incidente específico.		2.b.2. El sistema recupera y muestra toda la información del incidente seleccionado en modo de solo lectura, incluyendo los comentarios junto con las opciones para volver a la tabla de gestión de mis reportes de incidentes y añadir un nuevo comentario.	
2.b.3. El Usuario del sistema visualiza la información del reporte de incidente específico.			
2.b.4. El Usuario del sistema selecciona la opción para añadir un nuevo comentario.		2.b.5. El sistema presenta un campo de texto para que el usuario ingrese su comentario.	
		2.b.6. El sistema guarda el comentario en la base de datos, asociándolo al incidente y usuario.	
		2.b.7. El sistema registra la acción de creación del comentario en el log de auditoría.	
		2.b.8. El sistema notifica mediante correo electrónico a los operadores del edificio.	
		2.b.9. El sistema actualiza la vista de detalles del incidente con el nuevo comentario y muestra un mensaje de confirmación.	
Flujo Alternativo		El Usuario del sistema desiste de la creación de un nuevo reporte de incidente.	
Actor		Sistema	
4.a.1. El Usuario del sistema selecciona la opción “Cancelar”.		4.a.2. El sistema descarta cualquier dato ingresado en el formulario y vuelve al paso 2 del flujo normal.	
Flujo Alternativo		El sistema detecta que los datos de creación del incidente son inválidos.	
Actor		Sistema	
		6.a.1. El sistema muestra mensajes de error específicos indicando qué datos son incorrectos o faltantes.	
		6.a.2. El sistema vuelve al punto 4 del flujo normal con los datos ingresados previamente.	

Fuente: Elaboración propia

2.7.20 Diagrama de secuencia CU-19 Gestionar mis reportes de incidentes

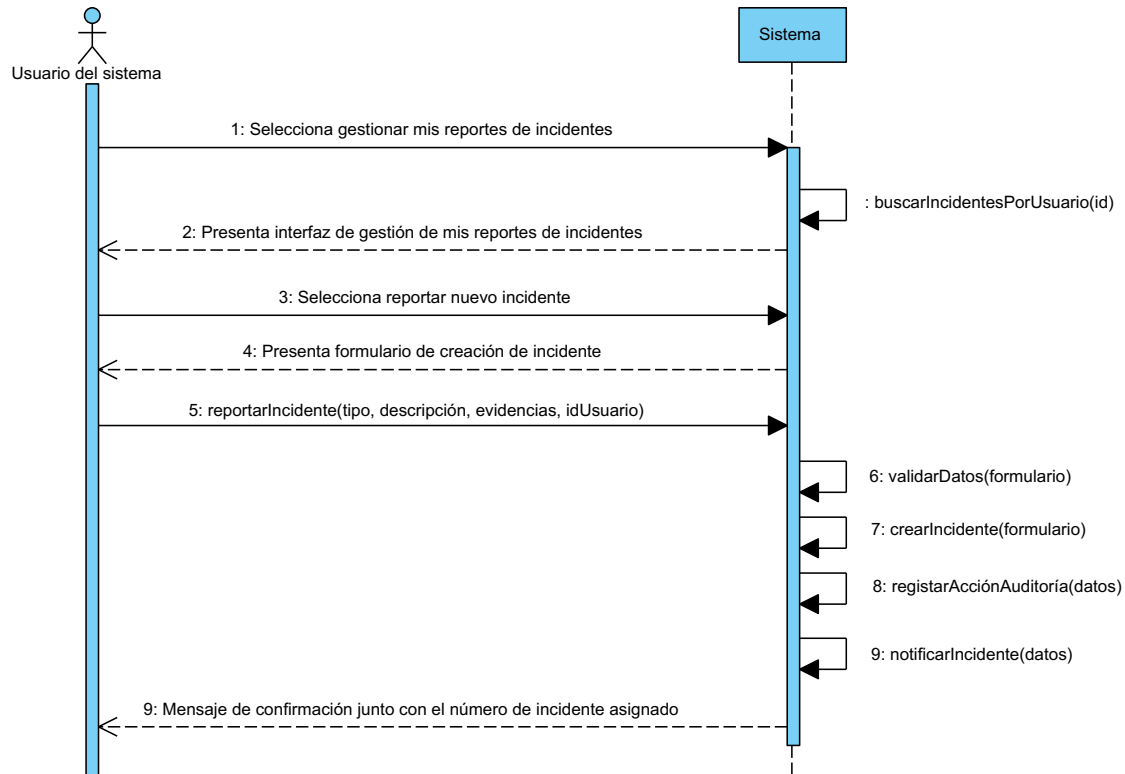


Figura 2-13. Diagrama de secuencia: CU-19 Gestionar mis reportes de incidentes

Fuente: Elaboración propia

2.7.21 Contratos CU-19 Gestionar mis reportes de incidentes

Tabla 2-17. Contrato: Reportar incidente en el sistema

Operación	crearIncidente(formulario)
Referencias cruzadas	CU-19 Gestionar mis reportes de incidentes
Precondiciones	Los datos del nuevo incidente son válidos.
Postcondiciones	- Se creó una nueva instancia de la entidad Incidente asociada al Usuario y Edificio respectivo.

Fuente: Elaboración propia

2.7.22 Caso de uso narrativo CU-20 Gestionar incidentes operativos reportados

Tabla 2-18. Caso de uso narrativo: CU-20 Gestionar incidentes operativos reportados

Caso de uso	Gestionar incidentes operativos reportados
Código	CU-20
Resumen	El CU permite al Operador del edificio gestionar todos los incidentes operativos que han sido reportados por los usuarios en su edificio. Puede visualizar la información de cada incidente, actualizar su estado (ej. “En Progreso”, “Resuelto”), y añadir comentarios para mantener un registro de las acciones realizadas y facilitar la comunicación con el usuario que reportó el incidente.
Tipo	Evidente
Referencias cruzadas	No tiene
Actor	Operador del edificio
Precondiciones	- El Operador del edificio debe estar autenticado en la plataforma tecnológica y debe estar en el contexto del edificio donde realizará las gestiones de este CU. - Debe existir un incidente reportado en estado “Reportado” para el edificio del contexto.
FLUJO NORMAL	
Actor	Sistema
1. El Operador del edificio selecciona la opción “Gestionar incidentes reportados” en el menú principal.	2. El sistema presenta la interfaz de gestión de incidentes reportados, la cual contiene: - Una tabla con todos los incidentes reportados en el edificio, con la información de número de incidente, tipo, fecha, estado, usuario que reportó el incidente y operador que lo está gestionando. - Opciones para localizar incidentes específicos. - Opciones asociadas a cada incidente en la tabla para visualizar el detalle.
3. El Operador del edificio selecciona un incidente específico con estado “Reportado” para gestionarlo.	4. El sistema recupera y muestra toda la información del incidente seleccionado, incluyendo los comentarios y las opciones para cambiar el estado y volver a la interfaz de gestión de incidentes.
5. El Operador del edificio, tras revisar la información, selecciona la opción para cambiar el estado del incidente a “En Progreso”.	6. El sistema valida que el incidente seleccionado se encuentre en el estado “Reportado”.
	7. El sistema actualiza el estado del incidente en la base de datos a “En Progreso”, asociándolo al operador del edificio.

	8. El sistema registra la acción de actualización del incidente en el log de auditoría.
	9. El sistema envía una notificación mediante correo electrónico al usuario que reportó el incidente, informándole que su incidente está siendo gestionado por un operador del edificio.
	10. El sistema muestra un mensaje de confirmación indicando que el incidente ha sido actualizado exitosamente.
Postcondición	- El estado del Incidente se actualizó a “En Progreso” en el sistema. - El Incidente se asoció al Usuario del operador del edificio como gestor del incidente. - La operación se registró en el log de auditoría del sistema. - Se notificó al autor del incidente que está siendo gestionado por un operador del edificio.
FLUJOS ALTERNATIVOS	
Flujo Alternativo	Añadir un comentario a un incidente específico
Actor	Sistema
2.a.1. El Operador del edificio selecciona un incidente específico con estado “En Progreso” o “Resuelto” para añadir un comentario.	2.a.2. El sistema recupera y muestra toda la información del incidente seleccionado, incluyendo los comentarios y las opciones para cambiar el estado, añadir un comentario y volver a la interfaz de gestión de incidentes.
2.a.3. El Operador del edificio añade un nuevo comentario detallando las acciones que se están tomando (ej. “Se necesita mayor información”, “Se ha contactado al propietario del vehículo infractor”).	2.a.4. El sistema crea un nuevo comentario en la base de datos, asociándolo al incidente y operador del edificio.
	2.a.5. El sistema registra la acción de creación del comentario en el log de auditoría.
	2.a.6. El sistema envía una notificación mediante correo electrónico al usuario que reportó el incidente con el nuevo comentario del operador del edificio.
Flujo Alternativo	Registra el incidente cómo resuelto
Actor	Sistema
2.b.1. El Operador del edificio selecciona un incidente específico con estado “En Progreso”.	2.b.2. El sistema recupera y muestra toda la información del incidente seleccionado, incluyendo los comentarios y las opciones para cambiar el estado, añadir un comentario y volver a la interfaz de gestión de incidentes.

2.b.3. El Operador del edificio actualiza el estado a “Resuelto”.	2.b.4. El sistema presenta un campo de texto para que el operador ingrese un comentario detallando la resolución del incidente.
2.b.5. El Operador del edificio ingresa la información solicitada y confirma la resolución del incidente.	2.b.6. El sistema actualiza el estado del incidente a “Resuelto” y añade un nuevo comentario en la base de datos, asociándolo al incidente y operador del edificio.
	2.b.7. El sistema registra la acción de actualización del incidente en el log de auditoría.
	2.b.8. El sistema envía una notificación mediante correo electrónico al usuario que reportó el incidente, informándole que su incidente está resuelto junto con el comentario final del operador.
	2.b.9. El sistema muestra un mensaje de confirmación indicando que el incidente ha sido registrado como resuelto exitosamente.

Fuente: Elaboración propia

2.7.23 Diagrama de secuencia CU-20 Gestionar incidentes operativos reportados

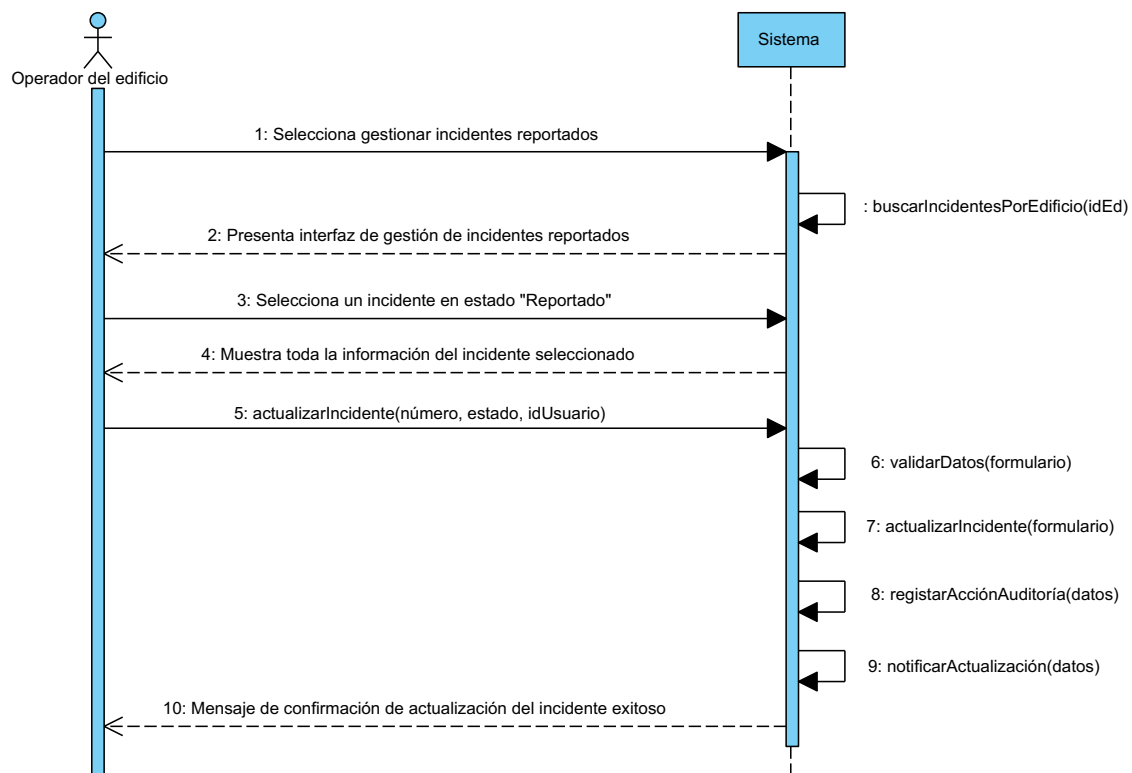


Figura 2-14. Diagrama de secuencia: CU-20 Gestionar incidentes operativos reportados

Fuente: Elaboración propia

2.7.24 Contratos CU-20 Gestionar incidentes operativos reportados

Tabla 2-19. Contrato: Actualizar el estado de un incidente reportado en el sistema

Operación	actualizarIncidente(formulario)
Referencias cruzadas	CU-20 Gestionar incidentes operativos reportados
Precondiciones	El incidente tiene el estado “Reportado”.
Postcondiciones	- El estado del Incidente se actualizó a “En Progreso”. - El Incidente se asoció al Usuario del operador del edificio como gestor del incidente.

Fuente: Elaboración propia

**CAPÍTULO 3. ASPECTOS RELEVANTES DEL DISEÑO DE LA ALTERNATIVA
SELECCIONADA**

3 ASPECTOS RELEVANTES DEL DISEÑO

Este capítulo se enfoca en los aspectos relevantes del diseño para el sistema de gestión y control de estacionamientos. A lo largo de sus secciones, se traducirá el análisis de requerimientos del capítulo anterior en un plano técnico detallado para la construcción de la solución.

La implementación se basará en Angular como framework de desarrollo web para el frontend, Quarkus para el backend y PostgreSQL como sistema de gestión de base de datos, con un despliegue sobre una infraestructura de contenedores en Kubernetes tanto para el frontend como el backend.

3.1 Arquitectura del software

Para el diseño de la solución, se ha seleccionado un estilo de Arquitectura Hexagonal, también conocida como “Puertos y Adaptadores”. Esta elección se fundamenta en su capacidad para desacoplar el núcleo de la lógica de la aplicación de los componentes externos, como la interfaz de usuario, la base de datos o servicios de terceros. Esto permite que el dominio del negocio permanezca aislado de las decisiones tecnológicas, facilitando las pruebas, el mantenimiento y la evolución del sistema a largo plazo.

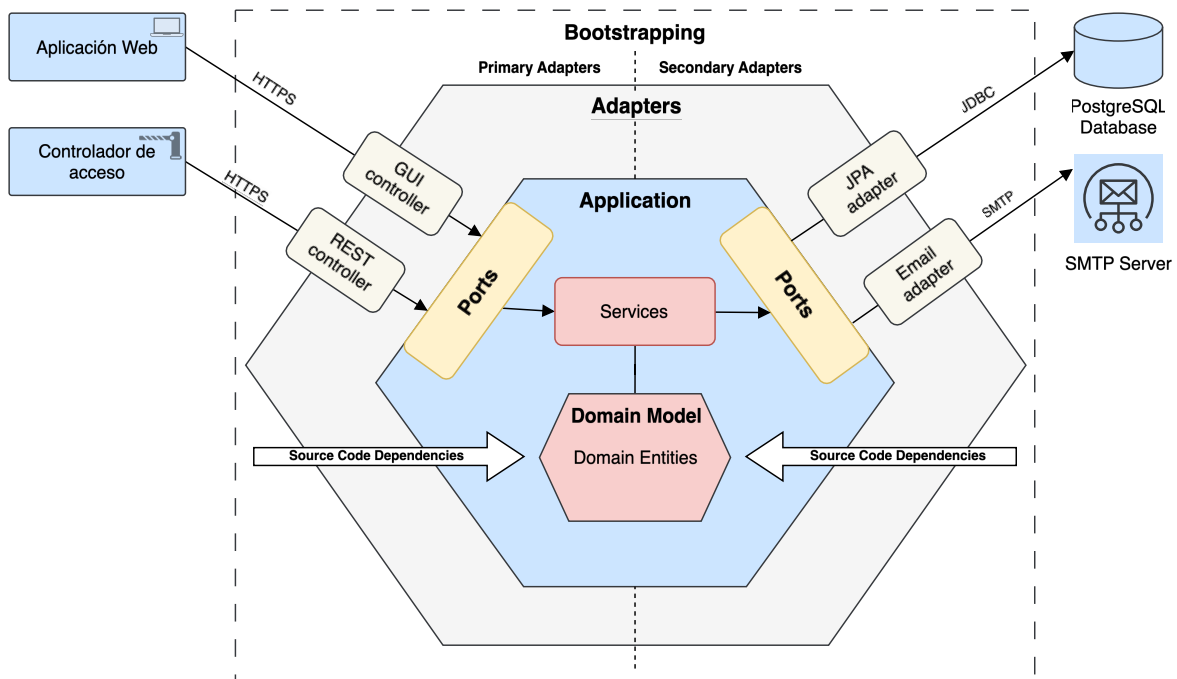


Figura 3-1. Diagrama de arquitectura del sistema – vista general

Fuente: Elaboración propia

La modularización de la arquitectura se logrará mediante el uso de Maven con submódulos, lo que reforzará la separación de responsabilidades entre los distintos módulos del sistema. La estructura del backend se organizará en los siguientes cuatro módulos:

1. **Domain Model:** Este módulo es el corazón del dominio. Contiene el modelo de negocio, es decir, las clases que representan los conceptos clave del sistema, como Usuario, Vehículo, Reserva de estacionamiento y Empresa Inquilina, entre otros.
2. **Application:** Este módulo contiene los servicios del dominio que implementan los casos de uso de la aplicación. Define los “puertos” (Interfaces de Java) a través de los cuales la lógica de negocio se comunica con el mundo exterior. Junto con el módulo “domain model”, conforma el núcleo de la aplicación.
3. **Adapter:** Este módulo agrupa las implementaciones de los puertos definidos en el módulo de “application”. Incluye adaptadores primarios, responsables de recibir solicitudes externas en invocar los casos de uso, y adaptadores secundarios, encargados de integrar el núcleo con infraestructuras y servicios externos.
4. **Bootstrapping:** Este es el módulo no contiene lógica de negocio y actúa como punto de ensamblaje y arranque de la aplicación. Su responsabilidad es instanciar e inyectar las dependencias de los adaptadores en los puertos del núcleo de la aplicación y poner en marcha el servidor web de Quarkus.

3.1.1 Arquitectura lógica del sistema

La vista lógica detalla la organización interna del sistema y muestra cómo los módulos Domain Model, Application y Adapter interactúan mediante puertos y adaptadores, respetando el principio de dependencia hacia el núcleo de la aplicación.

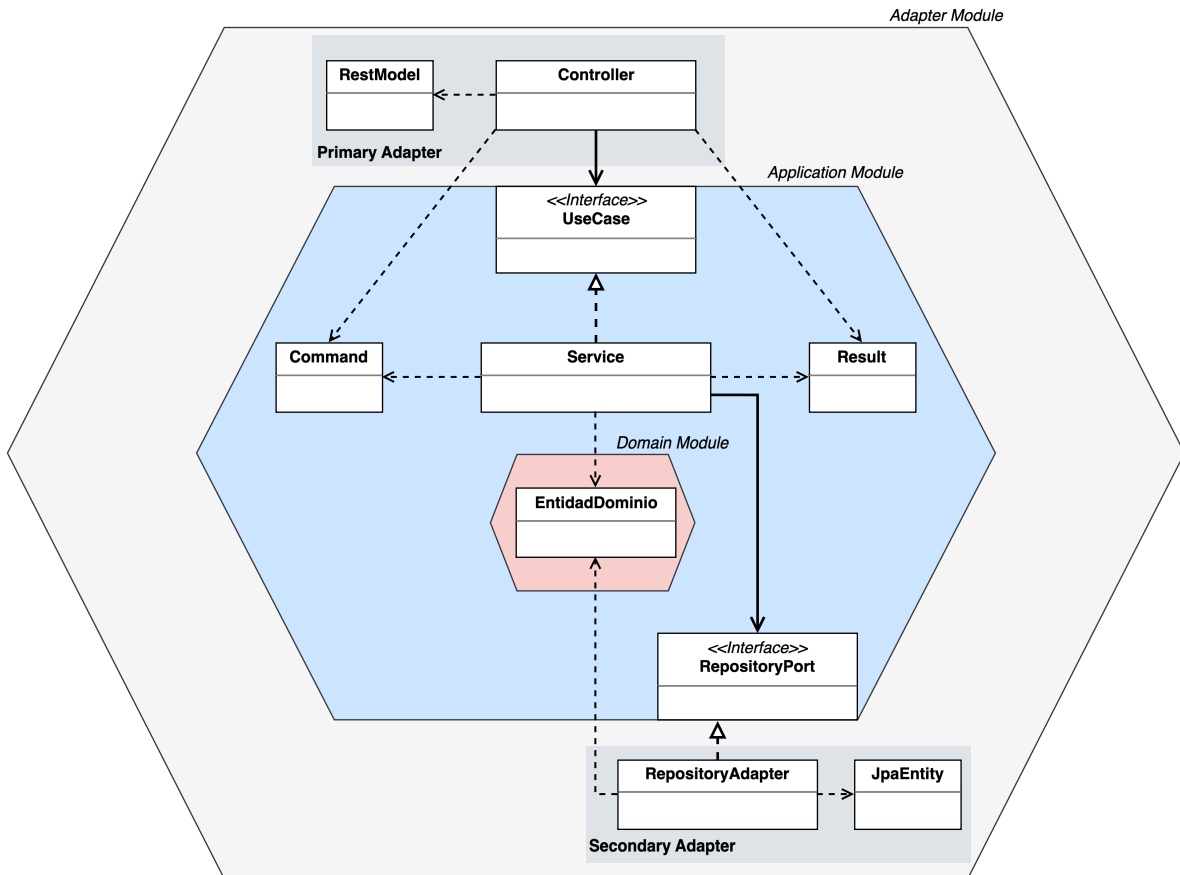


Figura 3-2. Diagrama de arquitectura del sistema - vista lógica

Fuente: Elaboración propia

A continuación, se describen los diferentes tipos de clases que serán parte del diseño de la solución.

Tabla 3-1. Arquitectura Hexagonal - tipos de clases

TIPO	PROPÓSITO	USO	ALCANCE
RestModel	Representar el modelo de datos específico para la comunicación HTTP dentro del adaptador REST.	Utilizado exclusivamente por el controlador REST para enviar y recibir datos hacia y desde el cliente HTTP.	Limitado al adaptador REST, actuando como un DTO entre la interfaz del usuario y el dominio. No tiene lógicas de negocio ni validaciones profundas.
Controller	Gestionar las solicitudes HTTP entrantes, realizar la validación básica de los datos recibidos y delegar la ejecución al UseCase correspondiente.	Convierte los objetos RestModel en Command para invocar al UseCase y transforma los Result devueltos por el módulo de aplicación en RestModel de respuesta.	Adaptador REST que interactúa directamente con los clientes externos (navegador) y mantiene una dependencia hacia el dominio mediante el UseCase.
UseCase	Representar la puerta de entrada del dominio, exponiendo un contrato de operaciones claramente definidas.	Agrupar todas las operaciones que pueden realizarse sobre una entidad del dominio (crear, obtener, actualizar, eliminar, etc.).	Situado en el módulo de aplicación, define la frontera hacia el exterior del dominio y se comunica con los adaptadores de entrada.
Service	Implementar las interfaces UseCase con la lógica de negocio correspondiente, orquestando las interacciones entre las entidades del dominio y puertos de salida.	Procesar los objetos Command, ejecutar las reglas de negocio, validar datos, gestionar transacciones y delegar operaciones a puertos de salida.	Pertenece al módulo de aplicación, Actúa como implementación concreta de los casos de uso (UseCase) definidos.
Command	Encapsular los datos necesarios para ejecutar operaciones específicas en el dominio, como creación o actualización de entidades.	Se crea a partir del RestModel recibido por el adaptador REST y se entrega al UseCase para su procesamiento.	Es utilizado dentro del módulo de aplicación como mecanismo para entregar datos estructurados al UseCase.
Result	Representar el resultado de un Caso de Uso, encapsulando los datos mínimos que el CU necesita entregar al exterior, sin filtrar detalles internos del dominio.	Se crea en el UseCase y se entrega al adaptador REST para que lo transforme a RestModel y responda al cliente HTTP.	Es utilizado dentro del módulo de aplicación. No contiene anotaciones REST/JPA ni lógica de negocio profunda; solo la estructura del resultado del CU.
Entidad Dominio	Representar un concepto clave del negocio con lógica e invariantes internas encapsuladas, manteniendo la integridad de los datos y reglas internas del negocio.	Se utiliza en el dominio para representar y manipular objetos con reglas de negocio que deben cumplirse siempre.	Exclusivamente dentro del dominio. No tiene conocimientos de infraestructura técnica ni dependencias externas.
Repository Port	Definir un contrato para la persistencia de entidades del dominio, desacoplando el módulo de aplicación del módulo de adaptadores.	Proporciona operaciones para interactuar con el almacenamiento persistente.	Ubicado en módulo de aplicación, actuando como un límite abstracto que debe implementarse en el módulo de adaptadores.

Repository Adapter	Implementar las interfaces RepositoryPort mediante una tecnología específica.	Gestionar la persistencia real de los objetos del dominio en una base de datos o sistema externo.	Pertenece exclusivamente al módulo de adaptadores. Su dependencia apunta hacia el dominio mediante la interfaz RepositoryPort.
JpaEntity	Representar el modelo de datos específico de la tecnología de persistencia utilizada (JPA).	Actuar como una entidad técnica para mapear entidades del dominio en registros almacenados en la base de datos.	Restringido al adaptador de persistencia en el módulo de adaptadores.

Fuente: Elaboración propia

3.1.2 Arquitectura de despliegue del sistema

La vista de despliegue describe la infraestructura sobre la cual se ejecuta el sistema, detallando el uso de un clúster Kubernetes que permite alta disponibilidad, escalabilidad horizontal y separación de responsabilidades entre frontend y backend.

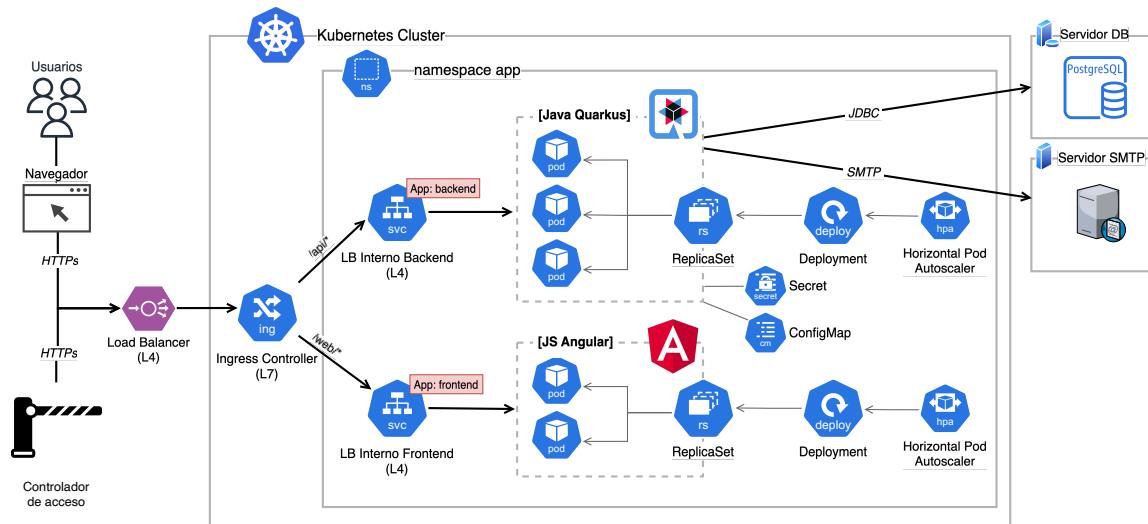


Figura 3-3. Diagrama de arquitectura del sistema - vista de despliegue

Fuente: Elaboración propia

Como se observa en la vista de despliegue del diagrama de arquitectura del sistema, el flujo de interacción es el siguiente:

1. Los usuarios (cómo el administrador del edificio o el empleado de la empresa inquilina) interactúan con la aplicación a través de un navegador web, consumiendo el frontend desarrollado en Angular. Por otro lado, los controladores de acceso físico se comunican directamente con la API del backend.

2. Todo el tráfico de entrada es gestionado por un Ingress Controller de Kubernetes, que actúa como un enrutador, dirigiendo las solicitudes al servicio de frontend o de backend según corresponda.
3. Dentro del clúster, las aplicaciones de frontend y backend se ejecutan en Pods. Kubernetes gestiona la creación y destrucción de estos Pods y puede escalarlos horizontalmente para responder a la carga de trabajo de un momento específico.
4. El backend, desarrollado en Quarkus, contiene toda la lógica de negocio estructurada según la Arquitectura Hexagonal.
5. Para la persistencia de los datos, el backend se comunica con la instancia de la base de datos PostgreSQL, que se encuentra fuera del clúster.

Esta arquitectura garantiza una solución robusta, escalable y mantenible, alineada con las prácticas modernas de desarrollo y despliegue de software.

3.1.3 Tecnologías seleccionadas

La selección tecnológica se fundamenta en su total alineación con el **stack tecnológico corporativo** y en el aprovechamiento del **conocimiento técnico** existente dentro de la organización, lo que permite reducir los riesgos de implementación y optimizar los tiempos de desarrollo. Estas herramientas actúan como los habilitadores técnicos necesarios para materializar eficazmente la arquitectura hexagonal y el despliegue en Kubernetes, garantizando la modularidad, el desacoplamiento de la lógica de negocio y la escalabilidad exigida para la solución de Arikok.

1. **Framework Frontend:** La interfaz de usuario se desarrollará utilizando JavaScript mediante el framework Angular. Esta elección permite la creación de una aplicación web responsiva y de una sola página (SPA), que ofrece una experiencia de usuario fluida y dinámica. Además, Angular facilita la modularización y la organización del código fuente, lo que simplifica su mantenimiento.
2. **Framework Backend:** El backend de la aplicación se implementará con Quarkus, un framework de Java nativo para Kubernetes. Quarkus es especialmente idóneo para operar en un entorno de contenedores gracias a su bajo consumo de memoria y su rápido de tiempo arranque. Estas características son esenciales para aprovechar las capacidades de auto-escalado y eficiencia de recursos que ofrece Kubernetes, asegurando que la aplicación sea resiliente y pueda manejar momentos de alta demanda.

3. **Infraestructura:** La solución informática ha sido diseñada para operar en un entorno moderno, escalable y de alta disponibilidad, aprovechando las ventajas de la contenerización. Esta elección permite aprovechar las capacidades nativas de Kubernetes para el escalamiento horizontal automático, el balanceo de carga y la auto-reparación de los servicios, garantizando que la aplicación pueda manejar momentos de alta demanda y sea resiliente ante fallos.
4. **Sistema de gestión de Base de Datos:** Para la persistencia de los datos, se ha seleccionado PostgreSQL. Esta base de datos relacional es reconocida por su robustez y fiabilidad. En la arquitectura propuesta, la instancia de PostgreSQL se alojará en un servidor externo al clúster de Kubernetes. Esta separación es una práctica recomendada para servicios con estado (stateful), ya que permite una gestión independiente del ciclo de vida de los datos (respaldos, actualizaciones, etc.), lo que proporciona mayor estabilidad y flexibilidad para la administración de los datos del sistema.

3.2 Diseño de Datos

En esta sección se detalla la estructura de datos que dará soporte a la lógica de negocio y a los requerimientos del sistema. El diseño presentado es la materialización técnica del modelo conceptual desarrollado en el capítulo de análisis.

3.2.1 Modelo de Clases

En esta sección se presenta el diseño estático de la solución, con el objetivo de centrar el análisis en la lógica de negocio. Para ello, se introduce un diagrama de clases enfocado exclusivamente en las entidades del dominio y en las relaciones que existen entre ellas.

Este diagrama se abstrae intencionadamente de las clases propias de la arquitectura hexagonal, tales como controladores, servicios o adaptadores, con el fin de ofrecer una visión clara y directa del modelo de negocio que sustenta el sistema.

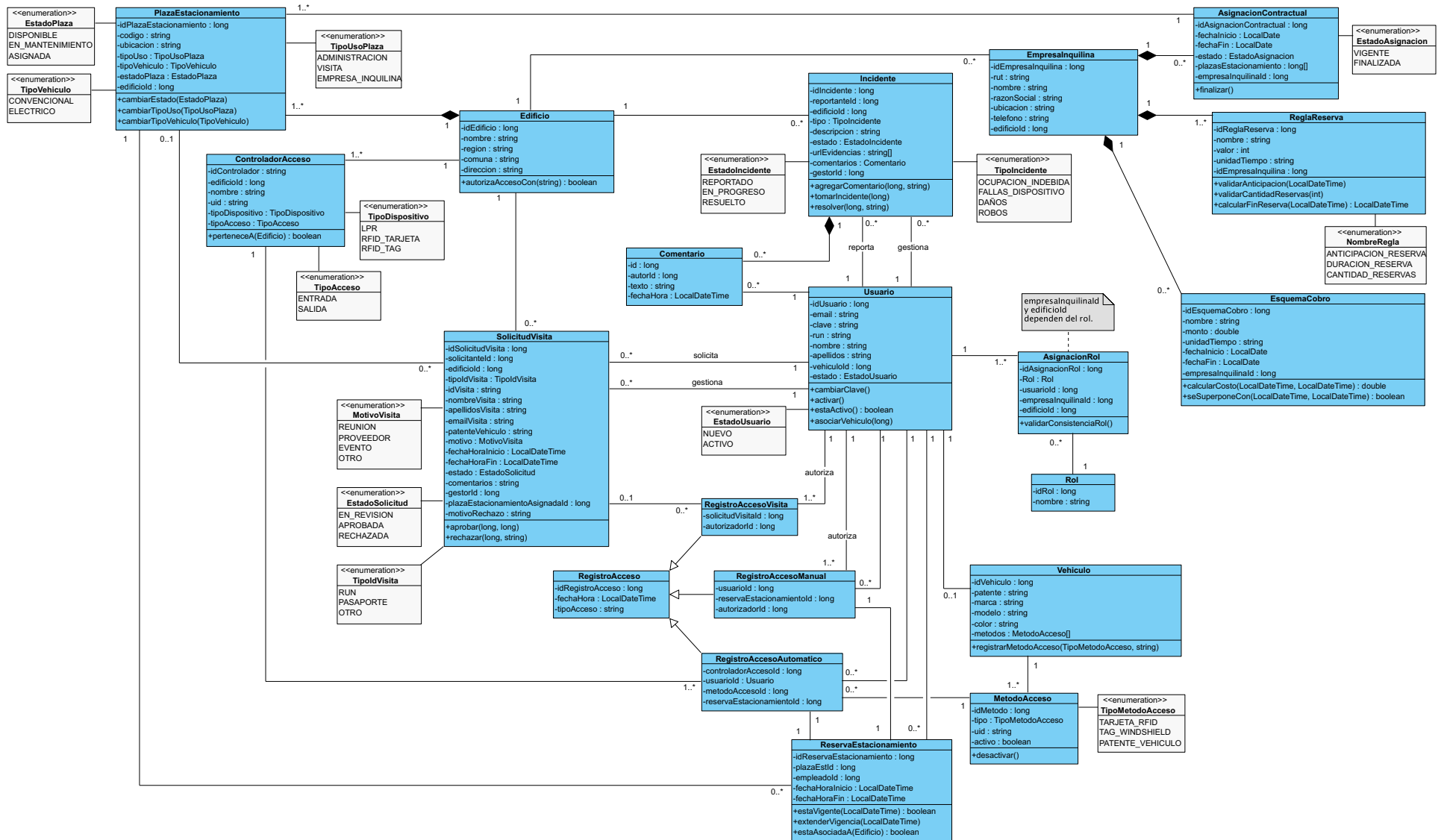


Figura 3-4. Diagrama de clases

Fuente: Elaboración propia

3.2.2 Modelo Relacional

Tras definir la estructura de clases de la aplicación, esta sección presenta el modelo relacional que define la capa de persistencia. Basado en el sistema gestor de base de datos PostgreSQL, el siguiente modelo traduce las entidades de dominio en un esquema de tablas, atributos y relaciones. Su propósito es garantizar la integridad y consistencia de los datos almacenados por el sistema.

Además, el modelo incorpora una estrategia de eliminación lógica para preservar la trazabilidad y facilitar la administración de datos. En lugar de eliminar físicamente los registros, las entidades se marcan mediante el atributo **fecha Eliminacion**, cuyo valor nulo indica vigencia. Esta aproximación permite definir procesos de eliminación periódica para remover de forma definitiva los registros eliminados tras un intervalo configurable, manteniendo la base de datos limpia sin comprometer la integridad referencial.

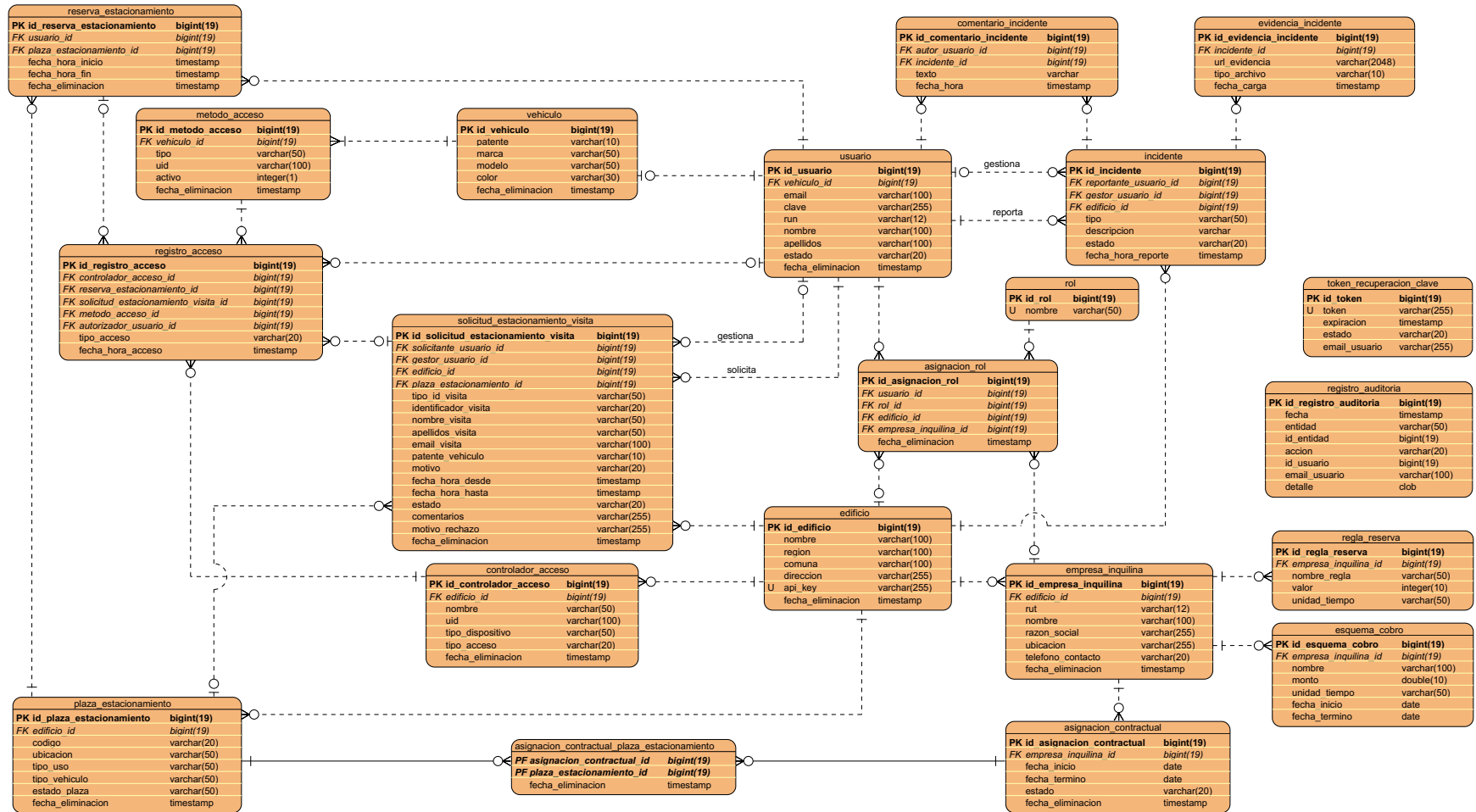


Figura 3-5. Diagrama de Entidad-Relación

Fuente: Elaboración propia

3.2.3 Diccionario de Datos

Como complemento al modelo relacional presentado en la sección anterior, este apartado detalla el diccionario de datos del sistema. Su propósito es ofrecer una descripción de cada uno de los atributos que componen las tablas definidas en el diagrama entidad-relación. Para cada tabla, se especificará el nombre de la columna, su tipo de dato y una descripción funcional, sirviendo como una referencia técnica para la implementación y el mantenimiento futuro de la base de datos.

Tabla 3-2. Diccionario de datos: Asignación contractual

Nombre:	asignacion_contractual	
Descripción:	Define la asignación de un conjunto de plazas de estacionamiento a una empresa inquilina por un periodo contractual.	
Clave Primaria:	id asignacion contractual	
Clave Foránea:	empresa_inquilina_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_asignacion_contractual	bigint(19)	Identificador único de la asignación contractual.
fecha_inicio	date	Fecha de inicio del contrato de asignación.
fecha_termino	date	Fecha de término del contrato de asignación.
estado	varchar(20)	Indica si la asignación está vigente o finalizada.
empresa_inquilina_id	bigint(19)	Empresa inquilina a la que se le asignan las plazas.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-3. Diccionario de datos: Asignación contractual – Plaza de estacionamiento

Nombre:	asignacion contractual plaza estacionamiento	
Descripción:	Tabla de unión que detalla qué plazas específicas están incluidas en una asignación contractual.	
Clave Primaria:	Compuesta: (asignacion_contractual_id, plaza_estacionamiento_id)	
Clave Foránea:	asignacion_contractual_id plaza_estacionamiento_id	
COLUMNA	TIPO	DESCRIPCIÓN
asignacion_contractual_id	bigint(19)	Identificador de la asignación contractual.
plaza_estacionamiento_id	bigint(19)	Identificador de la plaza de estacionamiento incluida.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-4. Diccionario de datos: Asignación - Rol

Nombre:	asignacion_rol	
Descripción:	Tabla que asigna un rol específico a un usuario dentro de un contexto determinado, como un edificio o una empresa inquilina. Permite gestionar permisos de manera granular.	
Clave Primaria:	id asignacion rol	
Clave Foránea:	usuario_id rol_id edificio_id empresa_inquilina_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_asignacion_rol	bigint(19)	Identificador único para cada registro de asignación de rol.
usuario_id	bigint(19)	Identificador del usuario al que se le asigna el rol.
rol_id	bigint(19)	Identificador del rol que se está asignando.
edificio_id	bigint(19)	Identificador del edificio en el que aplica el rol (es opcional).
empresa_inquilina_id	bigint(19)	Identificador de la empresa inquilina en la que aplica el rol (es opcional).
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-5. Diccionario de datos: Comentario de incidente

Nombre:	comentario incidente	
Descripción:	Almacena los comentarios y actualizaciones de seguimiento para un incidente específico.	
Clave Primaria:	id_comentario_incidente	
Clave Foránea:	incidente_id autor_usuario_id (referencia a usuario)	
COLUMNA	TIPO	DESCRIPCIÓN
id_comentario_incidente	bigint(19)	Identificador único del comentario.
texto	varchar	Contenido del comentario.
fecha_hora	timestamp	Fecha y hora del comentario.
Incidente_id	bigint(19)	Incidente al que pertenece el comentario.
autor_usuario_id	bigint(19)	Usuario que realizó el comentario.

Fuente: Elaboración propia

Tabla 3-6. Diccionario de datos: Controlador de acceso

Nombre:	controlador acceso	
Descripción:	Almacena la información de los dispositivos físicos de control de acceso (barreras, lectores de patente).	
Clave Primaria:	id_controlador_acceso	
Clave Foránea:	edificio_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_controlador_acceso	bigint(19)	Identificador único para cada controlador.
nombre	varchar(50)	Nombre descriptivo del controlador (ej: Barrera Acceso Norte).
uid	varchar(100)	Identificador único del dispositivo.
tipo_dispositivo	varchar(50)	Tipo de tecnología del controlador (ej: LPR, RFID).
tipo_acceso	varchar(20)	Indica si el controlador es de 'entrada' o 'salida'.
edificio_id	bigint(19)	Identificador del edificio donde está instalado el controlador.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-7. Diccionario de datos: Edificio

Nombre:	edificio	
Descripción:	Almacena la información de los edificios comerciales que son gestionados por el sistema.	
Clave Primaria:	id_edificio	
Clave Foránea:		
COLUMNA	TIPO	DESCRIPCIÓN
id_edificio	bigint(19)	Identificador único para cada edificio.
nombre	varchar(100)	Nombre del edificio.
direccion	varchar(255)	Dirección física del edificio.
comuna	varchar(100)	Comuna donde se ubica el edificio.
region	varchar(100)	Región donde se ubica el edificio.
api_key	varchar(255)	Clave de API única del edificio para integraciones seguras con sistemas externos.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-8. Diccionario de datos: Empresa inquilina

Nombre:	empresa_inquilina	
Descripción:	Contiene los datos de las empresas que arriendan espacios en los edificios.	
Clave Primaria:	id_empresa_inquilina	
Clave Foránea:	edificio_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_empresa_inquilina	bigint(19)	Identificador único para la empresa inquilina.
rut	varchar(12)	RUT de la empresa.
razon_social	varchar(255)	Razón social legal de la empresa.
nombre	varchar(100)	Nombre de fantasía o comercial de la empresa.
ubicacion	varchar(255)	Ubicación específica dentro del edificio (ej: piso, oficina).
telefono_contacto	varchar(20)	Teléfono de contacto principal de la empresa.
edificio_id	bigint(19)	Identificador del edificio donde se encuentra la empresa.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-9. Diccionario de datos: Esquema de cobro

Nombre:	esquema_cobro	
Descripción:	Define los esquemas de cobro que una empresa inquilina puede aplicar a sus empleados por el uso de estacionamientos.	
Clave Primaria:	id_esquema_cobro	
Clave Foránea:	empresa_inquilina_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_esquema_cobro	bigint(19)	Identificador único del esquema de cobro.
nombre	varchar(100)	Nombre del esquema de cobro.
monto	double(10)	Monto o tarifa a cobrar.
unidad_tiempo	varchar(50)	Periodicidad del cobro (minuto, hora, día).
fecha_inicio	date	Fecha de inicio de vigencia del esquema.
fecha_fin	date	Fecha de fin de vigencia del esquema.
empresa_inquilina_id	bigint(19)	Empresa inquilina que define el esquema.

Fuente: Elaboración propia

Tabla 3-10. Diccionario de datos: Evidencia de incidente

Nombre:	evidencia_incidente	
Descripción:	Guarda las rutas o referencias a los archivos de evidencia (fotos, documentos) adjuntos a un incidente.	
Clave Primaria:	id_evidencia_incidente	
Clave Foránea:	incidente_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_evidencia_incidente	bigint(19)	Identificador único de la evidencia.
url_evidencia	varchar(2048)	URL o ruta donde está almacenado el archivo de evidencia.
tipo_archivo	varchar(10)	Tipo de archivo (ej: jpg, png, pdf).
fecha_carga	timestamp	Fecha y hora en que se cargó la evidencia.
incidente_id	bigint(19)	Incidente al que pertenece la evidencia.

Fuente: Elaboración propia

Tabla 3-11. Diccionario de datos: Incidente

Nombre:	incidente	
Descripción:	Registra los incidentes reportados por los usuarios relacionados con el uso de los estacionamientos.	
Clave Primaria:	id_incidente	
Clave Foránea:	reportante_usuario_id (referencia a usuario) gestor_usuario_id (referencia a usuario) edificio_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_incidente	bigint(19)	Identificador único del incidente.
tipo	varchar(50)	Tipo de incidente (ocupación indebida, daño, robo).
descripcion	varchar	Descripción detallada del incidente.
fecha_hora_reporte	timestamp	Fecha y hora en que se reportó el incidente.
estado	varchar(20)	Estado del incidente (reportado, en progreso, resuelto).
reportante_usuario_id	bigint(19)	Usuario que reporta el incidente.
gestor_usuario_id	bigint(19)	Operador que gestiona el incidente.
edificio_id	bigint(19)	Identificador del edificio donde se reportó el incidente.

Fuente: Elaboración propia

Tabla 3-12. Diccionario de datos: Método de acceso

Nombre:	metodo_acceso	
Descripción:	Almacena los diferentes métodos o credenciales de acceso físico (ej: tarjeta RFID) asociados a un vehículo.	
Clave Primaria:	id_metodo_acceso	
Clave Foránea:	vehiculo_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_metodo_acceso	bigint(19)	Identificador único para el método de acceso.
tipo	varchar(50)	Tipo de identificador (ej: RFID, TAG).
uid	varchar(100)	Código del identificador.
activo	integer(1)	Indica si el método de acceso está habilitado.
vehiculo_id	bigint(19)	Identificador del vehículo al que está asociado el método de acceso.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-13. Diccionario de datos: Plaza de estacionamiento

Nombre:	plaza_estacionamiento	
Descripción:	Define cada una de las plazas o espacios de estacionamiento disponibles en un edificio.	
Clave Primaria:	id_plaza_estacionamiento	
Clave Foránea:	edificio_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_plaza_estacionamiento	bigint(19)	Identificador único para cada plaza de estacionamiento.
codigo	varchar(20)	Código o número de la plaza (ej: A-101).
tipo_uso	varchar(50)	Uso destinado para la plaza (ej: inquilino, visita, administración).
tipo_vehiculo	varchar(50)	Tipo de vehículo admitido (ej: convencional, eléctrico).
ubicación	varchar(50)	Descripción de la ubicación (ej: Piso -1, Sector A).
estado_plaza	varchar(50)	Estado actual de la plaza (ej: disponible, asignada, mantenimiento).
edificio_id	bigint(19)	Identificador del edificio al que pertenece la plaza.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-14. Diccionario de datos: Registro de acceso

Nombre:	registro acceso	
Descripción:	Guarda un historial de todos los eventos de entrada y salida de vehículos en los estacionamientos.	
Clave Primaria:	id_registro_acceso	
Clave Foránea:	autorizador_usuario_id (referencia a usuario) reserva_estacionamiento_id solicitud_estacionamiento_visita_id metodo_acceso_id controlador_acceso_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_registro_acceso	bigint(19)	Identificador único del registro de acceso.
fecha_hora_acceso	timestamp	Fecha y hora exactas del evento de acceso.
tipo_acceso	varchar(20)	Tipo de evento ('entrada' o 'salida').
autorizador_usuario_id	bigint(19)	Usuario que autoriza el acceso manual (si aplica).
reserva_estacionamiento_id	bigint(19)	Reserva asociada al acceso (si aplica).
solicitud_estacionamiento_visita_id	bigint(19)	Solicitud de visita asociada al acceso (si aplica).
controlador_acceso_id	bigint(19)	Controlador físico que registró el acceso.

Fuente: Elaboración propia

Tabla 3-15. Diccionario de datos: Registro de auditoría

Nombre:	registro_auditoria	
Descripción:	Almacena los registros de auditoría generados por el sistema ante operaciones de creación, modificación o eliminación de entidades persistentes.	
Clave Primaria:	id_registro_auditoria	
Clave Foránea:		
COLUMNA	TIPO	DESCRIPCIÓN
id_registro_auditoria	bigint(19)	Identificador único del registro de auditoría.
fecha	timestamp	Fecha y hora en que se registra el evento de auditoría.
entidad	varchar(50)	Nombre de la entidad del dominio afectada por la operación.
id_entidad	bigint(19)	Identificador de la entidad del dominio sobre la cual se realizó la operación.
accion	varchar(20)	Tipo de operación CUD ejecutada sobre la entidad.
id_usuario	bigint(19)	Identificador del usuario que ejecutó la operación, si aplica.
email_usuario	varchar(100)	Email del usuario responsable de la operación, si aplica.
detalle	text	Información detallada del cambio auditado. Puede incluir valores anteriores, posteriores u otros datos relevantes del evento.

Fuente: Elaboración propia

Tabla 3-16. Diccionario de datos: Regla de reserva

Nombre:	regla_reserva	
Descripción:	Permite a las empresas inquilinas configurar reglas específicas para las reservas de sus empleados.	
Clave Primaria:	id_regla_reserva	
Clave Foránea:	empresa_inquilina_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_regla_reserva	bigint(19)	Identificador único de la regla.
nombre	varchar(50)	Nombre de la regla (ej: ANTELACION_MAXIMA).
valor	integer(10)	Valor numérico de la regla.
unidad_tiempo	varchar(50)	Unidad de tiempo para el valor (ej: horas, días).
empresa_inquilina_id	bigint(19)	Empresa inquilina que define la regla.

Fuente: Elaboración propia

Tabla 3-17. Diccionario de datos: Reserva de estacionamiento

Nombre:	reserva estacionamiento	
Descripción:	Registra las reservas de plazas de estacionamiento realizadas por los usuarios.	
Clave Primaria:	id_reserva_estacionamiento	
Clave Foránea:	usuario_id plaza_estacionamiento_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_reserva_estacionamiento	bigint(19)	Identificador único de la reserva.
fecha_hora_inicio	timestamp	Fecha y hora en que comienza la reserva.
fecha_hora_fin	timestamp	Fecha y hora en que termina la reserva.
usuario_id	bigint(19)	Identificador del usuario que realizó la reserva.
plaza_estacionamiento_id	bigint(19)	Identificador de la plaza de estacionamiento reservada.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-18. Diccionario de datos: Rol

Nombre:	rol	
Descripción:	Define los diferentes roles o perfiles de usuario dentro del sistema, los cuales determinan sus permisos y funcionalidades accesibles.	
Clave Primaria:	id_rol	
Clave Foránea:		
COLUMNA	TIPO	DESCRIPCIÓN
id_rol	bigint(19)	Identificador único para cada rol.
nombre	varchar(50)	Nombre descriptivo del rol (ej: Administrador, Empleado, Operador).

Fuente: Elaboración propia

Tabla 3-19. Diccionario de datos: Solicitud de estacionamiento de visita

Nombre:	solicitud estacionamiento visita	
Descripción:	Almacena las solicitudes de estacionamiento para visitantes realizadas por los empleados.	
Clave Primaria:	id_solicitud_estacionamiento_visita	
Clave Foránea:	gestor_usuario_id (referencia a usuario) plaza_estacionamiento_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_solicitud_estacionamiento_visita	bigint(19)	Identificador único de la solicitud.
tipo_id_visita	varchar(50)	Tipo de documento del visitante (RUN, Pasaporte).
identificador_visita	varchar(20)	Número del documento del visitante.
nombre_visita	varchar(50)	Nombre(s) del visitante.
apellidos_visita	varchar(50)	Apellido(s) del visitante.
email_visita	varchar(100)	Correo electrónico del visitante.
patente_vehiculo	varchar(10)	Patente del vehículo del visitante.
fecha_hora_desde	timestamp	Inicio del periodo solicitado para la visita.
fecha_hora_hasta	timestamp	Fin del periodo solicitado para la visita.
motivo	varchar(20)	Motivo de la visita.
comentarios	varchar(255)	Comentarios adicionales sobre la solicitud.
estado	varchar(20)	Estado de la solicitud (en revisión, aprobada, rechazada).
gestor_usuario_id	bigint(19)	ID del usuario (operador) que gestiona la solicitud.
motivo_rechazo	varchar(255)	Motivo por el cual fue rechazada la solicitud.
plaza_estacionamiento_id	bigint(19)	ID de la plaza asignada si la solicitud es aprobada.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-20. Diccionario de datos: Token de restablecimiento de clave

Nombre:	token restablecer clave	
Descripción:	Almacena los tokens seguros y temporales generados para el proceso de restablecimiento de contraseña de los usuarios.	
Clave Primaria:	id_token	
Clave Foránea:		
COLUMNA	TIPO	DESCRIPCIÓN
id_token	bigint(19)	Identificador único del token.
token	varchar(255)	El valor del token único y seguro.
expiracion	timestamp	Fecha y hora en que el token deja de ser válido.
estado	varchar(20)	Estado del token.
email_usuario	varchar(255)	Email del usuario que solicitó el restablecimiento de contraseña.

Fuente: Elaboración propia

Tabla 3-21. Diccionario de datos: Usuario

Nombre:	Usuario	
Descripción:	Almacena la información de todos los usuarios que pueden interactuar con el sistema, como administradores, operadores o empleados.	
Clave Primaria:	id_usuario	
Clave Foránea:	vehiculo_id	
COLUMNA	TIPO	DESCRIPCIÓN
id_usuario	bigint(19)	Identificador único para cada usuario.
email	varchar(100)	Correo electrónico del usuario, usado para inicio de sesión y notificaciones.
clave	varchar(255)	Contraseña cifrada del usuario para el acceso al sistema.
run	varchar(12)	Rol Único Nacional del usuario.
nombre	varchar(100)	Nombre(s) del usuario.
apellidos	varchar(100)	Apellido(s) del usuario.
estado	varchar(20)	Estado actual de la cuenta del usuario (ej.: activa, nueva).
vehiculo_id	bigint(19)	Identificador del vehículo asociado al usuario (si aplica).
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

Tabla 3-22. Diccionario de datos: Vehículo

Nombre:	vehiculo	
Descripción:	Registra los vehículos asociados a los usuarios del sistema.	
Clave Primaria:	id_vehiculo	
Clave Foránea:		
COLUMNA	TIPO	DESCRIPCIÓN
id_vehiculo	bigint(19)	Identificador único para cada vehículo.
patente	varchar(10)	Patente única del vehículo.
marca	varchar(50)	Marca del vehículo (ej: Toyota).
modelo	varchar(50)	Modelo del vehículo (ej: Yaris).
color	varchar(30)	Color del vehículo.
fecha_eliminacion	timestamp	Fecha y hora de eliminación lógica; NULL indica registro activo.

Fuente: Elaboración propia

3.3 Diagramas de secuencia extendido

Esta sección complementa la vista estática del modelo de clases mediante la representación del comportamiento dinámico del sistema. Los diagramas de secuencia extendidos que se presentan a continuación modelan el flujo de interacciones entre los distintos componentes definidos por la arquitectura hexagonal.

Su objetivo es describir, de forma progresiva, cómo los componentes colaboran e intercambian mensajes para ejecutar un caso de uso, poniendo énfasis en las responsabilidades y en la orquestación del comportamiento del sistema. Con este fin, los diagramas se presentan a un nivel de abstracción arquitectónico, omitiendo deliberadamente detalles de implementación e infraestructura -como frameworks, mecanismos de persistencia o mapeos de datos- que no aportan a la comprensión del diseño lógico ni del flujo de negocio.

En particular, los mecanismos transversales asociados a requerimientos no funcionales, como la auditoría (RNF-01), se representan de forma abstracta y su implementación se documenta en el Anexo 1 “ADR – Mecanismo de auditoría (RNF-01)”.

3.3.1 Diagrama de secuencia extendido CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina

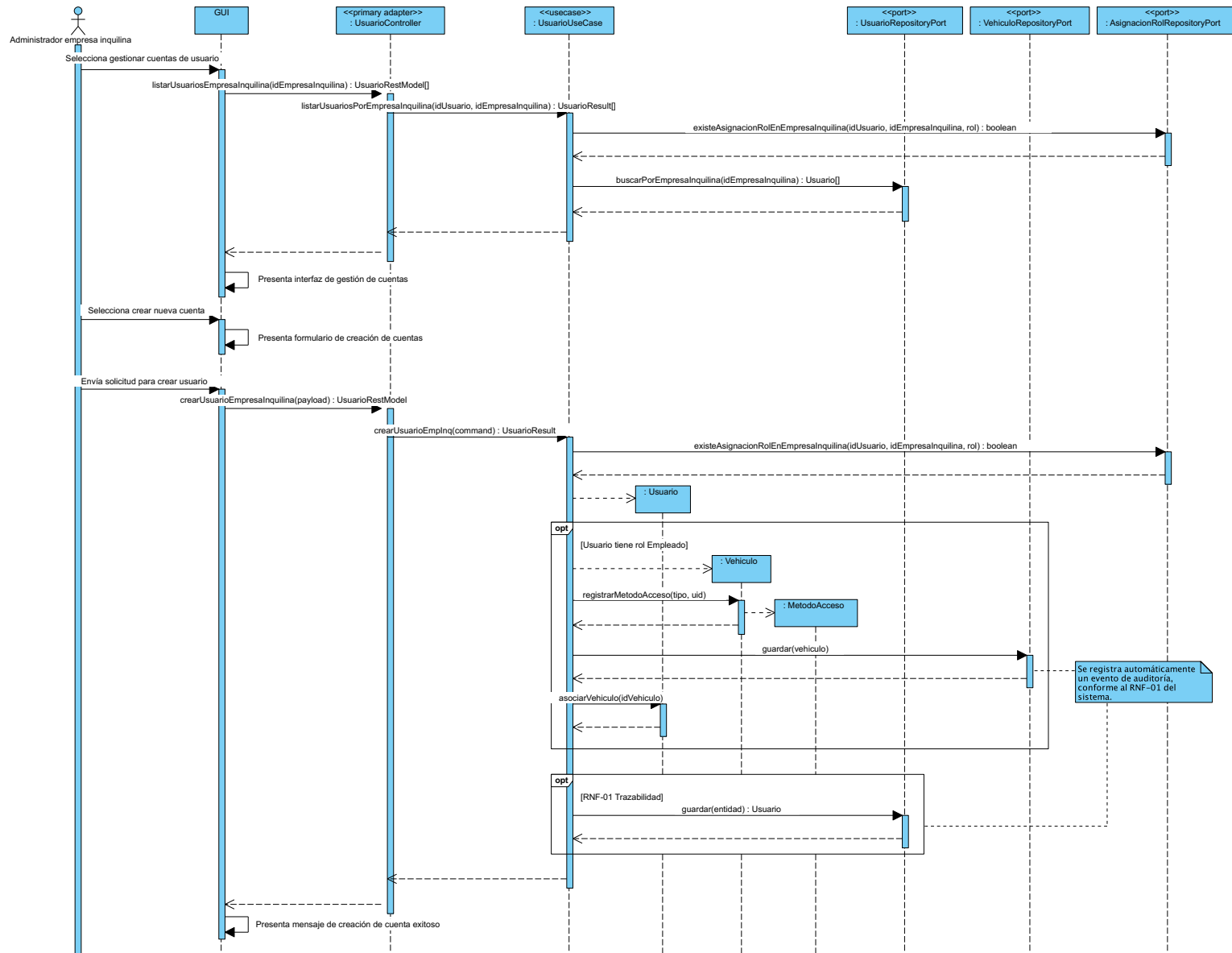


Figura 3-6. Diagrama de secuencia extendido: CU-07 Gestionar empleados y sus métodos de acceso de la empresa inquilina

Fuente: Elaboración propia

3.3.2 Diagrama de secuencia extendido CU-10 Controlar y registrar entradas/salidas de vehículos

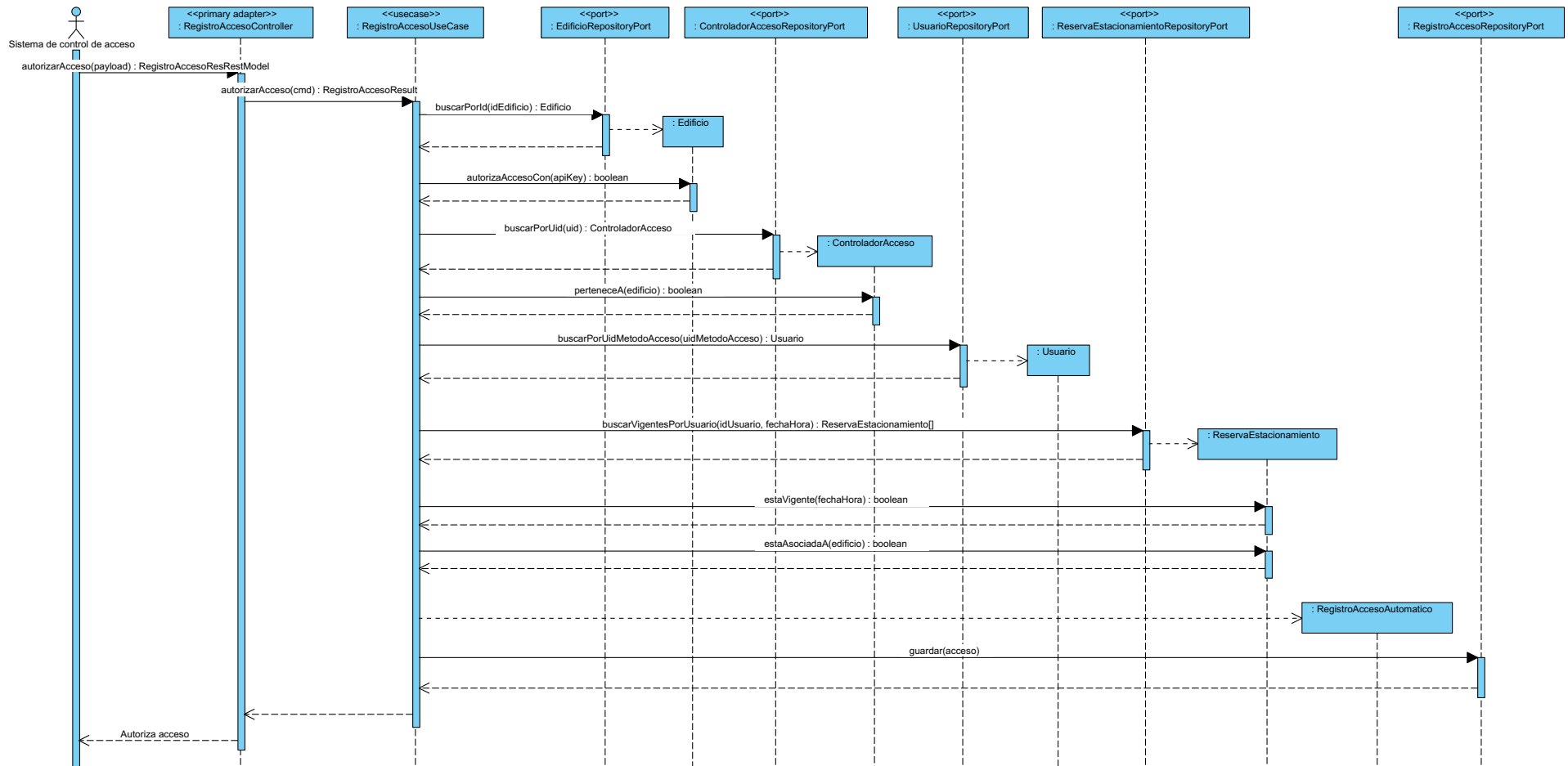


Figura 3-7. Diagrama de secuencia extendido: CU-10 Controlar y registrar entradas/salidas de vehículos

Fuente: Elaboración propia

3.3.3 Diagrama de secuencia extendido CU-14 Gestionar reservas de estacionamiento de empresa inquilina

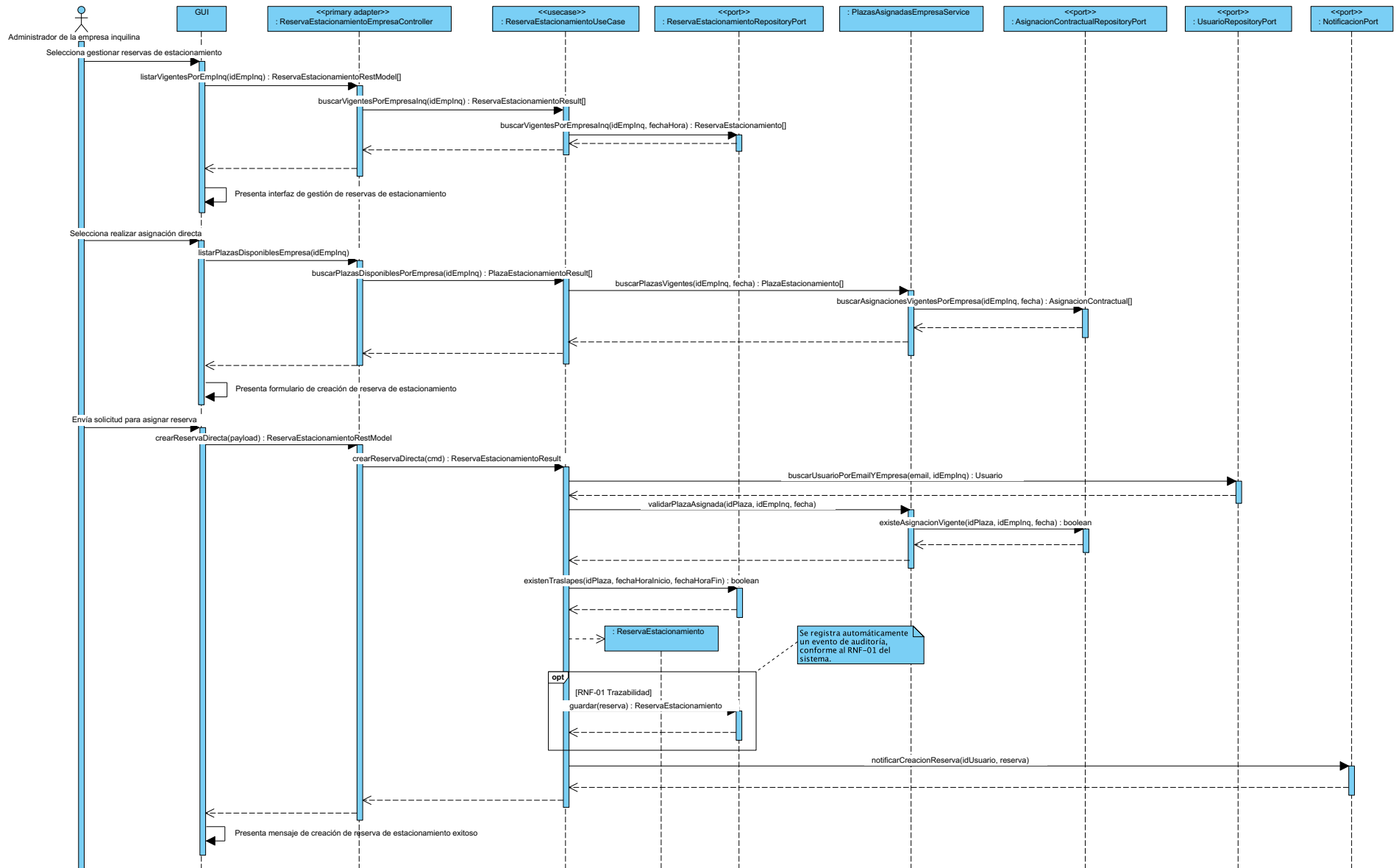


Figura 3-8. Diagrama de secuencia extendido: CU-14 Gestionar reservas de estacionamiento de empresa inquilina

Fuente: Elaboración propia

Diagrama de secuencia extendido CU-15 Gestionar mis reservas de estacionamiento

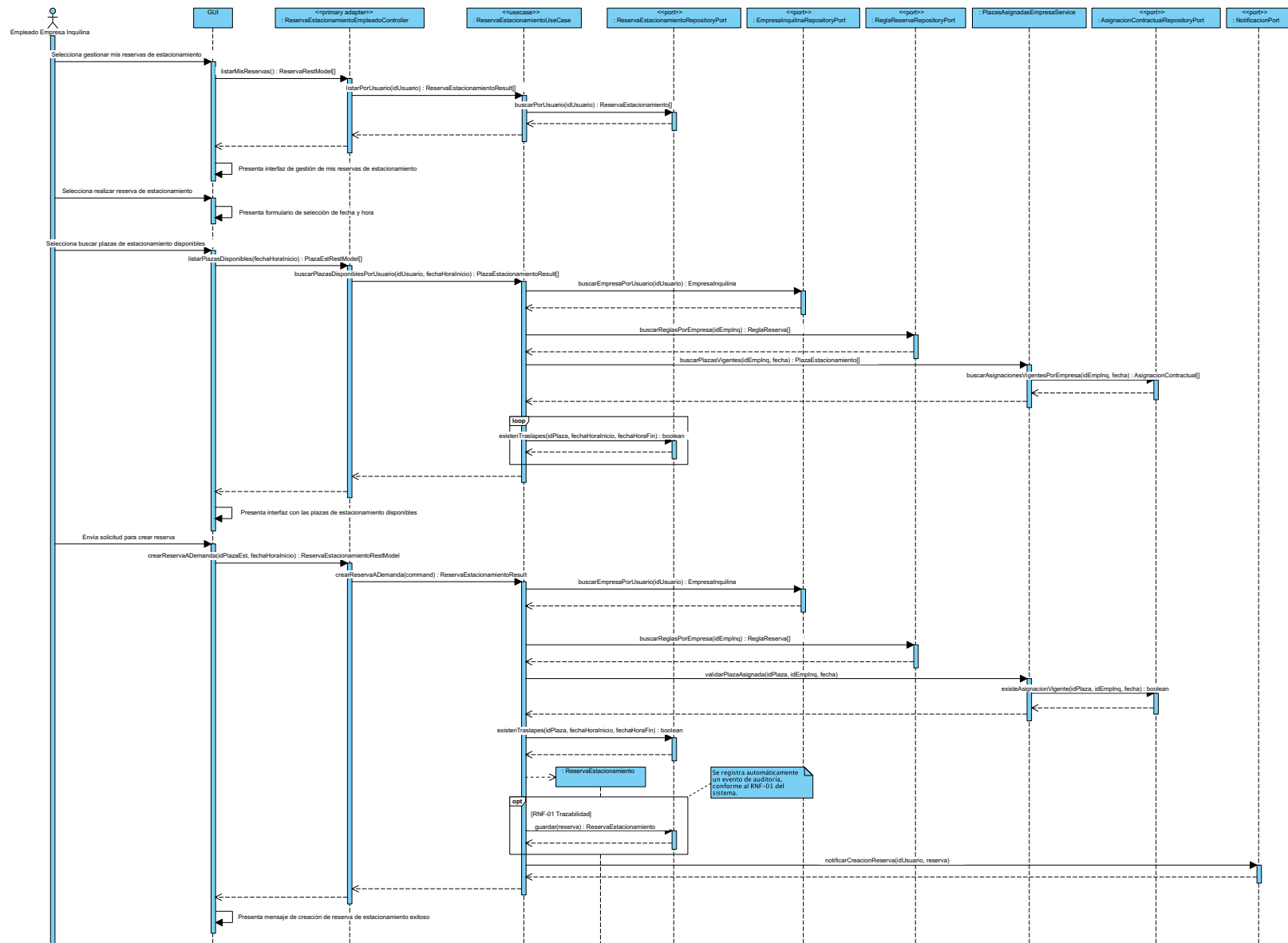


Figura 3-9. Diagrama de secuencia extendido: CU-15 Gestionar mis reservas de estacionamiento

Fuente: Elaboración propia

3.3.4 Diagrama de secuencia extendido CU-16 Gestionar mis solicitudes de estacionamiento de visitas

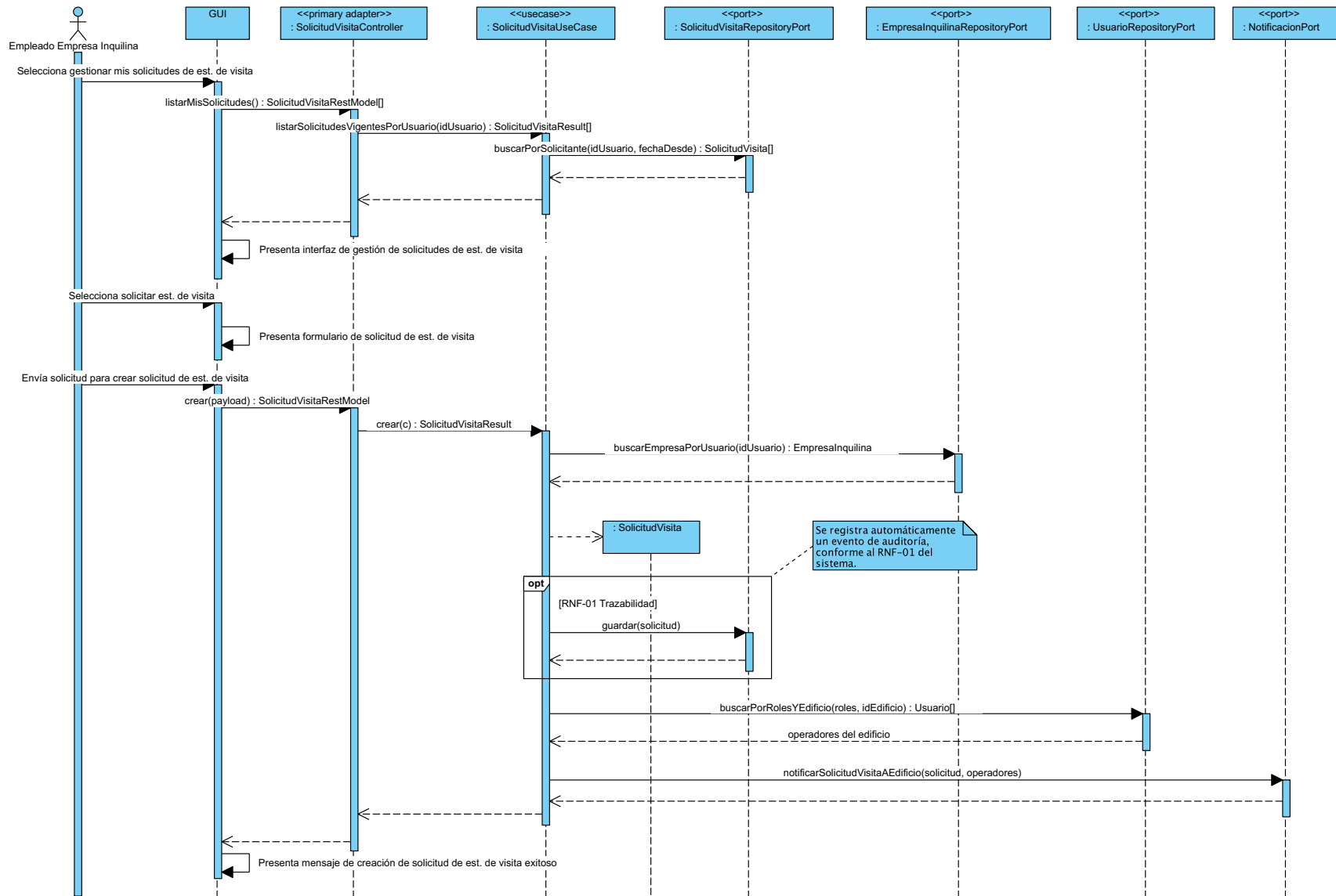


Figura 3-10. Diagrama de secuencia extendido: CU-16 Gestionar mis solicitudes de estacionamiento de visitas

Fuente: Elaboración propia

3.3.5 Diagrama de secuencia extendido CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita

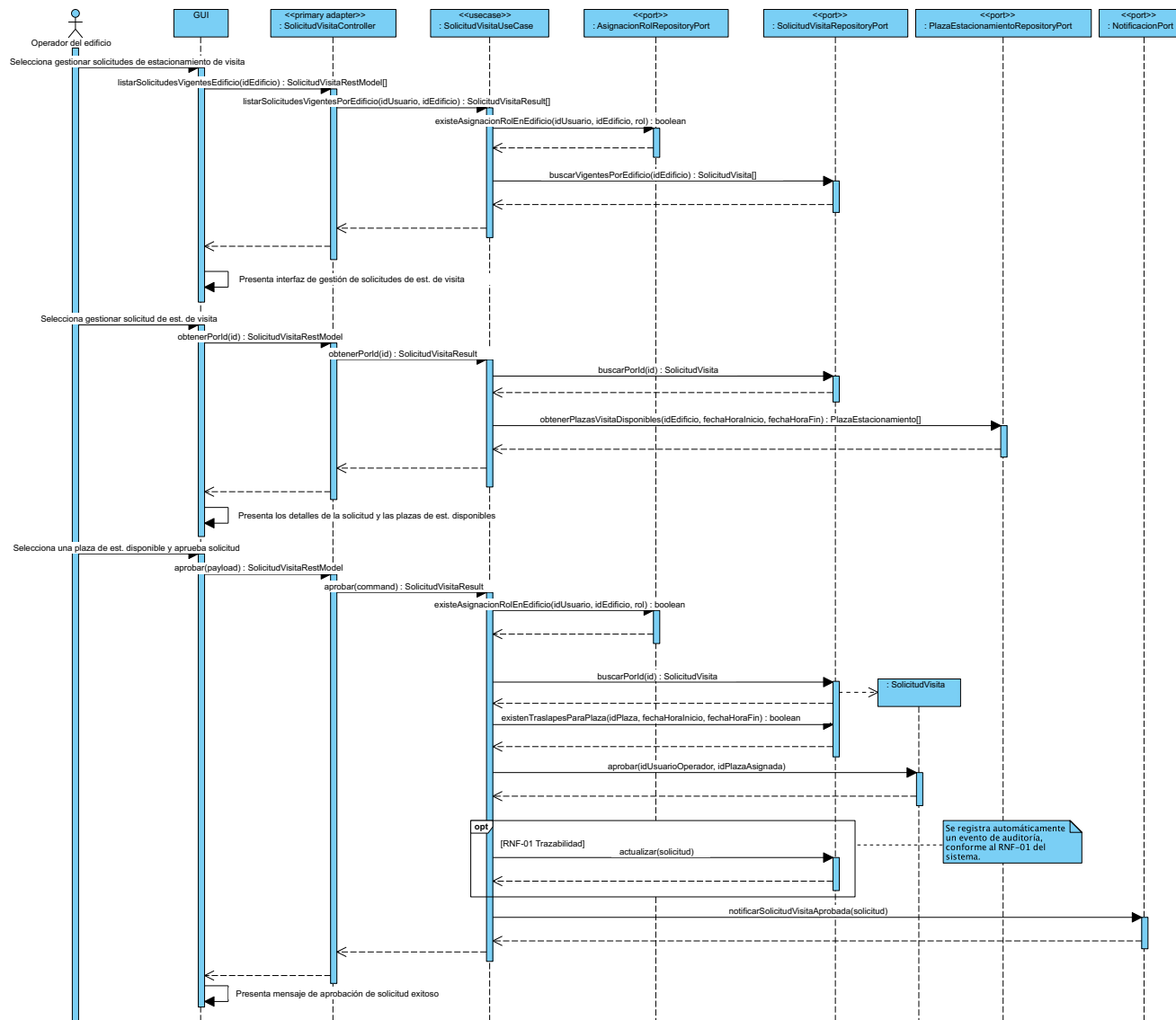


Figura 3-11. Diagrama de secuencia extendido CU-17 Aprobar/Rechazar solicitudes de estacionamiento de visita

Fuente: Elaboración propia

3.3.6 Diagrama de secuencia extendido CU-19 Gestionar mis reportes de incidentes

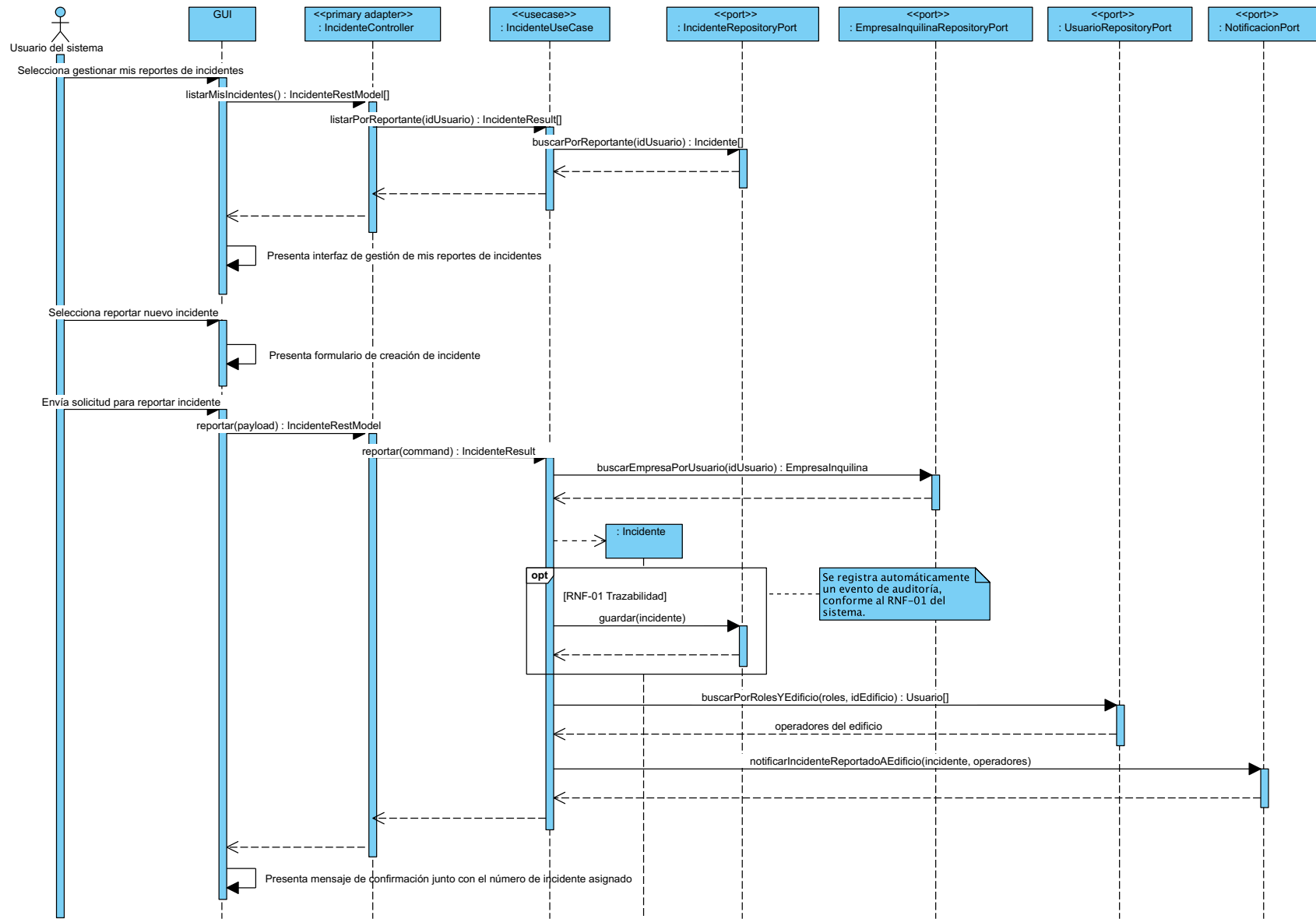


Figura 3-12. Diagrama de secuencia extendido: CU-19 Gestionar mis reportes de incidentes

Fuente: Elaboración propia

3.3.7 Diagrama de secuencia extendido CU-20 Gestionar incidentes operativos reportados

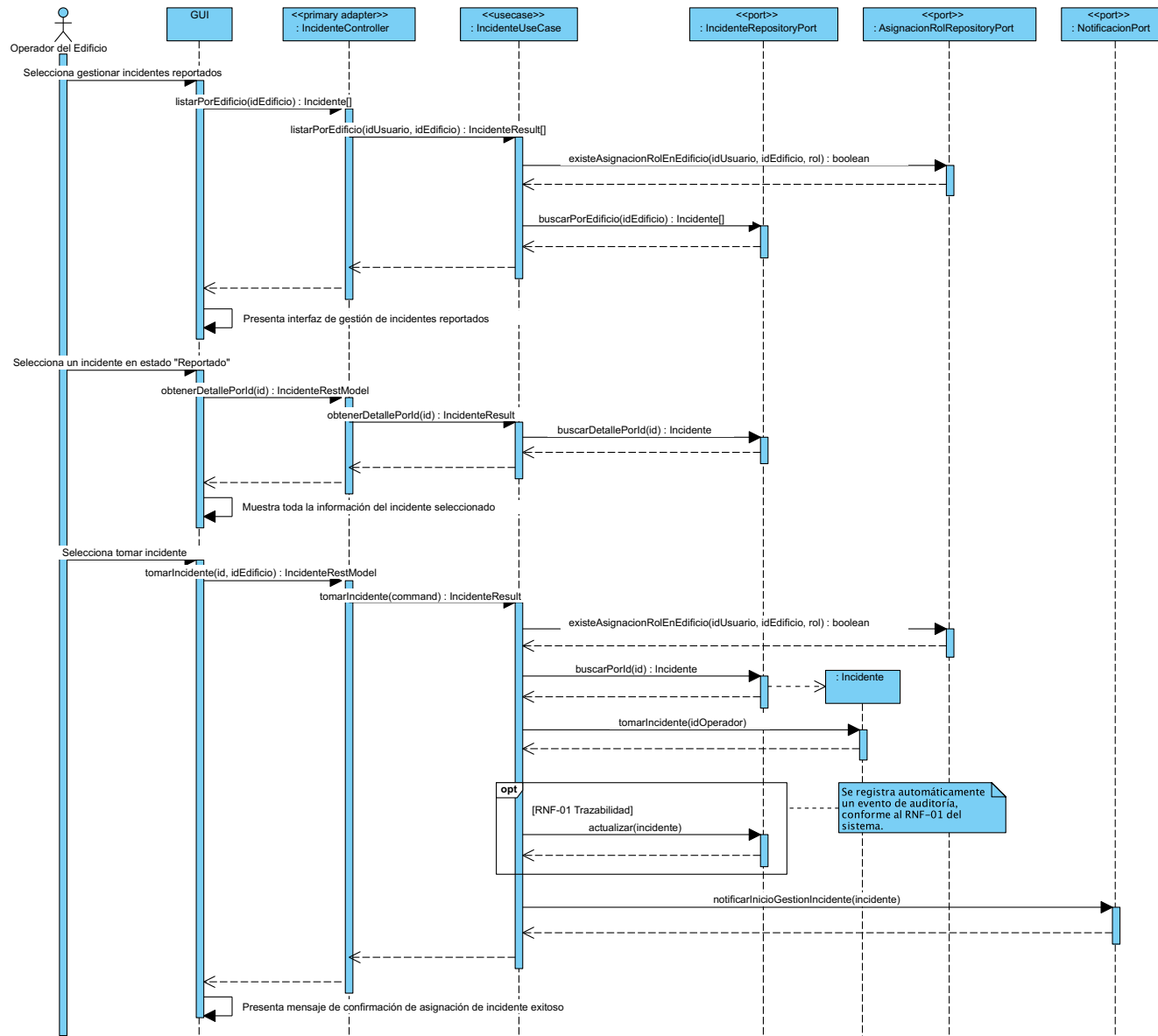


Figura 3-13. Diagrama de secuencia extendido: CU-20 Gestionar incidentes operativos reportados

Fuente: Elaboración propia

3.4 Diseño de Interfaz

El diseño de la interfaz es el componente que materializa la funcionalidad del sistema para el usuario final, siendo un factor clave para su usabilidad y adopción. Con el objetivo de garantizar una experiencia de usuario intuitiva, eficiente y satisfactoria, el diseño de todas las pantallas se guiará por las heurísticas de usabilidad de Jakob Nielsen. Estos principios, reconocidos en la industria del software, aseguran que la interfaz sea no solo funcionalmente correcta, sino también fácil de aprender y utilizar.

A continuación, se expondrá la arquitectura de navegación, la filosofía del diseño, los patrones de usabilidad definidos y ejemplos visuales de su implementación, demostrando la aplicación práctica de estas heurísticas para crear un flujo claro y coherente para el usuario final.

3.4.1 Diagrama de Navegación

Como un paso fundamental para materializar una interfaz de usuario coherente e intuitiva, a continuación se presenta el diagrama de navegación. Este diagrama, también conocido como mapa del sitio, actúa como el plano arquitectónico de la aplicación web, ilustrando su estructura de información y la jerarquía de contenidos.

La navegación se divide en dos grandes secciones:

- Un Portal Público, que permite el acceso a funcionalidad básicas como el inicio de sesión y la recuperación de contraseña.
- Un Portal Privado, accesible solo para usuarios autenticados, que agrupa todas las funcionalidades de gestión del sistema, cómo la configuración de edificios, la administración de reservas y la generación de reportes.

Este diseño tiene como propósito definir la organización de las pantallas y las rutas que los usuarios seguirán para completar sus tareas, sentando las bases para un flujo de interacción claro y predecible.

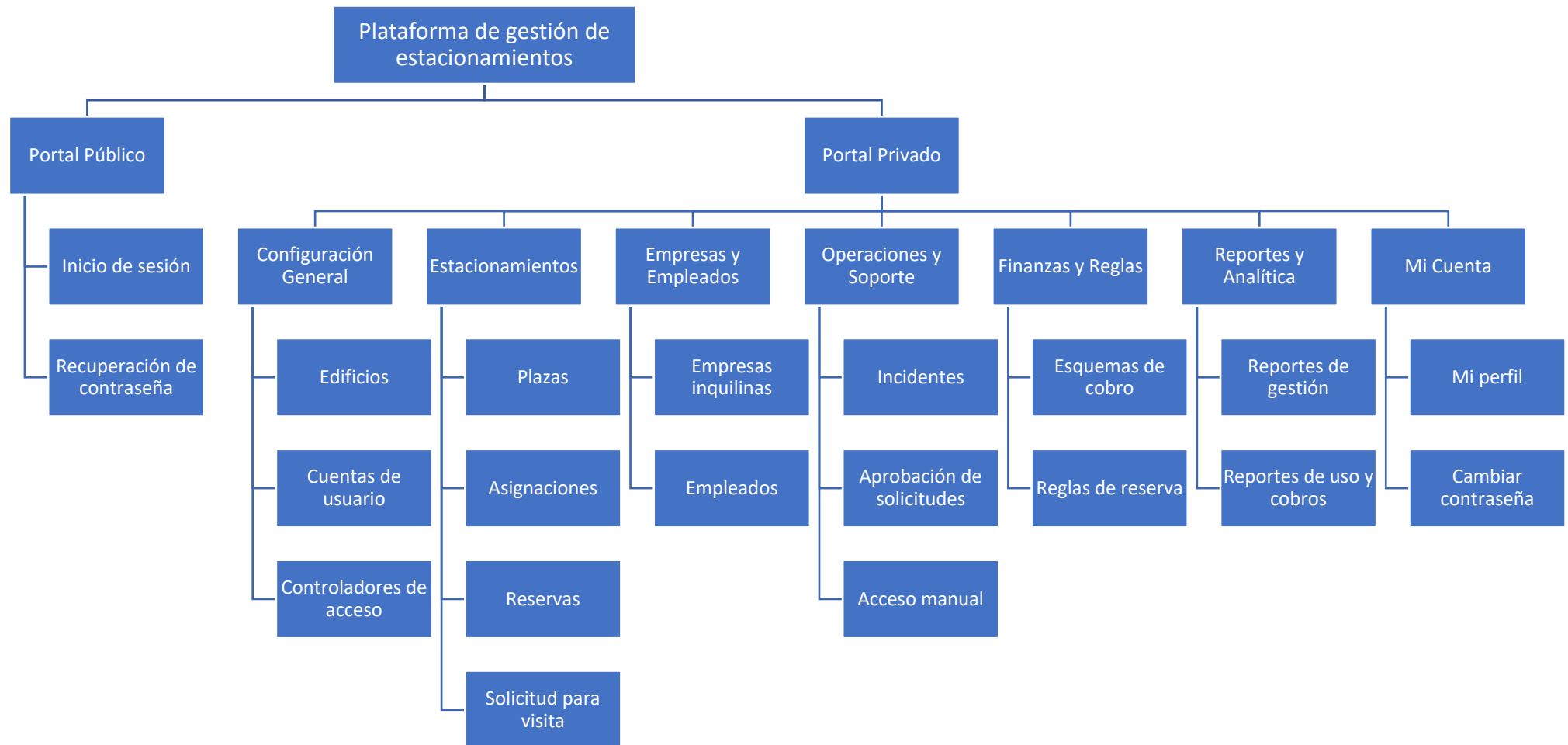


Figura 3-14. Mapa del sitio

Fuente: Elaboración propia

3.4.2 Filosofía del diseño

El diseño de la interfaz de usuario para el sistema se fundamenta en un enfoque centrado en el usuario, cuyo objetivo principal es minimizar la carga cognitiva y maximizar la eficiencia. Para lograrlo, se ha definido un sistema de diseño coherente basado en un conjunto de patrones de usabilidad aplicados de manera consistente a través de tres tipos principales de interacciones: la realización de tareas complejas, la visualización de información y el ingreso de información mediante formularios. Cada patrón seleccionado responde directamente a principios de usabilidad establecidos, como las heurísticas de Jakob Nielsen, para garantizar una experiencia intuitiva, predecible y libre de errores.

3.4.2.1 Patrones para la realización de tareas complejas

Para guiar al usuario a través de procesos que requieren múltiples acciones, como la creación de una reserva o el reporte de un incidente, se han definido los siguientes patrones:

Asistente (Wizard)

Este patrón descompone una tarea larga en una secuencia de pasos más pequeños y manejables. El sistema guía al usuario paso a paso hasta completar el objetivo, aplicando el principio de **Visibilidad del estado del sistema**.

The wireframe shows a form titled "Nueva Reserva (Paso 2 de 2)". In the top right corner, there is a yellow callout box labeled "Visibilidad del estado del sistema". Below the title, the instruction "Selecciona una de las plazas disponibles" is followed by three radio button options: "Plaza A-101 (Piso -1)", "Plaza A-102 (Piso -1)", and "Plaza B-054 (Piso -2)". The second option is selected. A yellow callout box labeled "Prevención de errores" points to the selected option. At the bottom, there are two buttons: "< Volver" on the left and "Confirmar Reserva" on the right.

Figura 3-15. Wireframe: Patrón Asistente

Fuente: Elaboración propia

Retroalimentación de Acción (Feedback)

Después de que el usuario completa una acción, el sistema muestra un mensaje breve y no intrusivo (conocido como “toast” o “snackbar”) que confirma el resultado, reforzando la **Visibilidad del estado del sistema**.

The wireframe illustrates the 'Editar Perfil de Empleado' (Edit Employee Profile) form. It includes several input fields and a feedback mechanism. At the top, a green toast message reads '✓ Cambios guardados correctamente' (Changes saved correctly). Below the form fields, a yellow toast message reads 'Visibilidad del estado del sistema' (System status visibility). The form fields include: 'Nombre(s)' (Name) with the value 'Eduardo', 'Apellidos' (Surnames), 'RUT' (Tax ID) with the value '16.994.991-3' and a format hint 'Formato: 12.345.678-9', 'Roles' (Roles) with a dropdown menu showing 'Empleado, Analista', 'Correo Electrónico' (Email) with the value 'ediaz.vasquez@empresa.com', and a 'Datos del Vehículo' (Vehicle Data) section with fields for 'Patente' (License Plate) 'ABCD-12', 'Marca' (Brand) 'Tesla', and 'Modelo' (Model) 'Model 3'. Below the vehicle data, there is a 'Métodos de Acceso' (Access Methods) section with a dashed border, containing 'TAG Windshield: A1-B2-C3-D4' and 'Patente Vehículo: ABCD-12'. At the bottom, there are two buttons: 'Cancelar' (Cancel) and 'Guardar Cambios' (Save Changes). A yellow toast message 'Prevención de errores' (Error prevention) is also visible near the 'Roles' dropdown.

Figura 3-16. Wireframe: Patrón Retroalimentación de Acción

Fuente: Elaboración propia

Confirmación Explícita

Antes de ejecutar una acción importante o destructiva, el sistema presenta un cuadro de diálogo (modal) pidiendo al usuario que confirme su decisión. Este patrón es una aplicación directa de los principios de **Prevención de errores** y **Control y libertad del usuario**.



Figura 3-17. Wireframe: Patrón Confirmación Explícita

Fuente: Elaboración propia

3.4.2.2 Patrones para la visualización de información

La presentación de conjuntos de datos, como listados de incidentes o plazas de estacionamiento, se estandariza mediante los siguientes patrones:

Tabla de Datos Interactiva

Los datos se presentan en una grilla que permite al usuario ordenar, filtrar y paginar, promoviendo la **Flexibilidad y eficiencia de uso**. Asimismo, se aplica **Reconocimiento en lugar de recuerdo** al situar los iconos de acción junto a cada registro, lo que elimina la necesidad de memorizar dichas funciones en menús ocultos, facilitando una interacción intuitiva.

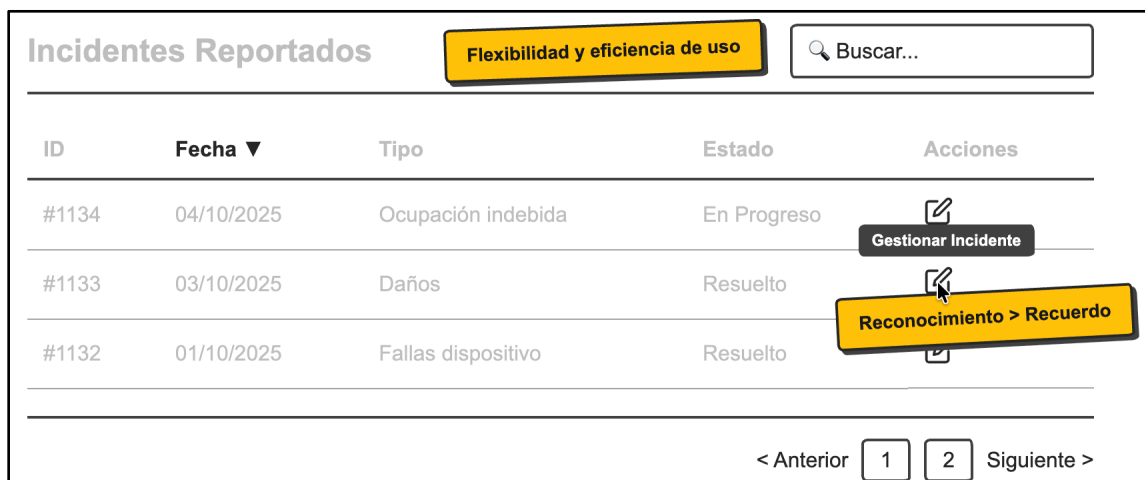


Figura 3-18. Wireframe: Patrón Tabla de Datos Interactiva

Fuente: Elaboración propia

Acción Principal Destacada

En una pantalla con múltiples opciones, la acción más común se resalta visualmente para asegurar la **Consistencia y estándares**.

The wireframe shows a header section with the title 'Plazas de Estacionamiento' and two buttons: 'Consistencia y estándares' (highlighted in yellow) and '+ Añadir Plaza'. Below the header is a table with the following data:

Código	Ubicación	Tipo	Estado	Acciones
A-101	Piso -1	Empresa	Asignada	
A-102	Piso -1	Empresa	Disponible	

Figura 3-19. Wireframe: Patrón Acción Principal Destacada

Fuente: Elaboración propia

Navegación por “Migas de Pan” (Breadcrumbs)

Muestra la ruta jerárquica de la página actual, materializando la **Visibilidad del estado del sistema** y otorgando al usuario mayor **Control y libertad del usuario** sobre la navegación.

The wireframe shows a breadcrumb trail at the top: 'Portal > Estacionamientos > Editar Plaza A-101'. Below the breadcrumb, the title 'Editar Plaza A-101' is displayed. The form contains the following fields and controls:

- Código de Plaza:** Input field with 'A-101'.
- Ubicación:** Input field with 'Piso -1'.
- Tipo de Uso:** Dropdown menu with 'Empresa' selected.
- Estado:** Dropdown menu with 'Asignada' selected.
- Buttons:** 'Cancelar' and 'Guardar Cambios'.

Highlighted elements include the breadcrumb trail, the 'Control y libertad del usuario' button, the 'Visibilidad del estado del sistema' button, and the 'Prevención de errores' button.

Figura 3-20. Wireframe: Patrón Navegación por Migas de Pan

Fuente: Elaboración propia

3.4.2.3 Patrones para Formularios y Entrada de Datos

La entrada de datos ha sido diseñada para ser sencilla y reducir al máximo la posibilidad de errores, utilizando los siguientes patrones:

Etiquetas Claras y Alineadas

Cada campo tiene una etiqueta descriptiva y ubicada en la parte superior para una fácil lectura, lo que se alinea con la **Relación entre el sistema y el mundo real** y **Consistencia y estándares**.

The wireframe shows a form titled "Añadir Empresa Inquilina". It contains the following elements:

- Razón Social:** A text input field with a yellow callout box labeled "Relación con el mundo real" pointing to it.
- Nombre de Fantasía:** A text input field.
- RUT:** A text input field with a yellow callout box labeled "Consistencia y estándares" pointing to it. Below the field is the text "Formato: 12.345.678-9".
- Teléfono de Contacto:** A text input field with the text "Formato: +56 9 1234 5678" below it.
- Ubicación (Piso / Oficina):** A text input field.
- Buttons:** "Cancelar" and "Guardar Empresa" at the bottom.

Figura 3-21. Wireframe: Patrón Etiquetas Claras y Alineadas

Fuente: Elaboración propia

Valores por Defecto Inteligentes (Smart Defaults)

Se pre-lleñan campos con valores más probables para acelerar el proceso, aplicando los principios de **Prevención de errores** y **Flexibilidad y eficiencia de uso**.

Nueva Reserva (Paso 1 de 2)

Fecha de reserva **Hora de inicio**

06/10/2025 08:00 ▼

Prevención de errores Flexibilidad y eficiencia de uso

Cancelar Buscar Disponibilidad

Figura 3-22. Wireframe: Patrón Valores por Defecto Inteligentes

Fuente: Elaboración propia

Validación en Línea y Mensajes de Error Constructivos

El sistema revisa la información a medida que el usuario la ingresa y muestra mensajes de error claros, lo cual es una aplicación directa de los principios de **Prevención de errores** y **Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de errores**.

Editar Perfil de Empleado

Nombre(s)

Apellidos

RUT

Formato: 12.345.678-9

Roles

Empleado, Analista ▼

Correo Electrónico

Formato de correo no válido. Debe incluir @.

Datos del Vehículo

Patente

Marca

Modelo

Métodos de Acceso

TAG Windshield:

Patente Vehículo:

Cancelar

Guardar Cambios

Figura 3-23. Wireframe: Patrón Validación en Línea y Mensajes de Error Constructivos

Fuente: Elaboración propia

CONCLUSIONES

El desarrollo de este trabajo de título, centrado en la creación de una “Plataforma para la gestión y control de estacionamiento”, ha permitido constatar que el proceso de creación de software, si bien produce un bien intangible, está expuesto a los mismos desafíos de rigurosidad, planificación y gestión de la complejidad que cualquier proyecto de ingeniería tradicional. La principal reflexión extraída de este proceso es que la elevada carga cognitiva inherente al desarrollo de software puede ser gestionada eficazmente mediante la aplicación sistemática de las prácticas de la ingeniería de software, lo que aumenta significativamente la probabilidad de construir una solución de alta calidad que responda a las necesidades reales de los clientes.

El capítulo 1, “Aspectos relevantes del proyecto y su gestión”, sentó las bases de este enfoque ingenieril. El punto de partida no fue la tecnología, sino un análisis metódico del negocio de la empresa Arikok y su problemática fundamental: una gestión de estacionamientos manual, ineficiente y propensa a errores que afectaba la experiencia de los inquilinos y la eficiencia operativa. La identificación de problemas específicos, como las demoras en los accesos, la falta de un control automatizado y la ineficiente comunicación con las empresas inquilinas, fue un paso crucial para delimitar el alcance del proyecto. La evaluación de alternativas de solución, donde se ponderan criterios como funcionalidad, costo y seguridad, y la posterior selección del desarrollo a medida, demostró que la toma de decisiones debe basarse en un análisis estructurado y no es suposiciones. Finalmente, la planificación, incluyendo la administración de riesgos y la estimación de costos reforzó la idea de que un proyecto de software exitoso requiere de la misma disciplina de planificación y control que una obra de ingeniería civil.

Posteriormente, el capítulo 2, “Aspectos relevantes del análisis”, representó la fase de mayor carga cognitiva, donde el problema abstracto definido en el capítulo anterior debía ser traducido a una especificación formal y sin ambigüedades. Es aquí donde las prácticas de la ingeniería de software demostraron su valor como herramientas para gestionar la complejidad. La definición de requerimientos funcionales y no funcionales transformó las necesidades del usuario en metas concretas y verificables. La creación de un modelo conceptual y diagrama de casos de uso general permitió establecer una visión compartida y estructurada del sistema, sus actores y sus funcionalidades clave.

La elaboración de los casos de uso narrativos, junto con sus diagramas de secuencia y contratos de operación, fue un ejercicio intensivo pero fundamental. Este proceso descompuso la complejidad del sistema en unidades funcionales manejables, asegurando que cada aspecto del problema original fuese atendido. La matriz de trazabilidad actuó como un mecanismo de control de calidad, garantizando que cada requerimiento funcional tuviera su correspondencia en el diseño del sistema. Este proceso de análisis disciplinado es la respuesta directa a cómo manejar la complejidad inherente al software, transformando una necesidad de negocio en un plano detallado y listo para ser construido.

Finalmente, el capítulo 3, “Aspectos relevantes del diseño”, completó el ciclo de ingeniería al traducir la especificación formal en un plano técnico detallado y listo para la construcción. Este paso fue crucial para materializar la solución, donde la elección de una Arquitectura Hexagonal se destacó como una decisión estratégica para garantizar la mantenibilidad y evolución del sistema, desacoplando la lógica de negocio de la tecnología

subyacente. La misma rigurosidad se aplicó en la selección de tecnologías, como Quarkus y Kubernetes, para asegurar la escalabilidad y eficiencia operativa. Artefactos como los modelos de clases, el modelo relacional y los diagramas de secuencia extendidos dieron forma concreta a los casos de uso definidos en el análisis, mientras que el diseño de la interfaz de usuario fue abordado con igual disciplina, validando su usabilidad mediante las heurísticas de Nielsen para asegurar que la solución no solo fuera funcionalmente correcta, sino también eficiente y satisfactoria para el usuario final.

En definitiva, el trabajo realizado evidencia que, para pasar de una necesidad de negocio a una solución tecnológica viable, el camino debe estar pavimentado con las metodologías y herramientas de la ingeniería de software. Esto demuestra que la calidad del software no es un producto del azar, sino el resultado de un proceso de ingeniería riguroso y bien ejecutado.

BIBLIOGRAFÍA

Pressman, R. S., & Maxim, B. R. (2020). Software Engineering: A practitioner's approach (9th ed., International Student Ed.). McGraw-Hill Education.

Kroll, P., & MacIsaac, B. (2006). Agility and discipline made easy: Practices from OpenUP and RUP. Addison-Wesley Professional.

Universidad de Ciencias Aplicadas de Dresde. (2012). OpenUP Methodology. Disponible en <<https://www2.htw-dresden.de/~anke/openup/about.htm>>

Vieira, D. (2023, September 29). Designing Hexagonal Architecture with Java: An architect's guide to building maintainable and change-tolerant applications with Java and Quarkus. Packt Publishing.

Nielsen, J. (2020, November 15). 10 Usability Heuristics for User Interface Design. Nielsen Norman Group. Disponible en <<https://www.nngroup.com/articles/ten-usability-heuristics>>

IT Hunter. (2024, December 26). 12º Guía Salarial TI 2024 - 2025. Disponible en <<https://www.it-hunter.cl/estudio-de-rentas-ti>>

Kalena. (s.f.). Kalena: Software para la gestión de espacios y recursos. Disponible en <<https://kalena.es>>

ClickBi. (s.f.). Click Visitas: Gestión de accesos y visitas corporativas. Disponible en <<https://clickbi.cl/Click-Visitas.html>>

ANEXOS

Anexo 1: ADR – Mecanismo de auditoría (RNF-01)

A continuación, se presenta la decisión de arquitectura para la implementación del RNF-01 Trazabilidad.

Contexto

El requerimiento no funcional RNF-01 establece que el sistema debe mantener un registro de auditoría de los cambios realizados sobre los datos persistentes, con el objetivo de asegurar trazabilidad, control y respaldo ante eventuales revisiones o análisis posteriores.

Dado el carácter transversal de este requerimiento, la auditoría afecta múltiples casos de uso y entidades del dominio, sin estar asociada directamente a una funcionalidad de negocio específica ni a la intención explícita de los actores del sistema.

Problema

La principal dificultad consiste en incorporar el registro de auditoría sin introducir acoplamiento indebido entre los casos de uso y la lógica técnica de auditoría, evitando duplicación de código en múltiples flujos, contaminación de la lógica de negocio con responsabilidades no funcionales y dependencia directa de mecanismos de infraestructura dentro del núcleo de la aplicación.

Por lo tanto, se requiere una solución que permita garantizar el cumplimiento del RNF-01 de forma consistente, automática y desacoplada.

Alternativas consideradas

1. **Auditoría explícita desde cada caso de uso:** Consistía en invocar manualmente un componente de auditoría desde cada caso de uso que realizara operaciones de persistencia. Esta alternativa fue descartada, ya que incrementa el acoplamiento, duplica lógica y dificulta el mantenimiento.
2. **Triggers a nivel de base de datos:** Consistía en definir disparadores sobre las tablas para registrar automáticamente los cambios. Esta alternativa fue descartada, debido a que desplaza lógica fuera del control de la aplicación, reduce la trazabilidad del origen de los cambios y dificulta su prueba y evolución.
3. **Registro mediante logs técnicos:** Consistía en registrar los cambios únicamente en archivos de log. Esta alternativa fue descartada, ya que no cumple con los requisitos de persistencia estructurada ni facilita la consulta posterior de la información de auditoría.

Decisión adoptada

Se decidió implementar la auditoría como un mecanismo transversal automático, integrado a la capa de persistencia de la aplicación.

La solución consiste en interceptar las operaciones de creación, modificación y eliminación de entidades persistentes, generando de forma automática un evento de auditoría que se almacena en una estructura dedicada. Este mecanismo se ejecuta de manera transparente para los casos de uso, sin requerir invocaciones explícitas desde la lógica de negocio.

De esta forma, el cumplimiento del RNF-01 se garantiza de manera centralizada y consistente en todo el sistema.

Relación con la arquitectura del sistema

La auditoría se implementa como una preocupación transversal (cross-cutting concern), alineada con los principios de la Arquitectura Hexagonal adoptada.

- Los casos de uso no conocen ni gestionan el mecanismo de auditoría.
- El núcleo de la aplicación permanece libre de dependencias técnicas.
- La auditoría se activa automáticamente como consecuencia de las operaciones de persistencia.

En los diagramas de secuencia del sistema, este comportamiento se representa de forma abstracta mediante notas o fragmentos opcionales asociados al RNF-01, evitando modelar detalles específicos de implementación.

Persistencia de la auditoría

Los eventos de auditoría se almacenan en la tabla “registro_auditoria”, la cual permite registrar información relevante como: la entidad afectada, el identificador de la entidad, el tipo de acción ejecutada, el usuario responsable, cuando corresponde, la fecha y hora del evento y un detalle del cambio realizado.

El atributo detalle se modela a nivel lógico como un objeto de texto de gran tamaño, permitiendo almacenar información estructurada o descriptiva según el tipo de evento.

Consecuencias de la decisión

La solución adoptada presenta las siguientes consecuencias positivas:

- Desacoplamiento entre lógica de negocio y requerimientos no funcionales.
- Mayor mantenibilidad y extensibilidad del sistema.
- Cumplimiento uniforme del RNF-01.
- Claridad en los diagramas y modelos, al mantener un nivel de abstracción adecuado.

Como contrapartida, la auditoría introduce un costo adicional de almacenamiento y procesamiento, el cual se considera aceptable dadas las ventajas en trazabilidad y control.

Conclusión

La implementación de la auditoría como un mecanismo transversal automático permite satisfacer el requerimiento no funcional RNF-01 de manera consistente con la arquitectura del sistema, preservando la claridad del diseño, la separación de responsabilidades y la calidad del modelo de software presentado en este trabajo de título.