

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

**SISTEMA PARA LA GESTIÓN AUTOMATIZADA DE ESTACIONAMIENTOS EN G&N
BRANDS**

Trabajo de titulación para optar
al título de Ingeniero de
Ejecución en Software

Estudiante:

Alan Ignacio Badal Yáñez

Profesor:

Carlos Alten López



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: SISTEMA PARA LA GESTIÓN AUTOMATIZADA DE ESTACIONAMIENTOS EN G&N BRANDS

Nombre del candidato(a): ALAN IGNACIO BADAL YÁÑEZ

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, Carlos Alten López, 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: 6 de abril de 2026 Firma: 

Estudiante o Candidato(a):

Fecha: 03/04/2026 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

A mi pareja María José Moya Oyarzo, por ser el pilar fundamental sobre el cual pude completar todo este desafío. Gracias por tu apoyo incondicional, especialmente en los momentos donde el camino se volvió más difícil. Tu valentía al aceptar el desafío de la paternidad a mitad de camino conmigo y por encargarte de que a nuestro hijo nunca le faltara nada mientras yo me dedicaba a mis estudios. Por tu capacidad de transformar nuestra casa en un verdadero hogar donde puedo tener paz, incluso cuando el cansancio te superaba, nunca te quejaste, siempre estuviste ahí, siendo atenta y servicial con nuestro hijo y conmigo. Gracias por impulsarme a ser una mejor versión de mí mismo y recordarme siempre por quién estaba luchando. Este título es tan tuyo como mío.

A mi hijo por llegar a mi vida para darme otro impulso para seguir adelante con todo y mejorar en todos los aspectos de la vida. Tu presencia fue la razón para nunca rendirme y la motivación para buscar un futuro mejor para los tres. Eres la mayor felicidad de mi vida y mi orgullo.

*DEDICATORIA Agradezco a María José e Isaac, por ser mi hogar, mi paz y el motor
inagotable que me permitió alcanzar esta meta. Todo mi esfuerzo es para nuestro
futuro.*

RESUMEN

Keywords: Gestión de estacionamientos, Automatización, Reconocimiento óptico de caracteres (OCR)

El presente informe tiene como propósito proponer e implementar una solución tecnológica que permita automatizar el proceso de reservas de estacionamientos en el Centro de Apoyo Restaurant (CAR), mejorando la eficiencia, la trazabilidad y el uso de los espacios disponibles. En una organización donde los estacionamientos son un recurso limitado, una gestión manual genera múltiples problemas: demoras, errores humanos, falta de control y una baja en la satisfacción de los colaboradores.

En el primer capítulo se exponen los antecedentes, objetivos, análisis del problema y las alternativas de solución evaluadas, dentro de las cuales se optó por desarrollar un sistema a medida. Esta decisión se basó en criterios como la flexibilidad, la adaptabilidad a los procesos propios del CAR y la posibilidad de incorporar funcionalidades específicas solicitadas por los usuarios, como la reasignación automática, la lista de espera dinámica, la validación de acceso mediante OCR, entre otras.

El segundo capítulo aborda los aspectos relevantes del análisis, como la descripción detallada de la solución propuesta, los requerimientos funcionales y no funcionales, el modelo conceptual, los casos de uso generales y específicos, así como los diagramas correspondientes (de actividades, secuencia y casos de uso). Todo este análisis sirvió como base para definir de forma precisa los requerimientos del sistema y asegurar que la solución propuesta sea coherente, trazable y viable desde el punto de vista técnico.

En el tercer capítulo se detalla el diseño arquitectónico y técnico de la solución propuesta. Se describe la implementación de una arquitectura de N-Capas y la aplicación de principios de diseño orientados a asegurar la escalabilidad y el bajo acoplamiento de los componentes del software. Asimismo, se presenta el modelado físico de la base de datos relacional y los diagramas de secuencia que orquestan los flujos críticos del sistema, tales como la lógica de reasignación automática de espacios y la integración con servicios externos de reconocimiento óptico de caracteres (OCR) y pasarelas de

pago. Finalmente, se establecen las pautas de diseño que garantizan una interfaz de usuario intuitiva y funcional para los operadores del centro de apoyo

Como conclusión general, se establece que la automatización del proceso de reserva de estacionamientos no solo resolverá los problemas operacionales actuales, sino que también permitirá al CAR optimizar sus recursos, mejorar la experiencia de los usuarios y sentar las bases para futuras integraciones con tecnologías más avanzadas.

ÍNDICE

INTRODUCCIÓN	1
1 ASPECTOS RELEVANTES DEL PROYECTO Y SU GESTIÓN	3
1.1 Antecedentes de la Empresa u Organización	3
1.2 Situación sin proyecto.....	4
1.2.1 Descripción de la Situación sin Proyecto	5
1.2.2 Problemas detectados	13
1.2.3 Requerimientos y Requisitos del Usuario.....	14
1.3 Objetivos	16
1.4 Alternativas globales de solución	17
1.4.1 Alternativa 1: Desarrollo a Medida (interno).....	17
1.4.2 Alternativa 2: Software Comercial (COTS) - ParkFacil.....	18
1.4.3 Alternativa 3 Software de Código Abierto – Parking Management System	20
1.5 Evaluación de Alternativas.....	22
1.5.1 Criterios para evaluar las alternativas	22
1.5.2 Ponderación de Criterios y Escala de Evaluación	23
1.5.3 Evaluación	23
1.6 Alternativa seleccionada.....	27
1.6.1 Descripción	27
1.6.2 Beneficios	30
1.7 Planificación	31
1.7.1 Desarrollo del plan de personal	31
1.7.2 Estimación de esfuerzo.....	33
1.7.3 Planificación temporal, y asignación de recursos (Carta Gantt)	34
1.7.4 Estimación de costos.....	35
1.7.5 Administración del riesgo.....	39
2 ASPECTOS RELEVANTES DEL ANÁLISIS.....	48
2.1 Descripción de la solución propuesta.....	48
2.2 Diagrama de actividades general UML (Flujo Administrativo)	49
2.3 Requerimientos del Sistema	52
2.4 Condicionantes del Diseño Tecnológico.....	61
2.5 Estructura funcional del sistema (Caso de Uso General)	62
2.5.1 Actores involucrados	63
2.5.2 Casos de uso	64
2.6 Matriz de Trazabilidad	66
2.7 Modelo Conceptual	68
2.8 Modelo de Casos de Uso.....	71
2.8.1 Caso de Uso CU-01: Crear usuario.....	71

2.8.2	Caso de Uso CU-02: Mantener usuario	74
2.8.3	Caso de Uso CU-03: Mantener reserva	77
2.8.4	Caso de uso CU-04: Realizar pago online por visita	83
2.8.5	Caso de uso CU-05: Validar acceso mediante OCR	87
2.8.6	Caso de uso CU-06: Emitir notificación	89
2.8.7	Caso de uso CU-07: Generar reporte	92
2.8.8	Caso de uso CU-08: Inicio de sesión	94
2.8.9	Caso de uso CU-09: Mantener penalización	97
3	ASPECTOS RELEVANTES DEL DISEÑO	102
3.1	Arquitectura del software	102
3.1.1	Plataforma tecnológica y herramientas de desarrollo	108
3.2	Diseño de Datos	111
3.2.1	Modelo de Clases	111
3.2.2	Modelo Relacional	113
3.2.3	Diccionario de Datos	115
3.3	Diagramas de secuencia extendido	130
3.3.1	Diagrama de Secuencia Extendido: CU-01 Crear usuario	131
3.3.2	Diagrama de Secuencia Extendido: CU-02 Mantener usuario	132
3.3.3	Diagrama de Secuencia Extendido: CU-03 Mantener reserva	133
3.3.4	Diagrama de Secuencia Extendido: CU-03.5 Reasignar estacionamiento liberado	134
3.3.5	Diagrama de Secuencia Extendido: CU-04 Realizar pago online por visita	135
3.3.6	Diagrama de Secuencia Extendido: CU-05 Validar acceso mediante OCR	136
3.3.7	Diagrama de Secuencia Extendido: CU-06 Recibir notificaciones	137
3.3.8	Diagrama de Secuencia Extendido: CU-07 Generar reporte	138
3.3.9	Diagrama de Secuencia Extendido: CU-08 Inicio de sesión	139
3.3.10	Diagrama de Secuencia Extendido: CU-09 Mantener penalización	140
3.4	Diseño de Interfaz	141
	PROTOTIPO DE INTERFAZ	142
3.4.1	Diagrama de Menú o de Navegación	142
3.4.2	Diseño de interfaz de la página principal	144
3.4.3	Diseño de Interfaz CU-09 Iniciar sesión	145
3.4.4	Diseño de interfaz CU-03 Mantener reserva	146
4	CONCLUSIONES	148
5	BIBLIOGRAFÍA	150
6	ANEXOS	¡ERROR! MARCADOR NO DEFINIDO.

ÍNDICE DE TABLAS

Tabla 1-1 Ejemplo de registro en Excel del encargado	12
Tabla 1-2 Ponderación de los criterios de evaluación	23
Tabla 1-3 Escala de Calificación	23
Tabla 1-4 Calificación de la Alternativa 1	23
Tabla 1-5 Calificación de la Alternativa 2	24
Tabla 1-6 Calificación de la Alternativa 3	25
Tabla 1-7 Evaluación de Alternativas: Resumen.....	26
Tabla 1-8 Estimación de esfuerzo por modulo y complejidad	33
Tabla 1-9 Carta Gantt del Proyecto	34
Tabla 1-10 Estructuras de Rentas Brutas	36
Tabla 1-11 Costos de los recursos.....	36
Tabla 1-12 Costos totales del proyecto.	37
Tabla 1-13 Distribución de esfuerzo entre requisitos	37
Tabla 1-14 Esfuerzo de actividades de no-desarrollo	38
Tabla 1-15 Resumen consolidad de esfuerzo	38
Tabla 1-16 Distribución del esfuerzo en actividades de la ingeniería de software.....	39
Tabla 1-17 Distribución de esfuerzo por rol y actividad de ingeniería de software (HH)	39
Tabla 1-18 Administración de Riesgos	41
Tabla 1-19 Plan de mitigación de riesgos	42
Tabla 2-1 Requerimientos Funcionales.....	53
Tabla 2-2 Requerimientos No Funcionales	56
Tabla 2-3 Matriz de trazabilidad.....	67
Tabla 2-4 Descripción de entidades del modelo conceptual	68
Tabla 2-5 Descripción narrativa CU-01 Crear usuario	71
Tabla 2-6 Contrato CU-01 Crear usuario.	73
Tabla 2-7 Descripción narrativa CU-02 Mantener usuario.	74
Tabla 2-8 Contrato CU-02 Mantener usuario	76
Tabla 2-9 Descripción narrativa CU-03 Mantener reserva	77
Tabla 2-10 Contrato CU-03 Mantener reserva	82
Tabla 2-11 Descripción narrativa CU-04 Realizar pago online por visita	83
Tabla 2-12 Contrato CU-04 Realizar pago online por visita	86
Tabla 2-13 Descripción narrativa CU-05 Validar acceso mediante OCR	87
Tabla 2-14 Contrato CU-05 Validar acceso mediante OCR	88
Tabla 2-15 Descripción narrativa CU-06 Emitir notificación	89
Tabla 2-16 Contrato CU-06 Recibir notificaciones.....	91
Tabla 2-17 Descripción narrativa CU-07 Generar reporte	92
Tabla 2-18 Contrato CU-07 Generar reporte.....	94
Tabla 2-19 Descripción narrativa CU-08 Inicio de sesión.....	94

Tabla 2-20 Contrato CU-08 Inicio de sesión	96
Tabla 2-21 Descripción narrativa CU-09 Mantener penalización	97
Tabla 2-22 Contrato CU-09 Mantener Penalización.....	99
Tabla 3-1 Tabla TipoEstacionamiento	115
Tabla 3-2 Tabla Estacionamiento	115
Tabla 3-3 Tabla Pago	116
Tabla 3-4 Tabla DetalleReserva	117
Tabla 3-5 Tabla EstadoReserva	118
Tabla 3-6 Tabla TransferenciaReserva	118
Tabla 3-7 Tabla Reserva	119
Tabla 3-8 Tabla Notificacion	120
Tabla 3-9 Tabla ValidacionOCR	121
Tabla 3-10 Tabla Reporte	122
Tabla 3-11 Tabla ListaEspera	123
Tabla 3-12 Tabla Trabajador	124
Tabla 3-13 Tabla Penalizacion	124
Tabla 3-14 Tabla Persona	125
Tabla 3-15 Tabla DetalleTrabajador	126
Tabla 3-16 Tabla EstadoTrabajador.....	127
Tabla 3-17 Estado Penalizacion	127
Tabla 3-18 Tabla DetallePenalizacion	127
Tabla 3-19 Tabla ReporteReserva	128
Tabla 3-20 Tabla ReportePenalizacion	129

ÍNDICE DE FIGURAS

Figura 1-1 Diagrama de Flujo: Situación Actual	11
Figura 2-1 Diagrama de actividades flujo administrativo (actual)	52
Figura 2-2 Diagrama de caso de uso general	62
Figura 2-3 Modelo conceptual	70
Figura 2-4 Diagrama de secuencia CU-01 Crear usuario.....	73
Figura 2-5 Diagrama de secuencia CU-02 Mantener usuario.	76
Figura 2-6 Diagrama de secuencia CU-03 Mantener reserva	81
Figura 2-7 Diagrama de secuencia CU-03.5 Reasignar estacionamiento liberado.....	82
Figura 2-8 Diagrama de secuencia CU-04 Realizar pago online por visita ..	86
Figura 2-9 Diagrama de secuencia CU-05 Validar acceso mediante OCR	88
Figura 2-10 Diagrama de secuencia CU-06 Recibir notificaciones	91
Figura 2-11 Diagrama de secuencia CU-07 Generar reporte.....	93
Figura 2-12 Diagrama de secuencia CU-08 Inicio de sesión	96
Figura 2-13 Diagrama de secuencia CU-09 Mantener Penalización.....	99
Figura 3-1 Diagrama de Arquitectura del Software.....	103

Figura 3-2 Diagrama de despliegue.....	106
Figura 3-3 Modelo de Clases.....	112
Figura 3-4 Modelo Relacional.....	114
Figura 3-5 Diagrama de secuencia extendido CU-01 Crear Usuario	131
Figura 3-6 Diagrama de secuencia extendido CU-02 Mantener usuario	132
Figura 3-7 Diagrama de secuencia extendido CU-03 Mantener reserva	133
Figura 3-8 Diagrama de secuencia extendido CU-03.5 Reasignar estacionamiento liberado.....	134
Figura 3-9 Diagrama de secuencia extendido CU-04 Realizar pago online por visita .	135
Figura 3-10 Diagrama de secuencia extendido CU-05 Validar acceso mediante OCR.	136
Figura 3-11 Diagrama de secuencia extendido CU-06 Recibir notificaciones	137
Figura 3-12 Diagrama de secuencia extendido CU-07 Generar reporte	138
Figura 3-13 Diagrama de secuencia extendido CU-08 Inicio de sesión	139
Figura 3-14 Diagrama de secuencia extendido CU-09 Mantener penalización.....	140
Figura 3-15 Diagrama de menú del sistema	143
Figura 3-16 Diseño de interfaz de la página principal del sistema.....	144
Figura 3-17 Diseño interfaz CU-09 Inicio de sesión	145
Figura 3-18 Diseño de interfaz CU-03 Mantener reserva	147

ÍNDICE DE IMGENES

Ilustración 1 Oficina CAR G&N BRANDS 1	7
Ilustración 2 Oficina CAR G&N BRANDS 2	8
Ilustración 3 Estacionamiento CAR G&N BRANDS 1	9
Ilustración 4 Estacionamiento CAR G&N BRANDS 2	9
Ilustración 5 Estacionamiento CAR G&N BRANDS 3	10
Ilustración 6 Estacionamiento CAR G&N BRANDS 4	10

SIGLAS Y SÍMBOLOS

Sigla	Descripción
CAR	Centro Apoyo Restaurant.
QA	Quality Assurance (Asegurador de calidad).
JP	Jefe de Proyecto.
AP	Analista Programador.
DFS	Desarrollador Full Stack.
UI/UX	User Interface / User Experience (Interfaz de usuario / Experiencia de usuario) (Diseñador).
UF	Unidad de fomento.
LDC	Líneas de código.
MLC	Miles de líneas de código.
PF	Puntos de función.
PFNA	Puntos de función no ajustados.
PFA	Puntos de función ajustados.
OCR	Optical Character Recognition, tecnología que permite el reconocimiento automático de texto o números, en este caso usado para leer patentes.
Log	Registro interno del sistema que almacena información sobre acciones realizadas por los usuarios o el sistema mismo. Útil para trazabilidad y auditoria.
Back-end	Parte lógica del sistema que procesa los datos y gestiona las operaciones en el servidor
Front-end	Interfaz visual del sistema con la que interactúa el usuario (navegador o app móvil.)
Pasarela de pago	Plataforma externa que permite realizar pagos en línea de forma segura. Ejemplo: WebPay, MercadoPago.
JMeter	Herramienta de software utilizada para pruebas de rendimiento, carga y concurrencia. (open source).
Open Source	Se refiere al código abierto, que significa que está a disponibilidad de la comunidad y no se debe pagar nada.

INTRODUCCIÓN

La gestión eficiente de los recursos es un aspecto clave para el correcto funcionamiento de cualquier organización, independientemente de su tamaño o sector. Entre estos recursos, los espacios de estacionamiento adquieren una relevancia particular en entornos corporativos, donde su disponibilidad suele ser limitada. Una asignación ineficaz puede derivar en problemas operacionales, incomodidad para los colaboradores y un uso ineficiente del espacio disponible.

Actualmente, muchas instituciones, como el Centro de Apoyo Restaurant (CAR), llevan a cabo la gestión de estacionamientos de forma manual, mediante planillas y comunicación por correo electrónico. Esta práctica conlleva diversas desventajas: falta de visibilidad en tiempo real, posibles errores humanos, demoras en la asignación de espacios y ausencia de trazabilidad o historial de uso. Además, se desaprovechan oportunidades para reutilizar espacios cancelados o no utilizados, lo que repercute directamente en la productividad y satisfacción de los usuarios.

En contraste, la implementación de un sistema automatizado para la gestión de estacionamientos permite optimizar la asignación de espacios, reducir errores, ofrecer mayor flexibilidad y mejorar la experiencia del usuario. Asimismo, permite incorporar funcionalidades avanzadas como la reasignación automática, la validación de accesos mediante OCR, notificaciones automáticas y reportes de uso.

Este informe tiene como propósito analizar la situación actual, definir los requerimientos funcionales y no funcionales, y proponer una solución tecnológica orientada a resolver los problemas detectados. A lo largo del documento se abordarán tanto los aspectos estratégicos como los técnicos, desde la evaluación de alternativas hasta la modelación del sistema, con el fin de asegurar la viabilidad y efectividad de la solución propuesta.

CAPITULO 1. ASPECTOS RELEVANTES DEL PROYECTO Y SU GESTION

1 ASPECTOS RELEVANTES DEL PROYECTO Y SU GESTIÓN

En este capítulo se presentan los antecedentes del proyecto, incluyendo el contexto organizacional, la problemática identificada, los objetivos planteados y las alternativas de solución consideradas. Además, se detallan los requerimientos del usuario, la planificación general y la estimación de recursos necesarios para su ejecución.

1.1 Antecedentes de la Empresa u Organización

G&N Brands es un holding gastronómico con presencia a nivel nacional, cuya misión y estructura organizacional buscan optimizar los procesos de sus marcas afiliadas [1]. Con más de 30 años de trayectoria en el rubro gastronómico y de franquicias. Desde su fundación en los años 90 con Doggis, su primera marca en el centro de Santiago, la compañía ha experimentado un crecimiento significativo. Actualmente, ha consolidado un robusto portafolio que incluye reconocidas marcas como Juan Maestro, Mamut, Tommy Beans, Blo-Ke Burger, Lovdo y Barrio Chicken, posicionándose como un actor relevante en el dinámico sector alimentario chileno.

El Centro de Apoyo Restaurant (CAR) se identifica como el usuario objetivo que es la oficina central de G&N Brands, situada en Coyancura 2283, Providencia. Este centro alberga aproximadamente a 100 personas, quienes desarrollan las operaciones administrativas y de soporte de la empresa. La problemática de gestión ineficiente de estacionamientos en el CAR afecta a todos los departamentos y áreas que operan en el edificio, incluyendo a RRHH, Marketing, TI, Construcción, Operación y el resto. Esta situación impacta directamente en la satisfacción y productividad de los colaboradores de estas diversas áreas ya que se producen favoritismos, errores en gestionar los estacionamientos, entre otros problemas.

La misión de G&N Brands se declara como: "Somos líderes en el rubro gastronómico y de franquicias, a través de nuestras reconocidas marcas entregamos experiencias diferenciadoras y de calidad para nuestros clientes. Somos la primera opción de empleo para muchos jóvenes, como también una atractiva forma de emprender en nuevos proyectos desarrollando lo que puede ser tu primer negocio, con frescas, deliciosas, accesibles y novedosas opciones gastronómicas, cautivando a diferentes tipos de consumidores. Siempre comprometidos con la calidad en nuestros procedimientos, innovando constantemente con mucha pasión en todo lo que hacemos".

Por su parte, **la visión** de la empresa aspira a "Ser una empresa reconocida en Latinoamérica por sus deliciosas y rentables experiencias gastronómicas, incorporando constantemente propuestas sustentables e innovadoras en nuestra operación, comprometidos con el desarrollo de un servicio de excelencia en todas nuestras marcas".

El sistema para la gestión automatizada de estacionamientos en G&N Brands se alinea directamente con esta misión y visión corporativa. Al abordar la optimización de un recurso interno limitado como los estacionamientos, el sistema propuesto contribuye a mejorar la experiencia y la calidad de vida de los colaboradores, reconociéndolos como parte de las "experiencias diferenciadoras y de calidad" que la empresa busca entregar. Asimismo, la implementación de una solución tecnológica que automatiza y eficientiza un proceso previamente manual refleja el compromiso de G&N Brands con la "innovación" y la incorporación de "propuestas sustentables e innovadoras en nuestra operación". De este modo, el proyecto busca un "servicio de excelencia" no solo hacia sus clientes externos, sino también en la gestión interna de sus recursos y el bienestar de los colaboradores, sentando las bases para futuras integraciones con tecnologías más avanzadas.

Actualmente, la gestión de los estacionamientos en el CAR se realiza de manera manual, con procesos que involucran el envío de correos electrónicos y el registro en planillas Excel. Esta dependencia de un proceso manual representa el punto de partida tecnológico que el presente proyecto busca transformar, evidenciando la necesidad de una solución automatizada para mejorar la eficiencia, la trazabilidad y la optimización de este recurso limitado.

1.2 Situación sin proyecto

En la actualidad, la gestión manual de estacionamientos en organizaciones con recursos limitados provoca conflictos, retrasos y un uso ineficiente de los espacios. La falta de automatización y la dependencia de procesos manuales agravan esta situación, afectando la operación diaria y la experiencia de los usuarios. A continuación, se detalla la situación actual del CAR de G&N Brands.

1.2.1 Descripción de la Situación sin Proyecto

Actualmente, la gestión de los estacionamientos en el CAR de G&N Brands se realiza de manera manual a través del envío de correos electrónicos. Los colaboradores interesados en reservar un espacio deben enviar un correo a la persona encargada (administrador de las reservas), quien responde indicando si existe disponibilidad y asignando un código de estacionamiento (por ejemplo, A23). El proceso depende únicamente del orden de llegada de los correos electrónicos y carece de mecanismos que permitan una administración eficiente de los espacios disponibles. Esta situación genera diversos problemas que afectan la operación diaria y la experiencia de los usuarios de todas las áreas y departamentos que operan en el edificio.

En primer lugar, la falta de automatización en el proceso de reserva provoca ineficiencia en la gestión y retrasos en la confirmación de las solicitudes. La persona encargada debe revisar manualmente cada correo, lo que consume tiempo valioso y genera demoras que podrían evitarse con un sistema automatizado. Además, debido a que esta persona tiene otras responsabilidades dentro de la organización, no siempre puede responder con la agilidad que los empleados requieren.

Por otro lado, la ausencia de un mecanismo de cancelación impide que los espacios reservados sean reutilizados en caso de que un empleado no utilice su reserva. Esto ocasiona un uso ineficaz de los estacionamientos, ya que los espacios quedan vacíos y no puedan asignarse a otros usuarios que realmente los necesitan.

Asimismo, el proceso actual carece de visibilidad en tiempo real sobre la disponibilidad de estacionamientos. Los empleados no pueden consultar de antemano si existe un espacio libre, lo que obliga a depender de la confirmación manual por parte de la persona encargada. Esta falta de información provoca incertidumbre y dificulta la planificación de los empleados.

Otro asunto es la falta de flexibilidad en el proceso de reservas. Los empleados deben realizar sus solicitudes los viernes para la semana siguiente, lo que no permite atender necesidades imprevistas que puedan surgir durante la semana. Esta rigidez en el proceso limita la capacidad de respuesta ante situaciones de emergencia o cambios de último minuto.

Adicionalmente, la gestión manual aumenta el riesgo de errores humanos, como la asignación duplicada de un mismo espacio o la omisión accidental de solicitudes de reserva. Este tipo de errores afecta la satisfacción de los empleados y genera conflictos que podrían evitarse con un sistema más robusto y automatizado.

Finalmente, el proceso manual también carece de criterios avanzados de asignación. Actualmente, los estacionamientos se asignan únicamente por orden de llegada de los correos, sin considerar otros factores como la rotación de usuarios, la frecuencia de uso o las necesidades específicas de los empleados.

En resumen, la gestión actual de estacionamientos en el CAR de G&N Brands se caracteriza por la dependencia de procesos manuales, la limitada visibilidad en tiempo real y la marcada rigidez en sus operaciones para el proceso de reserva. Estas particularidades de la gestión se traducen en un uso ineficiente de los espacios y un impacto directo en la experiencia diaria de los colaboradores, lo que subraya la necesidad de optimizar la administración de estos recursos cruciales.

En las siguientes ilustraciones se muestran algunas partes de la oficina del CAR de G&N Brands donde se encuentran los trabajadores realizando sus labores.

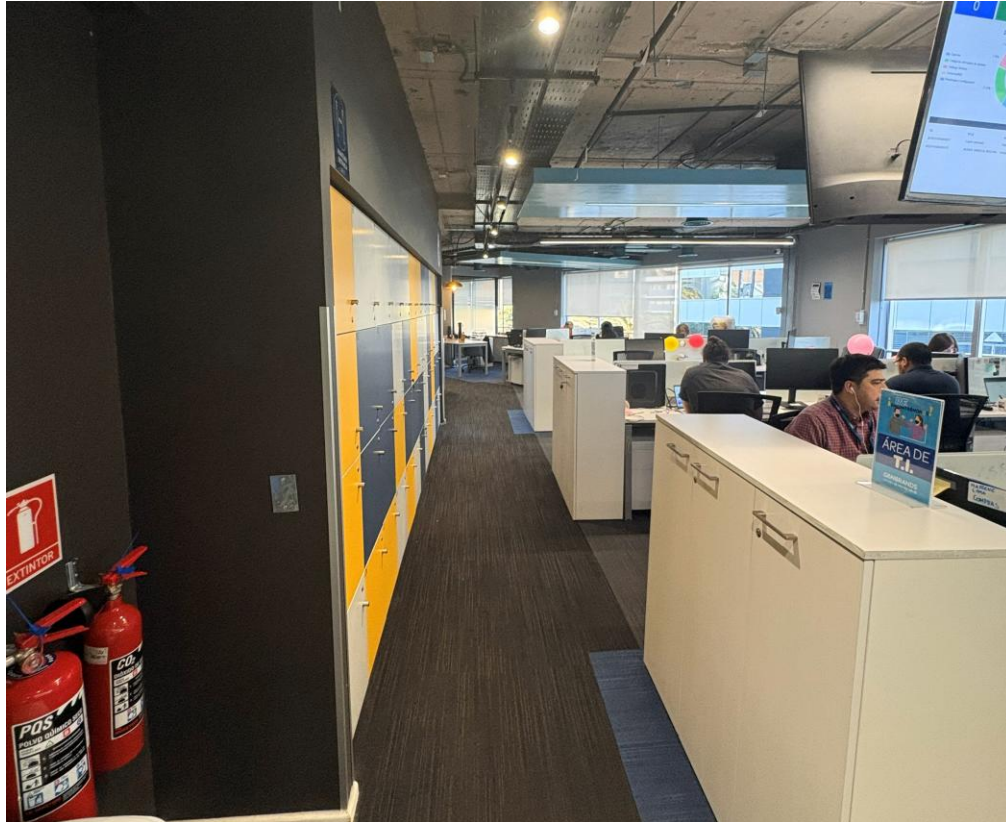


Ilustración 1 Oficina CAR G&N BRANDS 1

Fuente: Elaboración propia.



Ilustración 2 Oficina CAR G&N BRANDS 2

Fuente: Elaboración propia.

Las siguientes ilustraciones muestran la disposición física y el nivel de ocupación habitual de los estacionamientos. Se observa una alta densidad vehicular y una distribución estática de las plazas, donde la única forma de verificar la disponibilidad es mediante la inspección visual directa por parte del conductor. La ausencia de señalética dinámica o sensores de presencia provoca que los usuarios deban recorrer los pasillos innecesariamente en busca de un espacio libre.



Ilustración 3 Estacionamiento CAR G&N BRANDS 1

Fuente: Elaboración propia.

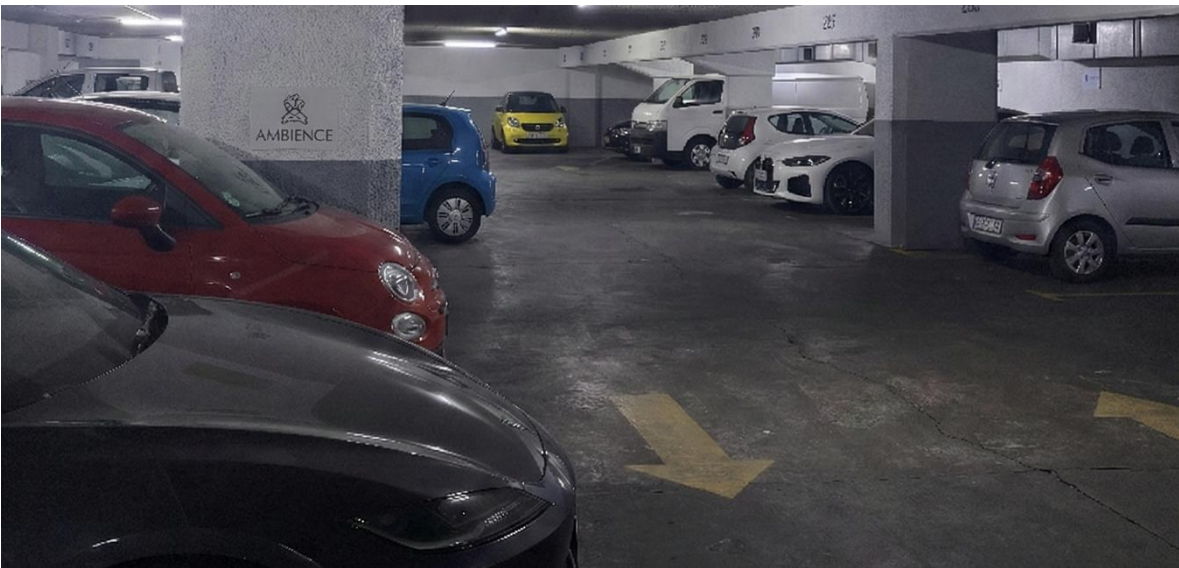


Ilustración 4 Estacionamiento CAR G&N BRANDS 2

Fuente: Elaboración propia.

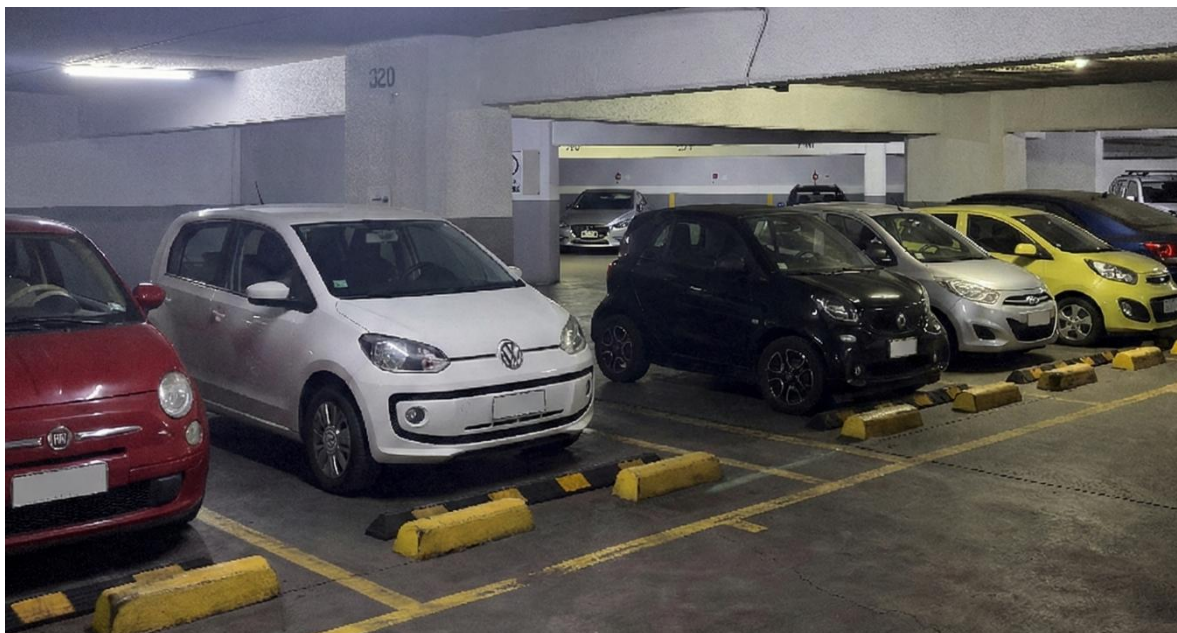


Ilustración 5 Estacionamiento CAR G&N BRANDS 3
Fuente: Elaboración propia.

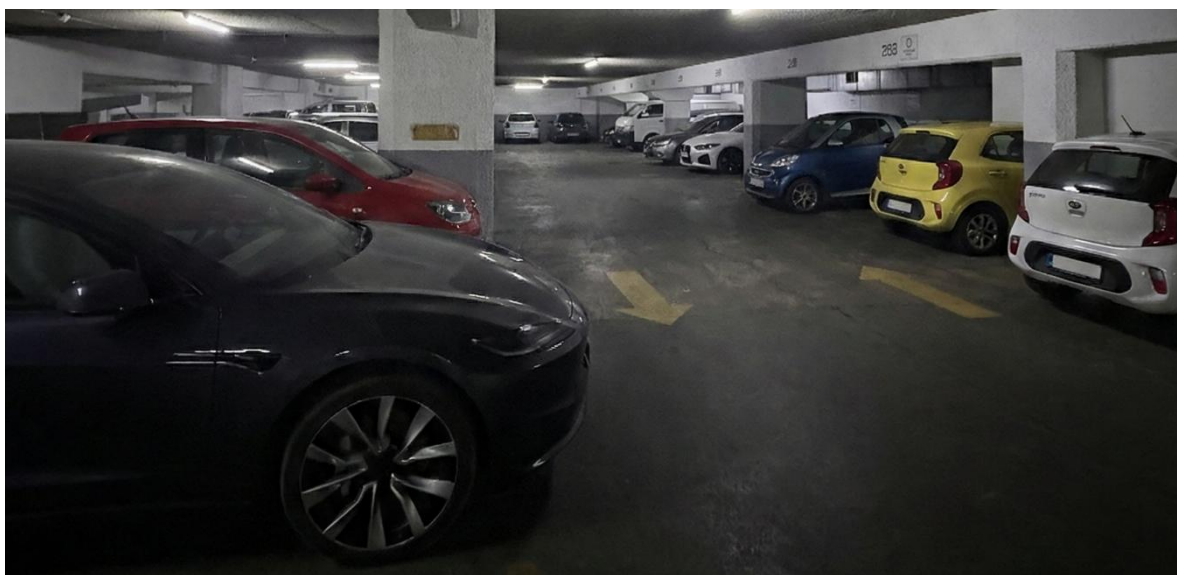


Ilustración 6 Estacionamiento CAR G&N BRANDS 4
Fuente: Elaboración propia.

Se adjunta un diagrama de flujo y un ejemplo de cómo se lleva el registro en Excel de las reservas de estacionamientos para entender de mejor manera la situación actual de la organización y como se reservan los estacionamientos.

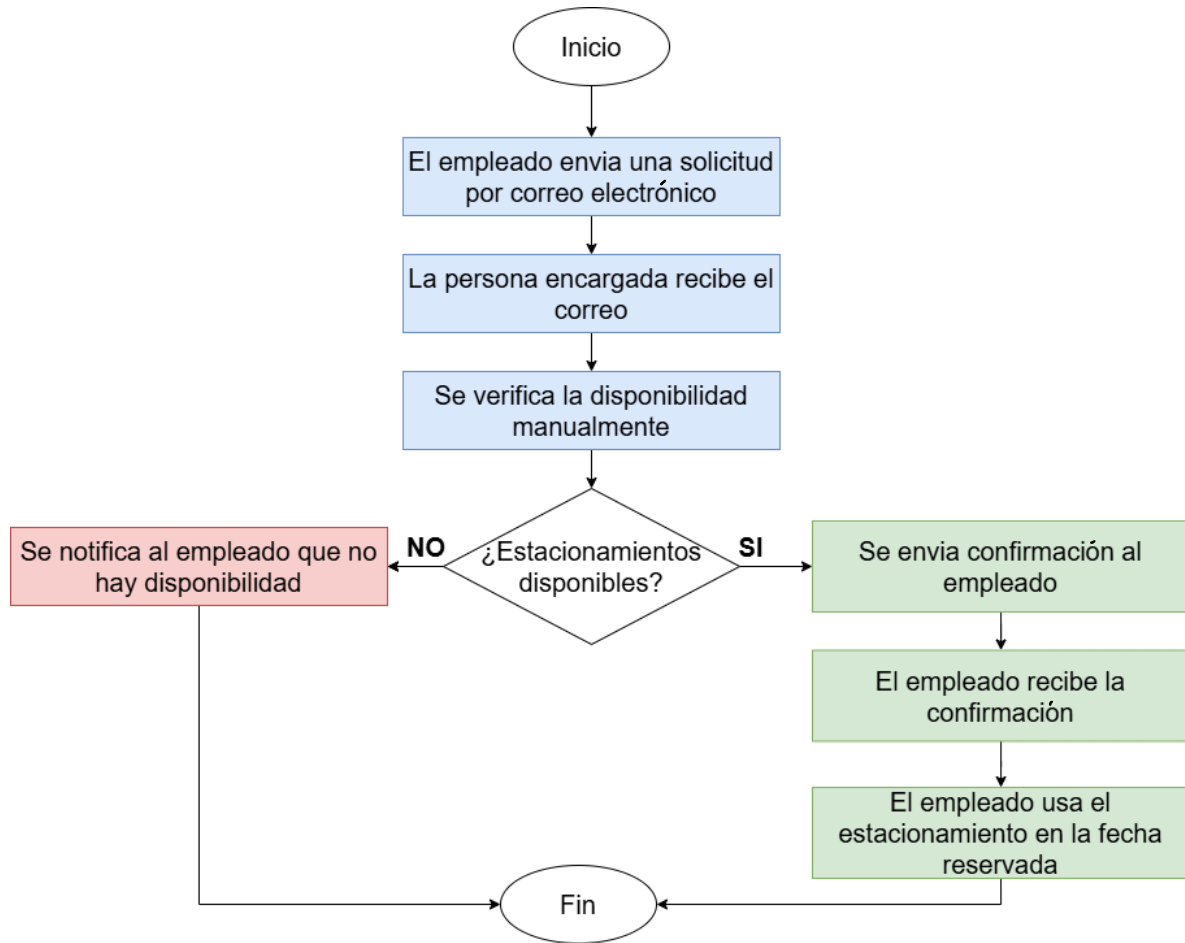


Figura 1-1 Diagrama de Flujo: Situación Actual

Fuente: Elaboración propia.

En la Tabla 1-1 se adjunta un ejemplo de cómo es el registro en Excel que lleva el encargado de las reservas.

Tabla 1-1 Ejemplo de registro en Excel del encargado

Empleado	Código Estacionamiento	Fecha Reserva	Hora Inicio	Hora Fin	Estado Confirmación	Asistencia
Empleado 4	A12	4/7/2024	08:30	17:30	Confirmada	Sí
Empleado 7	A05	8/7/2024	07:00	18:00	Pendiente	No
Empleado 2	A15	10/7/2024	09:00	17:30	Confirmada	Sí
Empleado 5	A20	12/7/2024	08:00	18:30	Rechazada	No
Empleado 3	A25	13/7/2024	07:30	18:00	Confirmada	Sí
Empleado 8	A02	14/7/2024	09:00	17:30	Pendiente	No
Empleado 9	A08	15/7/2024	08:00	18:00	Confirmada	Sí
Empleado 1	A18	16/7/2024	07:30	18:30	Rechazada	No
Empleado 10	A10	18/7/2024	08:30	17:00	Confirmada	Sí
Empleado 4	A28	19/7/2024	07:00	17:30	Pendiente	No
Empleado 3	A22	20/7/2024	09:00	18:30	Confirmada	Sí
Empleado 6	A14	21/7/2024	07:30	18:00	Confirmada	Sí
Empleado 7	A17	22/7/2024	08:00	18:30	Rechazada	No
Empleado 5	A06	23/7/2024	09:00	17:30	Pendiente	Sí
Empleado 8	A13	24/7/2024	07:30	18:00	Confirmada	Sí
Empleado 2	A19	25/7/2024	08:30	18:30	Pendiente	No
Empleado 9	A01	26/7/2024	07:00	17:00	Confirmada	Sí
Empleado 10	A09	27/7/2024	08:00	18:30	Rechazada	No
Empleado 6	A11	28/7/2024	07:30	18:00	Confirmada	Sí
Empleado 4	A23	29/7/2024	08:00	17:30	Confirmada	Sí
Empleado 3	A03	30/7/2024	07:30	18:00	Pendiente	No
Empleado 7	A16	31/7/2024	08:30	18:30	Confirmada	Sí
Empleado 5	A24	5/7/2024	07:00	17:30	Confirmada	Sí
Empleado 1	A21	6/7/2024	08:00	18:00	Rechazada	No
Empleado 9	A27	7/7/2024	07:30	18:30	Confirmada	Sí

Fuente: Elaboración propia.

1.2.2 Problemas detectados

Dada la descripción realizada, es posible identificar varios problemas, entre los cuales destacan los siguientes:

1. **Falta de automatización en el proceso de reserva:** Las reservas se gestionan manualmente a través de correos electrónicos, lo que genera ineficiencia y posibles retrasos en la asignación de los estacionamientos. Esto puede ocasionar molestias entre los colaboradores, traducándose en una posible baja productividad.
2. **Retrasos en la confirmación de reservas:** La persona encargada de gestionar los estacionamientos tiene otros deberes en la organización, lo que provoca demoras en responder las solicitudes de reserva.
3. **Reserva no reutilizable:** Si un empleado reserva un estacionamiento y no se presenta, el espacio queda vacío y no se asigna a otro empleado que podría necesitarlo, generando un uso ineficiente de los estacionamientos.
4. **Falta de cancelación de reservas:** No existe un mecanismo que permita a los empleados cancelar sus reservas si ya no las necesitan, lo que podría liberar espacios para otros usuarios.
5. **Orden de llegada como único criterio de asignación:** La asignación de estacionamientos se basa exclusivamente en el orden en que llegan los correos, sin considerar otros posibles criterios de prioridad o rotación.
6. **Poca flexibilidad para realizar reservas:** Las reservas se realizan con una semana de antelación, lo que no permite atender necesidades imprevistas de los empleados.
7. **No hay visibilidad sobre la disponibilidad en tiempo real:** Los empleados no tienen un sistema para consultar la disponibilidad de estacionamientos en tiempo real, lo que podría evitar la necesidad de esperar confirmación por correo.
8. **Posibles errores humanos en la gestión manual:** La gestión de correos y asignaciones de manera manual puede generar errores, como la asignación duplicada de un mismo estacionamiento o el no procesamiento a tiempo de todas las solicitudes.
9. **Ineficiencia en el uso de los estacionamientos:** Con solo 30 estacionamientos disponibles, una mala gestión o falta de cancelación de reservas desperdicia espacios que podrían ser utilizados por otros empleados.

1.2.3 Requerimientos y Requisitos del Usuario

Tomando como base los problemas detectados y las necesidades del usuario, se establecen los requerimientos específicos que debe cumplir la solución. Estos requisitos tienen como objetivo resolver las deficiencias identificadas en la situación actual, asegurando una gestión eficiente, automatizada y flexible de las reservas de estacionamientos, optimizando los recursos disponibles y mejorando la experiencia del usuario. Cabe destacar que los requerimientos expuestos a continuación se describen a un alto nivel de abstracción, por lo que, definen las necesidades del negocio y las funcionalidades esperadas sin profundizar en los detalles técnicos de bajo nivel (como los tipos de datos, protocolos de autenticación o arquitectura), los cuales serán especificados en las etapas posteriores de análisis y diseño.

1. **Automatización del proceso de reservas:** El sistema deberá gestionar automáticamente las reservas de estacionamientos, eliminando el uso de correos electrónicos y asignando de forma inmediata los espacios disponibles.
2. **Sistema de confirmación automática:** El sistema deberá confirmar automáticamente las reservas realizadas por los usuarios, notificándolos en tiempo real.
3. **Política de cancelación y reasignación:** El sistema deberá permitir la cancelación de reservas por parte del usuario y reasignar automáticamente los espacios liberados a otros usuarios en lista de espera o según disponibilidad.
4. **Criterios avanzados para asignación de estacionamientos:** El sistema deberá asignar los estacionamientos considerando factores como orden de solicitud, rotación de uso y prioridad.
5. **Acceso a la disponibilidad en tiempo real:** El sistema deberá permitir la visualización en tiempo real de la disponibilidad de estacionamientos, desde cualquier dispositivo con acceso a internet que pueda ejecutar el sistema.
6. **Gestión eficiente de cancelaciones:** El sistema deberá permitir a los usuarios cancelar sus reservas de forma sencilla y con suficiente anticipación.
7. **Flexibilidad en la reserva:** El sistema deberá permitir realizar reservas con plazos reducidos, incluyendo la opción de reservar el mismo día.
8. **Reducción de errores humanos:** El sistema deberá reducir la intervención manual en la gestión de reservas, evitando errores como duplicidad o ausencia de asignación.

9. **Reportes de uso y métricas de ocupación:** El sistema deberá permitir generar reportes sobre el uso de los estacionamientos, incluyendo métricas de ocupación diaria y mensual.
10. **Notificaciones automáticas:** El sistema deberá enviar notificaciones automáticas a los usuarios ante eventos como confirmaciones, cancelaciones o reasignaciones de reservas.
11. **Compatibilidad multiplataforma y facilidad de uso:** El sistema deberá soportar su ejecución de manera responsiva tanto en dispositivos móviles como de escritorio. Además, su interfaz gráfica deberá estar optimizada para que el usuario pueda completar una reserva en menos de tres pasos lógicos
12. **Estacionamientos preferenciales:** El sistema deberá permitir reservar estacionamientos designados como preferenciales para usuarios con esa condición.
13. **Inicio de sesión:** El sistema deberá permitir a los colaboradores autenticarse mediante credenciales registradas previamente.

1.3 Objetivos

A continuación, se define el propósito central del proyecto y las etapas metodológicas necesarias para garantizar su cumplimiento.

Objetivo General

Mejorar la eficiencia y transparencia en la gestión de reservas de estacionamientos del personal del CAR de la empresa G&N, mediante la implementación de un sistema informático que permita una asignación equitativa y oportuna de los espacios disponibles.

Objetivos Específicos

1. Analizar el proceso actual de asignación de estacionamientos para identificar los cuellos de botella y formalizar los requerimientos funcionales y no funcionales de la nueva solución.
2. Diseñar la arquitectura de software y el modelo de datos del sistema, asegurando el cumplimiento de las reglas de negocio, la escalabilidad y la adaptabilidad de la infraestructura tecnológica de la empresa.
3. Desarrollar un sistema informático mediante un enfoque incremental, construyendo los módulos de gestión de usuarios, motor de reservas y lógica de reasignación automática.
4. Validar la solución tecnológica a través de pruebas de aseguramiento de calidad (QA) y marchas blancas, garantizando la correcta integración de los componentes y la usabilidad de la interfaz antes de su despliegue final.

1.4 Alternativas globales de solución

A partir de esto se describen las posibles alternativas de solución para los problemas detectados en el punto 1.2.2, también así tratando de cumplir con los objetivos específicos y general mencionados en el punto 1.3. De las alternativas se mencionan las ventajas, desventajas, funcionamiento y algunos criterios que ayudaran a elegir la mejor alternativa que contribuya a solucionar el problema que presenta el CAR de G&N Brands con su gestión de estacionamientos.

1.4.1 Alternativa 1: Desarrollo a Medida (interno)

Esta alternativa de solución considera la creación de un sistema de gestión de reservas de estacionamientos diseñado específicamente para satisfacer las necesidades del CAR de G&N Brands. Esta opción garantiza que el sistema cumpla con todos o la mayoría de los requerimientos y requisitos, ya que su funcionalidad es desarrollada desde cero para cubrir cada especificación establecida en los puntos anteriores. Además, este enfoque permite una alta escalabilidad, lo que asegura que el sistema pueda adaptarse a futuros cambios o aumentar en complejidad, permitiendo la incorporación de nuevas funcionalidades conforme evolucionen las necesidades de la organización. Esto también facilita la mantención, ya que se tiene un conocimiento profundo del sistema, simplificando futuras actualizaciones y ajustes.

En términos de usabilidad, al ser un desarrollo a la medida, es más personalizado, es posible diseñar una interfaz amigable y optimizada para los usuarios específicos del CAR, lo cual garantiza que el sistema sea intuitivo y fácil de usar, incluso para aquellos sin experiencia técnica. Esta personalización contribuye significativamente a la experiencia del usuario y minimiza la necesidad de capacitación extensa.

Otro aspecto para considerar es el costo, que en este caso podrá ser más elevado inicialmente en comparación a otras alternativas. La inversión en un equipo de desarrollo y el tiempo requerido para implementar la solución a medida son factores relevantes, pero los beneficios a largo plazo, la capacidad de personalización y la escalabilidad hacen que esta alternativa sea una inversión estratégica que proporciona un sistema completamente ajustado a las necesidades del CAR de G&N Brands.

Ventajas:

- Personalización total del sistema según las necesidades específicas del CAR.
- Control total sobre el proceso de desarrollo y futuras actualizaciones.
- Flexibilidad para integrar nuevas funciones en el futuro (escalabilidad).
- Menor costo a largo plazo.

Desventajas:

- Requiere una inversión inicial significativa en tiempo y recursos humanos.
- Mayor tiempo de desarrollo en comparación con las alternativas de software comercial.

1.4.2 Alternativa 2: Software Comercial (COTS) - ParkFacil

Esta alternativa considera la adquisición de un software comercial ya desarrollado de gestión de estacionamientos a rasgos generales, conocida como COTS (Commercial Off-The-Self). ParkFacil [2], esta opción ofrece una solución lista para usar que puede cubrir de manera general la funcionalidad básica para la gestión de reservas de estacionamientos. Por otro lado, su capacidad para cumplir con todos los requerimientos y requisitos es limitada, ya que este tipo de software suele diseñarse para un mercado amplio, lo que implica que no siempre se adapta perfectamente a las particularidades de una organización como el CAR de G&N Brands. La funcionalidad puede ser suficiente para necesidades básicas, pero cualquier requerimiento específico probablemente no esté disponible o requiera adaptaciones costosas.

En cuanto a la usabilidad, el software comercial ParkFacil ofrece una interfaz moderna, intuitiva y fácil de operar para cualquier tipo de usuario. La plataforma cuenta con un panel de control vía web y reportes de forma rápida. La curva de aprendizaje es baja, facilitando su adopción por parte de los colaboradores del CAR.

Respecto a la funcionalidad, ParkFacil incluye módulos completos de control de acceso, monitoreo de ocupación en tiempo real, generación de tickets QR y visualización de reportes. Si bien ofrece una cobertura robusta de las funcionalidades estándar en un sistema de estacionamientos, presenta limitaciones en cuanto a personalización de

criterios de asignación o implementación de lista de espera dinámica, funcionalidades que el CAR considera necesarias.

En relación con la escalabilidad, el software permite administrar múltiples ubicaciones y usuarios, siendo adecuado para crecer en volumen de uso. Sin embargo, al tratarse de una solución comercial cerrada, su escalabilidad se encuentra sujeta a las capacidades y planes de servicio definidos por el proveedor.

Desde el punto de vista de la mantenibilidad, al ser un producto comercial, ParkFacil cuenta con soporte técnico y actualizaciones periódicas por parte del proveedor. Esto asegura estabilidad operativa y respaldo ante eventualidades técnicas, aunque limita las posibilidades de intervención directa sobre el código fuente por parte del equipo interno.

En cuanto a la seguridad, el sistema está diseñado para operar con control de acceso físico, validación de usuarios mediante credenciales, y soporte para integración con dispositivos de lectura de placas. No obstante, algunos aspectos específicos como alertas personalizadas, transferencias entre usuarios o políticas internas de cancelación podrían requerir ajustes no contemplados en la versión estándar.

Finalmente, en términos de costos, ParkFacil presenta un modelo de licenciamiento accesible en comparación con soluciones de gran escala. Su implementación no requiere un equipo de desarrollo interno, pero si puede implicar costos adicionales asociados a dispositivos físicos como sensores o cámaras, en caso de optar por un despliegue más automatizado.

Ventajas:

- Cuenta con una interfaz web y móvil, accesible desde múltiples dispositivos.
- Proporciona información en tiempo real sobre disponibilidad y uso de los estacionamientos.
- Incluye soporte técnico y actualizaciones periódicas, lo que reduce la carga de mantenimiento.
- Facilita la implementación sin necesidad de un desarrollo desde 0
- S un software desarrollado en Chile, lo cual mejora la trazabilidad, soporte y adaptación normativa local.

Desventajas:

- Presenta limitaciones en la personalización de funcionalidades específicas requeridas por el CAR, como políticas internas de cancelación o listas de espera dinámicas.
- Requiere posible inversión en infraestructura física adicional (sensores, cámaras).
- No permite una intervención directa sobre el código, lo cual restringe ajustes a futuro según las necesidades específicas de la organización.

1.4.3 Alternativa 3 Software de Código Abierto – Parking Management System

Se evaluó la implementación de un sistema de código abierto preexistente, como el proyecto 'Parking Management System' disponible en repositorios públicos [3]. Esta opción permite acceder al código que puede ser modificado, lo cual brinda cierta flexibilidad para adaptarlo a las necesidades del CAR de G&N Brands.

En cuanto a la usabilidad, muchas soluciones de código abierto de gestión de estacionamientos como Parking Management System presentan interfaces básicas y poco estandarizadas. Si bien permiten el acceso a través de navegadores web, estas soluciones requieren ajustes de diseño y traducción para ser amigables con el usuario final, lo que implica un esfuerzo adicional para su implementación práctica en un entorno corporativo como el CAR.

Respecto a la funcionalidad, este tipo de software ofrece funciones esenciales como la gestión de plazas, ingreso y salida de vehículos, registro de usuarios y reportes básicos. Sin embargo, no suelen incorporar de manera nativa funcionalidades avanzadas como integración con sistemas de control de acceso por OCR, lista de espera dinámica, notificaciones automáticas o gestión de visitas, por lo que deben ser desarrolladas o adaptadas manualmente.

En relación con la escalabilidad, las soluciones de código abierto tienen el potencial de escalar técnicamente, pero esta escalabilidad depende directamente de las capacidades del equipo de desarrollo. No existen garantías de soporte o mantenimiento evolutivo, por lo que su crecimiento en funcionalidad debe ser planificado y gestionado internamente.

Desde el punto de vista de la mantenibilidad, al contar con acceso total al código fuente, estas plataformas permiten personalización completa. Sin embargo, esto implica una alta dependencia del conocimiento técnico disponible y de la capacidad del equipo para mantener y evolucionar la aplicación sin soporte externo oficial.

En cuanto a la seguridad, estas soluciones requieren una evaluación y reforzamiento de sus mecanismos de autenticación, validación y cifrado, ya que muchas veces no están alineadas con estándares empresariales. Es responsabilidad del equipo interno implementar buenas prácticas de desarrollo seguro para evitar vulnerabilidades, lo que puede requerir tiempo y pruebas especializadas.

Finalmente, en términos de costos, las soluciones open source no implican costos por licenciamiento ni pagos por uso. Sin embargo, requieren una inversión importante en tiempo y recursos humanos para adaptarlas, probarlas e implementarlas de manera fiable. Esto incluye el esfuerzo de personal técnico capacitado para hacerlas funcionar en un entorno productivo como el del CAR.

Ventajas:

- No requiere pago por licencias ni suscripciones.
- Permite personalización completa del código fuente.
- Puede ser adaptado a requerimientos específicos con suficiente capacidad técnica.
- Representa una oportunidad de aprendizaje y control total sobre el sistema.

Desventajas:

- No incluye soporte técnico oficial ni actualizaciones garantizadas.
- Requiere experiencia técnica avanzada para implementación y mantenimiento.
- No cumple de forma inmediata con todos los requerimientos funcionales del proyecto.
- Mayor tiempo de desarrollo para alcanzar n estándar profesional de usabilidad y seguridad.

1.5 Evaluación de Alternativas

Para la selección de la alternativa más viable, se estructuró una matriz de evaluación multicriterio. La definición de los factores y sus respectivas ponderaciones se estableció cruzando los estándares de proyectos de software [4] con las restricciones estratégicas y presupuestarias del Centro de Apoyo Restaurant (CAR) G&N Brands.

Los pesos asignados a cada criterio no son arbitrarios, sino que reflejan el nivel de criticidad que cada dimensión representa para el éxito operativo e institucional del proyecto:

1.5.1 Criterios para evaluar las alternativas

- a) Funcionalidad: Evalúa la capacidad de la alternativa para cumplir con los requerimientos específicos del CAR, tales como la automatización del flujo de reservas, el manejo transparente de las listas de espera y la lógica de reasignación automática de estacionamientos sin intervención manual.
- b) Usabilidad: Evalúa la facilidad de uso y la compatibilidad multiplataforma de la interfaz gráfica. Un diseño intuitivo y responsivo minimiza la curva de aprendizaje y la resistencia al cambio operativo, garantizando que los colaboradores del CAR puedan gestionar sus reservas de manera ágil tanto desde equipos de escritorio como desde dispositivos móviles.
- c) Mantenimiento: Mide la viabilidad técnica para que el equipo de TI interno pueda dar soporte continuo a la solución, corregir incidencias y administrar la base de datos sin depender excesivamente de proveedores externos o incurrir en tiempos prolongados de inactividad.
- d) Escalabilidad: Es la capacidad del sistema de absorber aumentos de carga como usuarios concurrentes, transacciones por segundo y volumen de datos sin degradar su rendimiento, funcionalidad ni calidad de servicio, mediante el uso eficiente de recursos adicionales (por ejemplo, escalamiento horizontal de instancias y optimizaciones de datos).
- e) Costo: Representa el factor más crítico debido a las restricciones presupuestarias de la empresa. Se evalúa no solo el costo de adquisición o desarrollo inicial, sino que también los gastos recurrentes (licencias, hosting en la nube vs uso de servidores de la empresa).

1.5.2 Ponderación de Criterios y Escala de Evaluación

Los criterios de evaluación mencionados en el punto anterior tendrán la siguiente ponderación:

Tabla 1-2 Ponderación de los criterios de evaluación

CRITERIO	PONDERACIÓN
Funcionalidad	30%
Usabilidad	25%
Mantenición	15%
Escalabilidad	10%
Costo	20%
TOTAL	100%

Fuente: Elaboración propia.

Los criterios serán calificados a partir de la siguiente escala de notas:

Tabla 1-3 Escala de Calificación

CALIFICACIÓN	CONCEPTO
0	No Aplica
1	Muy Malo
2	Malo
3	Regular
4	Bueno
5	Muy Bueno

Fuente: Elaboración propia.

1.5.3 Evaluación

A partir de esto, se presenta la tabla que contiene la calificación obtenida por la Alternativa 1: Desarrollo a Medida. Para cada criterio asignado se detalla el motivo de la calificación asignada.

Tabla 1-4 Calificación de la Alternativa 1

CRITERIO	CALIFICACION	DETALLE
Funcionalidad	5	El Desarrollo a Medida (interno) satisface completamente todos los requisitos y

		requerimientos específicos, ya que se adapta a las necesidades del CAR.
Usabilidad	4	La solución puede diseñarse para ser altamente intuitiva, aunque requiere tiempo para personalizar la interfaz según las preferencias de los usuarios.
Mantenición	4	La solución permite una mantención integral, correctiva, adaptativa, perfectiva y preventiva. Sustentada en arquitectura modular en n-capas y propiedad total del código, aplicando estándares de codificación.
Escalabilidad	5	La alternativa demuestra escalabilidad al manejar aumentos de usuarios, transacciones y datos sin degradar el servicio, mediante escalamiento horizontal de servicios sin estado, balanceo de carga, caché, paginación/índices y, de ser necesario, réplicas/particionado en la base de datos
Costo	3	Aunque tiene un costo inicial más elevado que las otras alternativas, es una inversión estratégica a largo plazo debido a su flexibilidad y personalización.

Fuente: Elaboración propia.

Asimismo, se presenta la tabla que contiene la calificación obtenida por la Alternativa 2: Software Comercial (COTS). Para cada criterio asignado se detalla el motivo de la calificación asignada.

Tabla 1-5 Calificación de la Alternativa 2

CRITERIO	CALIFICACION	DETALLE
Funcionalidad	3	La solución cubre funcionalidades generales, pero no se adapta completamente a los requisitos y requerimientos específicos del CAR.
Usabilidad	5	El software comercial ParkFacil está diseñado para ser fácil de usar, con un diseño amigable y

		una fácil configuración inicial como indican en su página web https://www.parkfacil.cl
Mantenición	4	Incluye soporte del proveedor, pero la dependencia externa puede ser un inconveniente para realizar modificaciones rápidamente.
Escalabilidad	3	La alternativa escala la carga principalmente mediante configuración del proveedor (planes/tiers, autoscaling y límites de cuota), con soporte de balanceo, caché/CDN y opciones gestionadas de réplicas
Costo	4	Representa un menor costo inicial, aunque los costos de licencias y soporte pueden aumentar a largo plazo.

Fuente: Elaboración propia.

Seguidamente, se presenta la tabla que contiene la calificación obtenida por la Alternativa 3: Software de Código Abierto. Para cada criterio asignado se detalla el motivo de la calificación asignada.

Tabla 1-6 Calificación de la Alternativa 3

CRITERIO	CALIFICACION	DETALLE
Funcionalidad	3	Ofrece características básicas, pero requiere personalización significativa para cumplir con todos los requisitos y requerimientos.
Usabilidad	3	La solución presenta fortalezas en la visibilidad inmediata de resultados y en la alineación básica con el dominio; sin embargo, exhibe debilidades típicas de UI de escritorio: validaciones tardías, controles pequeños, jerarquía visual débil y ausencia de ayudas/atajos. Estas falencias no impiden su uso, pero reducen eficiencia y aumentan errores evitables en escenarios de alta concurrencia.

Mantención	3	La mantención depende de la comunidad de desarrolladores, lo que podría ser menos confiable y con soporte limitado.
Escalabilidad	4	La alternativa puede escalar la carga mediante balanceo y réplicas del servicio (si es sin estado), caché de lecturas, colas asíncronas y optimización de BD (índices, paginación, réplicas/particionado).
Costo	5	Tiene un costo inicial debido a la ausencia de licencias, aunque los costos de personalización y mantención pueden aumentar con el tiempo.

Fuente: Elaboración propia.

A continuación, se presenta la síntesis de la evaluación comparativa de alternativas Tabla 1-7. La propuesta de Desarrollo a Medida obtiene el mejor rendimiento global al maximizar los criterios de mayor peso que son la funcionalidad (30%) y usabilidad (25%) y destacar en escalabilidad y mantención, lo que se traduce en una solución capaz de alinearse con los procesos del CAR, incorporar funcionalidades específicas y crecer de forma controlada en el tiempo. Si bien su costo inicial es superior al de las otras opciones, su mayor ajuste a los requisitos y su potencial de evolución justifican la selección.

Tabla 1-7 Evaluación de Alternativas: Resumen.

CRITERIO	PONDERACION	ALTERNATIVAS CALIFICACION			ALTERNATIVAS CALIFICACIONES PONDERADAS		
		#1	#2	#3	#1	#2	#3
Funcionalidad	30%	5	3	3	1.5	0.9	0.9
Usabilidad	25%	4	5	3	1	1.25	0.75
Mantención	15%	4	4	3	0.6	0.6	0.45
Escalabilidad	10%	5	3	4	0.5	0.3	0.4
Costo	20%	3	4	5	0.6	0.8	1
Total	100%	21	19	18	4.2	3.85	3.5

Fuente: Elaboración propia.

La Tabla 1-7 resume las calificaciones ponderadas y muestra que la alternativa 1 alcanza 4,2 puntos, superando a la alternativa 2 (3,85) y la alternativa 3 (3,5). Este resultado se explica principalmente por:

- **Funcionalidad (5/5; 30% de peso):** Cubre íntegramente los requerimientos del CAR (p. ej., lista de espera dinámica, reasignación automática, integración con OCR y pasarela de pago, políticas internas de cancelación/transferencia), lo que aporta 1,5 puntos al total.
- **Usabilidad (4/5; 25%):** Permite diseñar una IU centrada en los usuarios reales del CAR, sumando 1,0 punto y reduciendo la curva de adopción.
- **Mantención (4/5; 15%) y Escalabilidad (5/5; 10%):** El control sobre el código facilita correcciones/evoluciones y soporta el crecimiento funcional sin degradar el desempeño (0,6 y 0,5 puntos, respectivamente).
- **Costo (3/5; 20%):** Aunque la inversión inicial es mayor (0,6 puntos), se compensa por el ajuste a los procesos, la reducción de reprocesos y la menor dependencia de terceros, mejorando el costo total de propiedad en el horizonte del proyecto.

En conjunto, la alternativa 1 Desarrollo a Medida “gana” porque maximiza los criterios más influyentes y asegura un encaje funcional exacto con las necesidades del CAR, habilitando mejoras futuras sin limitarse a planes o restricciones de proveedores externos.

1.6 **Alternativa seleccionada**

En esta parte del documento se describe la alternativa de solución seleccionada para resolver la problemática del CAR de G&N Brands, se menciona una descripción más detallada de la alternativa que la que se dio anteriormente, se menciona el objetivo que se pretende lograr, la planificación para su desarrollo y los beneficios que entregara al CAR una vez implementada.

1.6.1 Descripción

- ✓ **Propósito y enfoque:** La alternativa seleccionada corresponde a Desarrollo a Medida (interno) para el Centro de Apoyo Restaurant (CAR) de G&N Brands. La solución se diseña de forma específica para los procesos y políticas del CAR,

priorizando funcionalidad, usabilidad, mantención y escalabilidad. Este enfoque permite cubrir los requerimientos definidos y habilitar la evolución controlada del sistema conforme cambien las necesidades de la organización.

✓ **Alcance funcional:** La solución contempla, entre otras, las siguientes capacidades:

- Consulta de disponibilidad en tiempo real.
- Creación, modificación y cancelación de reservas.
- Lista de espera dinámica y reasignación automática de cupos liberados.
- Registro de reservas para vistas con pago en línea.
- Transferencia de reservas entre usuarios autorizados.
- Reservas preferenciales para perfiles habilitados.
- Notificaciones automáticas ante eventos relevantes.
- Reportes de uso y ocupación.
- Administración de usuarios, roles y permisos.

La plataforma contará con soporte multiplataforma y un diseño responsivo para su ejecución vía web y móvil, permitiendo que los colaboradores realicen sus gestiones desde diversos dispositivos con conexión a internet. El sistema integra una interfaz intuitiva que minimiza la curva de aprendizaje y reduce la carga operativa del encargado.

✓ **Límite del alcance:** Quedan incluidos en esta fase: gestión integral de reservas (colaboradores y visitas), reglas de negocio parametrizables, lista de espera y reasignación automática, notificaciones, reportaría, administración de usuarios y seguridad basada en roles, integración con sistema OCR para validaciones de uso de reservas. Quedan excluidos cualquier otro requerimiento o integraciones adicionales no expresamente requeridas.

✓ **Requisitos no funcionales (algunos como metas verificables):** Se establecen objetivos de calidad que orientan diseño y pruebas. Tiempo de respuestas p95 $\leq$ 3s en operaciones críticas; disponibilidad $\geq$ 96% en horario hábil; concurrencia mínima de 50 usuarios sin degradación perceptible; trazabilidad mediante registro de auditoria para operaciones relevantes; seguridad con cifrado en tránsito y en reposo; y mantenibilidad que permita correcciones menores en menos de una jornada. Estas metas serán verificadas con pruebas de rendimiento y disponibilidad, y revisión de controles de seguridad.

✓ **Principios de arquitectura:** La solución adopta arquitectura en n-capas con patrón de diseño MVC, separando presentación, servicios (API REST), negocio y datos. Se privilegian modularidad, bajo acoplamiento y alta cohesión, lo que

facilita despliegues por componente, pruebas aisladas y escalamiento progresivo. El diseño prepara contratos de integración versionados para futuras ampliaciones sin afectar estabilidad.

- ✓ **Integraciones y dependencias:** El sistema considera integración con OCR para validación de acceso por patente; pasarela de pago para reservas de visitas; servicios institucionales (autenticación/directorio y correo o mensajería para notificaciones). Las integraciones se implementarán a través de APIs con manejo de errores, reintentos y registro de eventos por auditoria.
- ✓ **Seguridad y cumplimiento:** La plataforma implementará control de acceso basado en roles, gestión de sesiones, cifrado de credenciales y datos sensibles, segregación de ambientes (desarrollo, pruebas y producción), y políticas de respaldo y recuperación. Se mantendrá auditoria de acciones críticas (creación, modificación, cancelación, reasignación y pago), resguardando confidencialidad, integridad y disponibilidad de la información.
- ✓ **Riesgos y mitigaciones:** Se reconocen riesgos de sobre carga transaccional superior a la prevista. Como mitigación se aplicará control de cambios, pruebas de estrés y monitoreo continuo con alertas y capacidad de escalar por componente. Se mitigará la dependencia de personas clave mediante estándares de codificación, documentación y transferencia de conocimiento al equipo interno de desarrollo.
- ✓ **Criterios de aceptación y métricas de éxito:** La aceptación del entregable se basará en:
 - Validación de casos de uso críticos en preproducción.
 - Cumplimiento de las metas no funcionales.
 - Correcta operación de notificaciones, lista de espera y reasignación automática.
 - Evidencia de adopción por parte de usuarios piloto.

Se monitorean indicadores de tasa de errores, tiempos de respuesta, utilización de cupos y uso efectivo de funcionalidades clave.
- ✓ **Trazabilidad con la evaluación comparativa:** La selección del Desarrollo a Medida se alinea con la evaluación de las alternativas y su mejor rendimiento en criterios de mayor peso (funcionabilidad y usabilidad), además de ventajas en mantención y escalabilidad. Aunque la inversión inicial resulta superior a otras opciones, el ajuste a los procesos del CAR y la capacidad de evolución contribuyen a optimizar el costo total en el horizonte del proyecto.

1.6.2 Beneficios

En consecuencia, se mencionan numeradamente algunos de los beneficios que tendrá el CAR de G&N Brands con la implementación de la alternativa seleccionada.

1. **Experiencia ajustada al uso real:** El sistema ofrece flujos de reserva claros y coherentes con la operación del CAR, reduciendo tiempos de búsqueda y confirmación y minimizando errores. Para el usuario final, esto se traduce en menos pasos, mayor certeza sobre la disponibilidad y menos incidencias al reservar o modificar.
2. **Acceso multiplataforma:** Al ser un sistema basado en la web, se permitirá el acceso desde cualquier dispositivo con conexión a internet (dentro del rango PC, móvil de cierta gama en adelante), facilitando las reservas incluso fuera del entorno de la oficina.
3. **Automatización del proceso de reservas:** La automatización eliminará la necesidad de gestionar correos manualmente, lo que reducirá la carga de trabajo para el personal encargado y minimizará errores humanos.
4. **Notificaciones automáticas:** El sistema enviará alertas y recordatorios sobre las reservas, lo que garantizará un mejor uso de los estacionamientos y reducirá los espacios no utilizados.
5. **Consulta en tiempo real de disponibilidad:** Se permitirá la consulta en tiempo real de la disponibilidad de estacionamientos, facilitando una gestión más eficiente y rápida de los espacios.
6. **Facilidad para cancelar y modificar reservas:** Los usuarios podrán cancelar o modificar sus reservas de forma sencilla, lo que optimizará el uso de los espacios disponibles y permitirá la reasignación de cupos.
7. **Interfaz intuitiva y de fácil uso:** El diseño amigable e intuitivo permitirá que los empleados utilicen el sistema sin necesidad de capacitación extensa, mejorado así la experiencia del usuario.
8. **Escalabilidad para futuras mejoras:** El sistema estará diseñado para ser escalable, lo que permitirá la incorporación de nuevas funcionalidades sin comprometer su funcionamiento actual.
9. **Reducción de retrasos en la asignación de estacionamientos:** La automatización del proceso reducirá los tiempos de respuesta, asegurando una asignación más ágil y eficiente de los espacios.

10. **Seguridad en el almacenamiento de datos:** El sistema protegerá todos los datos relacionados con reservas y usuarios, restringiendo el acceso según roles y permisos definidos por los administradores.
11. **Optimización del uso de recursos:** Al mejorar la gestión de los estacionamientos, se optimizará el uso de los espacios disponibles, reduciendo los estacionamientos desperdiciados por reservas no utilizadas.
12. **Mejor experiencia para los empleados:** Con un sistema ágil y automatizado, se permitirá a los empleados enfocarse en sus tareas principales sin preocuparse por la disponibilidad de estacionamientos.
13. **[Flexibilidad para adaptarse a cambios en la organización:** El sistema podrá ajustarse rápidamente a cambios en las políticas o necesidades del CAR, lo que garantizará que la solución se mantenga relevante a lo largo del tiempo.

1.7 Planificación

Se establece la planificación del proyecto bajo una metodología de desarrollo incremental orientada a la entrega de un Producto Mínimo Viable (MVP). Esta aproximación metodológica consiste en la descomposición del ciclo de vida del software en iteraciones sucesivas, donde cada incremento resulta en una versión operativa del sistema que añade valor funcional sobre la anterior. La decisión de adoptar este enfoque responde a la necesidad de mitigar riesgos técnicos de manera temprana y garantizar que el Centro de Apoyo Restaurant (CAR) disponga de las funcionalidades críticas de reserva en un plazo reducido. A diferencia de modelos tradicionales secuenciales, el desarrollo incremental permite validar la lógica de negocio y la usabilidad del sistema con usuarios reales antes de proceder con la integración de módulos de mayor complejidad [4], como el reconocimiento óptico de caracteres (OCR) y la pasarela de pagos, asegurando así una solución final alineada con los requerimientos del CAR.

1.7.1 Desarrollo del plan de personal

El desarrollo del plan de personal se estructura para dar soporte a una metodología incremental, donde el equipo técnico no opera de forma aislada en fases secuenciales, sino que participa activamente en los ciclos de definición, construcción y validación de cada incremento. Este enfoque asegura que el conocimiento técnico y el

funcional se mantengan integrados a lo largo de todo el proyecto, facilitando la adaptación del software según los resultados obtenidos en cada iteración del MVP.

Se definen los siguientes roles y responsabilidades para asegurar el cumplimiento de los objetivos del sistema:

- ✓ **Jefe de Proyecto (1 persona):** El rol asume la responsabilidad de liderar la planificación estratégica y el control de hitos. Bajo el enfoque incremental, coordina la definición del alcance en cada iteración, gestiona los recursos y asegura que cada incremento funcional cumpla con las expectativas del CAR. Actúa como el principal enlace con los interesados de G&N Brands. (Inicial del recurso: JP).
- ✓ **Analista Programador (1 persona):** Este rol se encarga de transformar los requerimientos del negocio en especificaciones técnicas detalladas para cada incremento. Participa de la fase de diseño lógico y apoya la implementación de componentes complejos, asegurando la integridad de los datos y la correcta comunicación entre las capas del sistema. (Inicial del recurso: AP).
- ✓ **QA (1 persona):** A diferencia de un modelo tradicional, el rol de QA interviene de forma continua al finalizar cada incremento. Se encarga de diseñar y ejecutar planes de pruebas funcionales, de integración y de regresión, garantizando que las nuevas funcionalidades no afecten la estabilidad de los módulos previamente implementados. (Inicial del recurso: QA).
- ✓ **Diseñador UI/UX (1 persona):** Se enfoca en la experiencia de usuario (UX) y el diseño de interfaces (UI). Su labor es crítica en las primeras etapas de cada incremento, donde debe prototipar soluciones visuales que faciliten la gestión de estacionamientos. Valida la usabilidad de los componentes desarrollados para asegurar una curva de aprendizaje mínima para el personal del CAR. (Inicial del recurso: D).
- ✓ **Desarrollador Full Stack (1 persona):** Constituye el núcleo técnico del equipo. Su responsabilidad abarca la construcción tanto del backend como del frontend en cada ciclo de desarrollo. Se encarga de la codificación de la lógica de negocio, la integración con servicios externos (OCR y pasarelas de pago) y la implementación de las interfaces de usuario definidas para el MVP. (Inicial recurso: DFS).

1.7.2 Estimación de esfuerzo

Para determinar el tiempo y los recursos necesarios para el desarrollo del sistema, se utilizará una técnica de estimación por módulos (o estimación basada en componentes funcionales). Esta aproximación consiste en desglosar el proyecto en sus principales bloques de funcionalidad, asignando a cada uno un esfuerzo estimado en horas en función de su complejidad técnica y los requerimientos identificados anteriormente.

A partir de esta división, se calcula el esfuerzo individual de cada módulo operativo, sumando posteriormente estos valores para obtener el esfuerzo total requerido por el proyecto.

Se establecen tres niveles de complejidad para estandarizar la asignación de horas por funcionalidad:

- **Complejidad Baja:** Funcionalidades con lógica estándar y mínima interacción de datos (ej. Autenticación simple, CRUDs básicos).
- **Complejidad Media:** Funcionalidades que requieren lógica de negocio específica y manipulación de múltiples entidades (ej. Mapa interactivo, Reportes dinámicos).
- **Complejidad Alta:** Funcionalidades con algoritmos complejos, integraciones con APIs de terceros o alta criticidad de datos (ej. Reasignación automática, OCR, Pasarela de pago).

Tabla 1-8 Estimación de esfuerzo por módulo y complejidad

Módulo/ Funcionalidad	Incremento	Complejidad	Esfuerzo Estimado (HH)
Configuración e infraestructura	0	Media	180
Gestión de reservas (MVP)	1	Media	410
Reasignación y Lista de Espera	2	Alta	455
Integraciones (OCR y Pagos)	3	Alta	280
Cierre y Despliegue	Final	Baja	80
TOTAL ESFUERZO DESARROLLO			1405

Fuente: Elaboración propia.

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

La planificación temporal se proyecta sobre un horizonte de 119 días, estructurados bajo un enfoque de desarrollo incremental. Esta organización permite la gestión eficiente de los tiempos mediante la entrega sucesiva de módulos funcionales, reduciendo la incertidumbre y permitiendo ajustes basados en la retroalimentación de cada iteración.

El cronograma se divide en cuatro bloques lógicos, donde se prioriza la construcción del núcleo de reservas antes de avanzar hacia las integraciones de hardware y servicios externos. Además cabe recalcar que las actividades detalladas se presentan a nivel de macro-tareas funcionales para facilitar el seguimiento del desarrollo incremental. Los requerimientos específicos como la gestión de estacionamientos preferenciales y la lógica de seguridad se encuentran absorbidos en las tareas de desarrollo de la API en el backend, mientras que el diseño de formularios de inicio de sesión se integra en el esfuerzo estimado para el frontend.

Tabla 1-9 Detalle de Planificación, Recursos y Costos

Nombre de la tarea	Duración (Días)	Trabajo (HH)	Costo (UF)	Recurso Asignado
SISTEMA PARA LA GESTIÓN AUTOMATIZADA DE ESTACIONAMIENTOS EN G&N BRANDS	119	1.405	314.40	
Hito 0: Inicio del Proyecto	Día 1			
INCREMENTO 0: REQUERIMIENTOS Y DISEÑO BASE	15 días	180	45.60	
Definición de requerimientos y especificación de requisitos de software	7 días	80	22.40	JP, AP
Diseño de arquitectura y Modelo de base de datos	8 días	100	23.20	AP, DFS
INCREMENTO 1: MVP RESERVAS	35 días	410	83.60	
Diseño UI/UX y Prototipo	7 días	65	10.40	D
Desarrollo Backend Core y API (Reservas, preferenciales y autenticación)	13 días	150	30.00	DFS

Desarrollo Frontend y Mapa (Vistas de usuario e interfaz de inicio de sesión)	10 días	125	28.00	DFS, AP
Pruebas unitarias y QA MVP	5 días	70	15,20	QA, AP
Hito 1: MVP Operativo	Día 50			
INCREMENTO 2: AUTOMATIZACION	35 días	455	104.80	
Implementación Lista de Espera	10 días	130	28.60	DFS, AP
Lógica Reasignación Automática	15 días	215	47.40	DFS, AP
Sistema de Notificaciones	5 días	60	15.20	DFS, JP
Pruebas de Integración y QA	5 días	50	13.60	QA, AP
Hito 2: Lógica de Negocio Validada	Día 85			
INCREMENTO 3: INTEGRACIONES	24 días	280	66.80	
Integración OCR (patentes)	10 días	120	26.40	DFS, AP
Integración Pasarela de Pagos	10 días	120	32.40	DFS, AP
Pruebas de Regresión y Estrés	4 días	40	8.00	QA
CIERRE Y DESPLIEGUE	10 días	80	13.60	
Marcha Blanca y Capacitación	7 días	70	10.40	JP, DFS
Ultimas revisiones y Cierre	3 días	10	3.20	AP
Hito 3: Cierre del Proyecto	Día 119			

Fuente: Elaboración propia.

1.7.4 Estimación de costos

Se exponen los costos estimados para el desarrollo a medida del sistema de reservas de estacionamientos. Se incluye el detalle correspondiente a cada recurso, considerando la capacidad máxima, tasa estándar, costo total, además, del trabajo total a lo largo del proyecto. Se utilizo el método de estimación por esfuerzo basado en la cantidad de horas asignadas a cada recurso dentro de las tareas definidas en la planificación temporal.

Tabla 1-10 Estructuras de Rentas Brutas

Rol	Renta Bruta (CLP)	Renta Bruta (UF)
Jefe de Proyecto	1.802.650	46,1
Analista Programador	1.351.987	34,6
QA	901.325	23,0
Diseñador UI / UX	901.325	23,0
Desarrollador Full Stack	1.126.656	28,8
Total	6.083.943	155,5
Valor UF en este momento: 39.120 CLP		

Fuente: Elaboración propia

Tabla 1-11 Costos de los recursos.

Nombre del recurso	Capacidad máxima	Tasa estándar	Costo	Trabajo
Jefe de Proyecto	1	UF 0.32/hora	UF 70.40	220 horas
Analista Programador	1	UF 0.24/hora	UF 91.20	380 horas
QA	1	UF 0.16/hora	UF 22.40	140 horas
Diseñador UI/UX	1	UF 0.16/hora	UF 10.40	65 horas
Desarrollador Full Stack	1	UF 0.20/hora	UF 120.00	600 horas
		TOTAL	UF 314.40	1405 horas

Fuente: Elaboración propia.

A partir de esto, se muestra una tabla final con el resumen de los costos totales del proyecto, donde el costo de imprevistos corresponde al 5% del total del proyecto.

Tabla 1-12 Costos totales del proyecto.

Recurso	% Participación	UF Participación	/20 días (UF)	Días proyecto	Costo Proyecto Empresa (UF)
JP	29.65	54.99	2.75	19	52.24
AP	22.25	41.28	2.06	58	119.70
QA	14.79	27.44	1.37	14	19.21
D	14.79	27.44	1.37	7	9.60
DFS	18.52	34.36	1.72	88	151.17
				Total	351,92 UF

Fuente: Elaboración propia.

Se presenta la distribución del esfuerzo técnico asociado exclusivamente a los requisitos funcionales y de sistema, totalizando 1.171 horas hombre (HH). Esta cifra representa el núcleo de ingeniería y desarrollo del proyecto, excluyendo las actividades transversales de gestión y planificación que se detallan en secciones posteriores. La asignación de horas para cada requisito macro se realiza mediante la aplicación de pesos relativos, los cuales consideran la complejidad algorítmica, la interacción con bases de datos y la integración con servicios externos. Esta segmentación asegura que los módulos críticos, como la reasignación automática y las reservas, reciban la carga horaria necesaria para garantizar su correcta implementación y estabilidad operativa.

Tabla 1-13 Distribución de esfuerzo entre requisitos

Requisito (macro)	Peso Relativo	Esfuerzo (HH)
Reservas (CRUD + consulta tiempo real)	0,18	210,75
Lista de espera dinámica	0,10	117,08
Reasignación automática	0,12	140,50
Visitas con pago en línea	0,10	117,08
Transferencia de reservas	0,09	105,38
Administración de usuarios/roles	0,08	93,67
Notificaciones automáticas	0,06	70,25
Reportes y ocupación	0,08	93,67
OCR / validación de uso	0,09	105,38
Auditoría y seguridad	0,10	117,08
Total	1,00	1171,00

Fuente: Elaboración propia.

Siguiendo con el análisis se calcula el esfuerzo de las actividades de no-desarrollo que se muestran en la siguiente tabla

Tabla 1-14 Esfuerzo de actividades de no-desarrollo

Concepto	% sobre desarrollo	Esfuerzo (HH)
Gestión	0,15	175,65
Inception	0,05	58,55
Total, no-desarrollo	0,20	234,20

Fuente: Elaboración propia.

Resumiendo, los esfuerzos calculados nos dan esta tabla

Tabla 1-15 Resumen consolidad de esfuerzo

Concepto	HH
Total, desarrollo (Tabla A)	1171,00
Total, no-desarrollo (Tabla B)	234,00
Total, general proyecto (HH)	1405,00

Fuente: Elaboración propia.

Se desagrega el esfuerzo de desarrollo en actividades de ingeniería según porcentajes que suman 100% del total estimado. La asignación refleja el peso relativo de cada fase en el ciclo de vida definido para el proyecto y mantiene trazabilidad con la programación temporal y el dimensionamiento previo.

Tabla 1-16 Distribución del esfuerzo en actividades de la ingeniería de software

Actividad de Ingeniería	% sobre desarrollo	Esfuerzo (HH)	% del total proyecto (incl. no-dev)
Análisis	0,15	175,63	12,5
Diseño	0,15	175,63	12,5
Construcción	0,50	585,50	41,7
Pruebas/QA	0,15	175,63	12,5
Puesta en marcha / UAT	0,05	58,55	4,2
Total (desarrollo)	1,00	7.444,00	83,4

Fuente: Elaboración propia.

Se detalla la matriz de horas por actividad y por rol, garantizando que las sumas por fila y columna coincidan con el esfuerzo total de desarrollo. La proporción por rol en cada actividad responde a responsabilidades típicas, conservando consistencia con el plan de personal.

Tabla 1-17 Distribución de esfuerzo por rol y actividad de ingeniería de software (HH)

Actividad	JP	DFS	AP	D	QA	Total, fila (HH)
Análisis	35,1	52,7	52,7	17,6	17,6	175,7
Diseño	17,6	52,7	52,7	35,1	17,6	175,7
Construcción	29,3	351,3	204,9	0,0	0,0	585,5
Pruebas/QA	8,8	17,6	17,6	17,6	114,1	175,7
Puesta en marcha / UAT	11,7	11,7	11,7	0,0	23,4	58,5
Totales por rol	102,5	486,0	339,6	70,3	172,7	1171,1

Fuente: Elaboración propia.

1.7.5 Administración del riesgo.

Para anticipar situaciones que puedan afectar el desarrollo normal del proyecto, resulta fundamental llevar a cabo una planificación adecuada que contemple las contingencias y eventualidades, ya sean de origen interno o externo, con el propósito de mitigar sus posibles efectos. En este sentido, es necesario diseñar una estrategia efectiva para gestionar los riesgos identificados, definiendo para cada uno de ellos las señales de alerta que permitan activar el plan de mitigación correspondiente, con el objetivo de prevenir o minimizar su impacto.

La metodología de evaluación y las escalas utilizadas en la matriz de riesgos se fundamentan en los estándares de gestión de proyectos de software [4]. Para asegurar una evaluación objetiva, a continuación se define el alcance de cada dimensión analizada en la tabla:

- **Posibilidad:** Determina la posibilidad de que el evento de riesgo se materialice durante el ciclo de vida del proyecto. Se evalúa desde 1 (ocurrencia altamente improbable) hasta 10 (ocurrencia inminente o casi segura).
- **Impacto:** Mide el grado de afectación sobre los objetivos críticos del proyecto (tiempo, costo, calidad o alcance) si el evento llega a ocurrir. Se evalúa desde 1 (afectación mínima que no altera la ruta crítica) hasta 10 (impacto crítico que compromete la viabilidad del desarrollo del sistema).
- **Costo de retiro:** Representa el nivel de esfuerzo técnico, horas hombre o recursos necesarios para aplicar el plan de contingencia y revertir los daños de una vez materializado el riesgo. Se clasifica desde 1 (esfuerzo de corrección mínimo y bajo costo) hasta 10 (esfuerzo técnico extremo y alto costo de mitigación).
- **Prioridad:** Es el indicador que define el nivel de urgencia para gestionar el riesgo. A diferencia de las métricas anteriores, su valoración funciona de manera inversamente proporcional: **un valor numérico menor indica una mayor criticidad (siendo 1 la máxima prioridad de mitigación y asignación de recursos)**, mientras que los valores más altos representan riesgos de menor impacto. La prioridad se calcula de la siguiente forma:
Prioridad = (11-Posibilidad) * (11-Impacto) * Costo

Tabla 1-18 Administración de Riesgos

N	Categoría	Riesgo	Posibilidad (1-10)	Impacto (1-10)	Costo de Retiro (1-10)	Prioridad
1	Personas	Un miembro clave del equipo decide abandonar el proyecto	5	10	6	36
2	Producto de Software	Problemas de compatibilidad con dispositivos móviles o navegadores específicos	7	9	5	40
3	Tecnología	Interrupción del servicio por fallos en la infraestructura tecnológica	6	10	8	40
4	Requerimientos	Los requerimientos obtenidos son ambiguos o incompletos	7	8	4	48
5	Estimaciones	Se producen retrasos en la entrega de entregables	6	9	5	50
6	Producto de Software	Errores en la lógica de asignación automática o prioridades	7	8	5	60
7	Personas	Existe mala comunicación entre el equipo y el cliente	5	7	3	72
8	Organizacional	Se solicitan cambios tardíos	7	7	5	80
9	Producto de Software	Ocurren problemas técnicos con herramientas o software	6	7	5	100
10	Personas	Resistencia al cambio por parte de los empleados	6	7	5	100

11	Producto de Software	Sobrecarga del sistema por alta demanda simultánea	6	8	7	105
12	Personas	Se presentan enfermedades del personal clave	5	5	3	108
13	Proyecto	Existe pérdida de documentación clave	4	9	8	112
14	Personas	Falta de experiencia técnica en alguna etapa del proyecto	5	6	4	120
15	Personas	Hay baja disponibilidad de algún recurso del equipo	5	6	4	120

Fuente: Elaboración propia

A continuación, se detalla el plan de acción establecido para la gestión de cada riesgo identificado. Para asegurar un abordaje integral, se definen estrategias de respuesta que contemplan tanto acciones preventivas (orientadas a evitar la materialización del evento adverso o disminuir su probabilidad de ocurrencia) como acciones correctivas y de contingencia (diseñadas para controlar, reparar y asumir el impacto técnico o económico una vez que el riesgo se ha materializado).

Tabla 1-19 Plan de estrategias preventivas y correctivas de riesgos

Riesgo	Prioridad	Señal de alerta	Plan de Mitigación Preventiva	Plan de Mitigación Correctiva	Responsable
Un miembro clave del equipo decide abandonar el proyecto	36	Aumento en ausencia y/o disminución en entregables	Se documentan los procesos críticos y se	Reemplazar al recurso clave de manera	JP

			mantienen reuniones de feedback con el equipo	inmediata y se asigna un periodo de transición	
Problemas de compatibilidad con dispositivos móviles o navegadores específicos	40	Reportes recurrentes de incompatibilidad en pruebas internas.	Realizar pruebas de compatibilidad durante el desarrollo con diversos dispositivos y navegadores	Implementar actualizaciones rápidas y enfocadas en dispositivos específicos	AP, DFS
Interrupción del servicio por fallos en la infraestructura tecnológica	40	Reportes frecuentes de intermitencia o caída del sistema	Contratar servicios confiables y monitorear infraestructura permanentemente	Activar rápidamente un respaldo en servidores secundarios o nube	DFS, JP
Los requerimientos obtenidos son ambiguos o incompletos	48	Se identifican respuestas poco claras del cliente en reuniones iniciales	Se realizan reuniones periódicas de validación y se documentan los requerimientos claramente	Revisar y ajustar los requerimientos faltantes de forma prioritaria	JP, AP
Se producen retrasos en la entrega de entregables	50	Las tareas muestran	Se realiza control de tiempos con	Asignar más recursos	JP

		avance lento en las revisiones semanales	herramientas de gestión y se asignan metas parciales	temporalmente o se ajustan plazos si es necesario	
Errores en la lógica de asignación automática o prioridades.	60	Reportes frecuentes de conflictos o errores en asignaciones	Realizar pruebas lógicas exhaustivas y validaciones durante el desarrollo	Ajustar inmediatamente algoritmos y comunicar claramente correcciones a usuarios afectados.	AP, QA
Existe mala comunicación entre el equipo y el cliente	72	Se detectan retrasos o falta de respuesta a correos importantes	Se implementan reuniones de seguimiento con una agenda definida y se utilizan herramientas colaborativas	Definir canales de comunicación urgentes y se prioriza el feedback	JP
Se solicitan cambios tardíos	80	El cliente menciona posibles ajustes en fases avanzadas del desarrollo	Se definen políticas claras sobre la aceptación de cambios y su	Evaluar el impacto de los cambios y se implementan solo los críticos	JP

			impacto en los plazos		
Ocurren problemas técnicos con herramientas o software	100	Se reportan fallas ocasionales en las herramientas de desarrollo	Se validan las herramientas antes de implementarlas y se cuenta con planes alternativos	Solicitar soporte técnico especializado para resolver las fallas	DFS
Resistencia al cambio por parte de los empleados	100	Comentarios negativos frecuentes o bajo uso inicial del sistema	Capacitación efectiva previa y difusión clara de los beneficios del nuevo sistema	Reforzar capacitaciones adicionales y ofrecer soporte continuo	JP
Sobrecarga del sistema por alta demanda simultánea	105	Tiempos de respuesta elevados en pruebas de estrés	Optimizar rendimiento desde la etapa inicial de desarrollo	Implementar escalabilidad inmediata en caso de saturación (cloud)	DFS, JP
Se presentan enfermedades del personal clave	108	Se detecta una alta frecuencia de licencias médicas cortas	Se documentan todas las tareas del personal clave y se designa un recurso de respaldo	Redistribuir las tareas entre los miembros activos	JP

Existe pérdida de documentación clave	112	Se observa desorganización en el almacenamiento de archivos	Se realizan copias de seguridad periódicas y se almacenan en plataformas seguras	Restaurar la última copia de seguridad disponible	JP, AP
Falta de experiencia técnica en alguna etapa del proyecto	120	Se reportan dificultades recurrentes en la ejecución de tareas	Se asignan tareas según la experiencia y se planifican capacitaciones técnicas	Solicitar apoyo externo para resolver bloqueos críticos	JP, DFS
Hay baja disponibilidad de algún recurso del equipo	120	Los recursos muestran dificultades para cumplir con las asignaciones	Se planifica la disponibilidad del equipo de forma anticipada y se monitorea el rendimiento	Redistribuir las tareas y se priorizan las más críticas	JP

Fuente: Elaboración propia

CAPITULO 2. ASPECTOS RELEVANTES DEL ANÁLISIS

2 ASPECTOS RELEVANTES DEL ANÁLISIS

Este capítulo aborda el análisis funcional y técnico del sistema propuesto, identificando los actores involucrados, los casos de uso, y la estructura lógica que permitirá satisfacer los requerimientos definidos previamente. Se definen los elementos esenciales que servirán como base para el diseño e implementación del sistema.

Para el modelamiento formal de los requerimientos del sistema, se utilizó el Lenguaje de Modelado Unificado (UML), estándar en la industria para el diseño orientado a objetos [4], [5]

2.1 Descripción de la solución propuesta

La solución propuesta consiste en el desarrollo e implementación de un sistema informático que permita automatizar la gestión de reservas de estacionamientos del Centro Apoyo Restaurant (CAR) de G&N Brands. Esta plataforma estará compuesta por una aplicación web y móvil, orientada a los colaboradores que requieran reservar un espacio de estacionamiento, y un módulo de administración destinado al personal encargado de la gestión.

Visión operativa: La solución habilita la gestión integral de reservas de estacionamiento mediante una interfaz web responsiva y una app móvil, con procesos guiados que reducen errores y tiempos de atención. Se prioriza la ocupación efectiva de cupos, la trazabilidad de eventos y la capacidad de reasignación automática, manteniendo la experiencia de usuarios consistente con los criterios de usabilidad definidos en el punto 1.6.1

Escenarios de uso prioritarios (flujos de alto nivel):

1. Reserva estándar de colaborador: El usuario ingresa fecha, consulta disponibilidad para esa fecha, selecciona estacionamiento y confirma. El sistema valida reglas (sanciones o penalizaciones activas, disponibilidad nuevamente) y registra la reserva. Se emite notificación de confirmación.
2. Lista de espera y reasignación automática: Cuando no existen estacionamientos disponibles, la solución ofrece la incorporación a lista de espera bajo política definida (prioridad por ingreso, rol, etc.). Si se libera un estacionamiento, se detecta el evento y se reasigna automáticamente al primer usuario elegible en la lista de espera. Se emite notificación de confirmación.

3. Reserva de visita con pago en línea: El colaborador registra datos de la visita, verifica disponibilidad y se procesa el pago en línea cuando corresponda. La reserva queda sujeta a las mismas reglas de todos los usuarios, con notificaciones específicas al colaborador que registre la visita.
4. Transferencia de reserva: Un colaborador con reserva vigente puede transferirla a otro usuario autorizado, conservando la trazabilidad del traspaso (origen, destino, hora y motivo).
5. Gestión administrativa: El encargado del CAR parametriza reglas (prioridades, penalizaciones), consulta reportaría y gestiona usuarios y permisos, con bitácora de cambios.

Modelo de roles y permisos (resumido):

- Colaborador: Crear/modificar/cancelar reservas propias, incorporarse a la lista de espera, transferir reserva, gestionar reservas de visita.
- Encargado CAR/Administrador: parametrizar reglas, administrar reservas, revisar auditoria, generar reportes, gestionar usuarios/roles.
- Sistemas externos (OCR/pasarela de pago): operar vía APIs bajo contratos versionados, no acceden a funciones de administración.

Entidades y datos clave: La operación se apoya en entidades principales como Usuario, Reserva, Notificación, ListaEspera y EventoAuditoria. Cada transacción relevante persiste su estado, marcas temporales y actor, garantizando la trazabilidad. La identificación vehicular puede soportarse con OCR de patente cuando el flujo lo requiere.

2.2 Diagrama de actividades general UML (Flujo Administrativo)

A continuación, se representa mediante un diagrama de actividades UML el flujo de trabajo actual correspondiente a la gestión manual de reservas de estacionamientos en el Centro Apoyo Restaurant (CAR). Este proceso no cuenta con automatización, y se realiza principalmente mediante correos electrónicos y validaciones manuales por parte del encargado. A continuación, se detallan los actores involucrados:

- **Colaborador:** Corresponde al usuario del CAR que necesita reservar un espacio de estacionamiento. Envía el correo al administrador de las reservas de estacionamientos para poder hacer una reserva.

- **Administrador del sistema (Encargado de reservas):** Es quien gestiona manualmente las solicitudes de los colaboradores. Recibe las solicitudes por correo de los colaboradores, consulta la disponibilidad en una planilla Excel que el mismo va llenando, responde las solicitudes vía correo, ya sea confirmando o rechazando y también valida manualmente (esperando el correo del colaborador) quien accede el día de la reserva.
- **Planilla Excel:** Representa el único “sistema” involucrado actualmente, el cual es una planilla Excel sin integración. Contiene el registro de disponibilidad de los estacionamientos, además del registro de la reserva y no permite interacción directa con el colaborador y depende completamente del administrador para ser consultada.

También detallare las actividades que se realizan dentro del proceso de reserva:

- **Envía correo solicitando reserva:** Inicia el proceso de forma manual, redactando y enviando un correo al Administrador del sistema con su solicitud, cabe recalcar que no tiene visibilidad de la disponibilidad en tiempo real, hace la solicitud “a ciegas”.
- **Revisa disponibilidad en planilla Excel:** El encargado abre el archivo Excel para verificar si hay espacios disponibles en la fecha solicitada.
- **Planilla contiene disponibilidad:** Representa el uso de la planilla como base de consulta. No está conectada a nada y tampoco se actualiza automáticamente.
- **¿Hay disponibilidad?:** Punto de decisión en el flujo donde el encargado verifica en la planilla Excel si existe un espacio libre para la fecha solicitada.
SI: Se procede a la confirmación.
NO: Se deriva al flujo de rechazo para notificar al usuario.
- **Confirma reserva por correo:** Si hay cupo, el administrador responde manualmente al colaborador informando que la reserva fue aceptada, indicando el estacionamiento.

- **¿Usa el estacionamiento?:** Punto de decisión en el flujo, si no se usa la reserva, entonces el espacio se “pierde” ya que no se reasigna, en caso de usarlo, entonces se envía un correo para validar el uso.
- **No se asigna ni se deja en lista de espera:** Si no hay cupo, simplemente se ignora o se rechaza la solicitud. No existe seguimiento ni fila de espera.
- **No se genera ningún tipo de reporte ni historial:** No existe un sistema que registre el uso real de los espacios ni estadística de demanda u ocupación.
- **Rechaza reserva por correo:** Actividad que realiza el encargado al confirmar que no hay cupos. Redacta y envía un correo electrónico de respuesta al colaborador, indicando explícitamente que no hay disponibilidad para la fecha solicitada.
- **Recibe notificación de rechazo por correo:** El colaborador recibe el correo enviado por el encargado con la respuesta negativa. Con esta acción, el usuario se da por enterado de la falta de cupo y el proceso finaliza.
- **Día de reserva: acceso manual por correo:** El acceso al estacionamiento se valida manualmente. No hay sistema de verificación física o automatizada.
- **Reserva no utilizada se pierde:** Si el colaborador no utiliza su reserva, esta se pierde, no se libera automáticamente ni se reasigna.

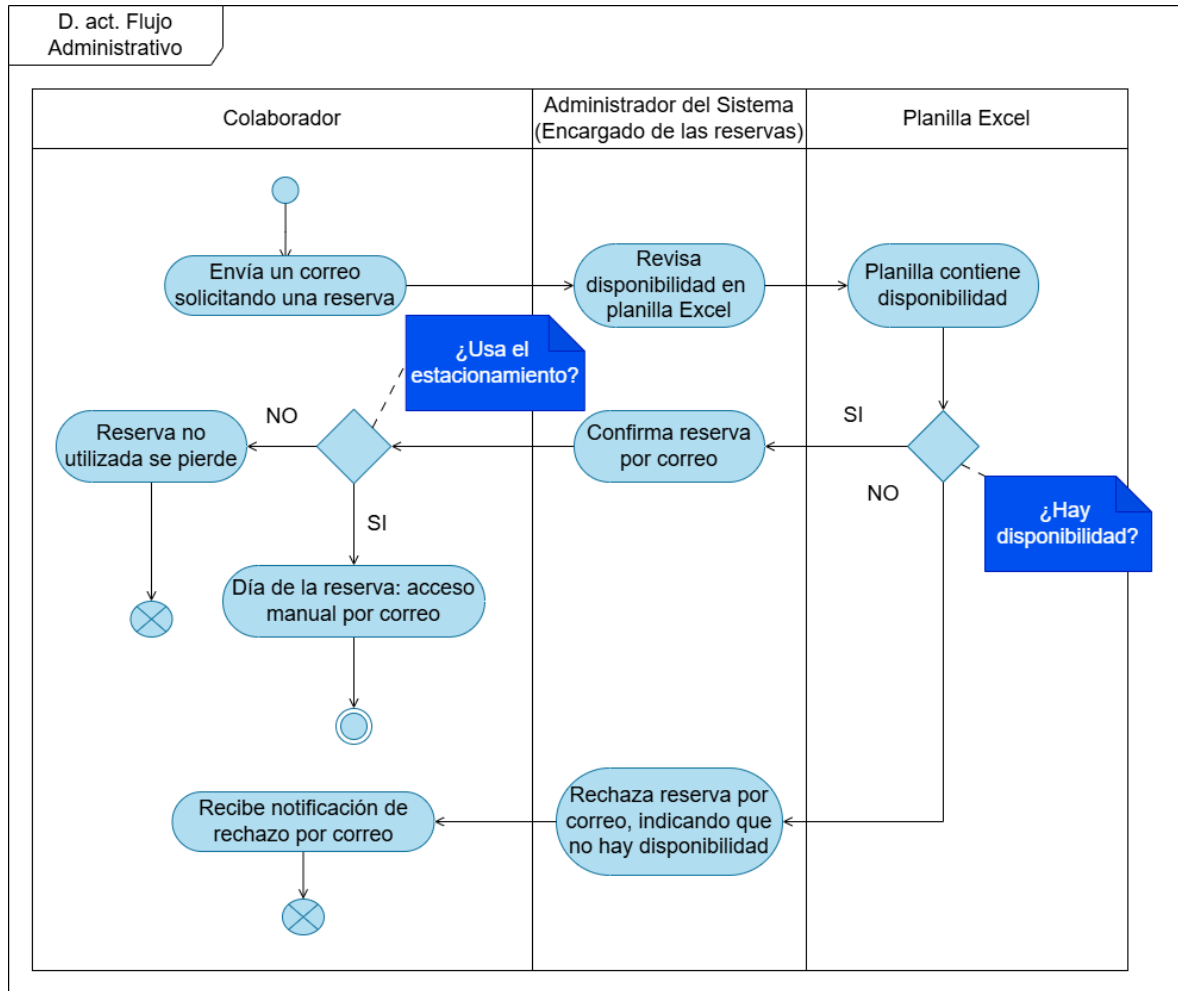


Figura 2-1 Diagrama de actividades flujo administrativo (actual)

Fuente: Elaboración propia.

2.3 Requerimientos del Sistema

En esta sección se presentan los requerimientos del sistema necesarios para dar respuesta a las necesidades identificadas durante el análisis. Los requerimientos permiten definir de forma estructurada que debe hacer el sistema (requerimientos funcionales, RF-nn) y en qué condiciones debe operar (requerimientos no funcionales, RNF-nn). Esta especificación sirve como base para el diseño, desarrollo y validación de la solución propuesta, asegurando que el sistema cumpla con las expectativas del usuario final.

✓ Requerimientos funcionales (RF-nn):

Tabla 2-1 Requerimientos Funcionales

ID	Nombre	Descripción
RF-01	Crear nuevo usuario	El sistema debe permitir a un nuevo colaborador o al administrador crear una cuenta de usuario desde la interfaz web o móvil, requiriendo como credenciales el ingreso de un correo corporativo válido y el registro de una contraseña de seguridad en formato alfanumérico (con un mínimo de 8 caracteres)
RF-02	Listar usuarios	El sistema debe permitir al administrador visualizar una lista completa de usuarios activos e inactivos desde la interfaz web o móvil. La información desplegada por cada registro debe incluir obligatoriamente: RUT, nombre completo, correo corporativo, patente del vehículo asociado, rol en el sistema y estado actual de la cuenta
RF-03	Modificar usuario	El sistema debe permitir al administrador actualizar los datos de un usuario existente desde la interfaz web o móvil, restringiendo la modificación exclusivamente a los siguientes campos: nombres, apellidos, patente del vehículo, rol del sistema y cambio de estado (activo/inactivo). (Nota técnica: El RUT o el correo rara vez se modifican por ser llaves únicas, por eso no se incluyen aquí).
RF-04	Eliminar usuario	El sistema debe permitir al administrador deshabilitar a un usuario (sin eliminar sus datos del sistema) desde la interfaz web o móvil, impidiéndole el acceso a la plataforma.
RF-05	Crear reserva	El sistema debe permitir a un colaborador crear una reserva indicando fecha y horario específicos, desde la interfaz web o móvil, siempre que haya disponibilidad.

RF-06	Listar reservas	El sistema debe listar las reservas previamente creadas, permitiendo a colaboradores y administradores visualizar desde la interfaz web o móvil la siguiente información de trazabilidad: identificador de la reserva, fecha, bloque horario, número de estacionamiento asignado, patente del vehículo registrado y el estado actual de la solicitud (Ejemplo: Confirmada, Cancelada, En espera).
RF-07	Modificar reserva	El sistema debe permitir a un colaborador modificar una reserva confirmada desde la interfaz web o móvil, condicionado a que dicha acción se ejecute hasta 1 hora después de su creación, luego de esa hora no podrá ser modificada, solo eliminada.
RF-08	Eliminar (cancelar) reserva	El sistema debe permitir a un colaborador o administrador eliminar una reserva, entendiéndose eliminar con cancelar esta misma desde la interfaz web o móvil.
RF-09	Reasignación automática	El sistema debe permitir reasignar automáticamente un estacionamiento liberado por cancelación a otro usuario en lista de espera dinámica. La asignación se ejecutará evaluando dos criterios de prioridad estrictos: en primer lugar, el perfil del usuario (asignando cupos específicos si requiere estacionamiento preferencial) y en segundo lugar, el orden cronológico de ingreso a la lista de espera (sistema FIFO - First In, First Out).
RF-10	Lista de espera dinámica	El sistema debe mantener una lista de espera dinámica, priorizando solicitudes pendientes por orden cronológico y prioridad organizacional.
RF-11	Registrar reserva externa	El sistema debe permitir al colaborador o administrador registrar reservas externas para visitantes, indicando obligatoriamente nombre del

		visitante, Rut y patente del vehículo desde la interfaz web o móvil.
RF-12	Pago online de visitas	El sistema debe permitir que las visitas realicen el pago online de su reserva mediante una pasarela de pago desde la interfaz web o móvil.
RF-13	Integración con OCR	El sistema debe integrarse con un lector OCR que valide el ingreso del vehículo mediante lectura de placa, confirmando reserva activa en la base de datos.
RF-14	Notificaciones automáticas	El sistema debe permitir enviar notificaciones automáticas por correo o app móvil en los siguientes eventos: confirmación de reserva, cancelación, recordatorio previo al uso
RF-15	Transferencia de reserva	El colaborador podrá transferir su reserva a otro usuario autorizado, manteniendo la trazabilidad en el historial.
RF-16	Generación de reportes	El sistema debe permitir al administrador generar reportes mensuales de uso, cancelación y tasas de ocupación.
RF-17	Reserva preferencial	El sistema debe permitir reservar ciertos estacionamientos exclusivamente para usuarios que tengan el atributo "preferencial".
RF-18	Inicio de sesión	El sistema debe permitir al Colaborador y al Administrador autenticarse en la plataforma ingresando sus credenciales de acceso, las cuales corresponden estrictamente al correo corporativo registrado y a la contraseña alfanumérica secreta.
RF-19	Crear penalización	El sistema debe permitir al Administrador crear una penalización indicando días, fecha de inicio, fecha de termino, razón y estado desde la interfaz web o móvil.
RF-20	Listar penalización	El sistema debe permitir al Administrador visualizar una lista completa de penalizaciones previamente creadas desde la interfaz web o móvil.

RF-21	Modificar penalización	El sistema debe permitir al Administrador modificar los detalles de una penalización previamente creada, como días, fechas o motivo, desde la interfaz web o móvil.
RF-22	Eliminar penalización	El sistema debe permitir al Administrador eliminar (deshabilitar) una penalización desde la interfaz web o móvil, impidiendo su aplicación futura sin borrarla del historial.

Fuente: Elaboración propia

✓ Requerimientos no funcionales (RNF-nn):

Tabla 2-2 Requerimientos No Funcionales

ID	Nombre	Descripción	Método de verificación
RNF-01	Multiplataforma	El sistema debe estar disponible en versión web y móvil, garantizando compatibilidad mínima con: Android 13 (API 33) o superior. iOS 16 o superior. Navegadores de escritorio y móviles: Chrome 120+, Edge 120+, Firefox 115 ESR+, Safari 16+. La interfaz debe ser responsiva y mantener equivalencia funcional entre web y móvil para los flujos críticos de reserva, cancelación y reasignación.	Pruebas de acceso y operación en matriz de dispositivos/navegadores mínimos (físicos o emulados), evidenciadas mediante registro de casos de prueba, capturas y resultados. Se incluirá al menos 1 dispositivo Android 13+, 1 iPhone con iOS 16+ y ejecución en los navegadores mínimos indicados.

RNF-02	Tiempo de respuesta	El sistema debe responder solicitudes críticas (como reserva y cancelación) en un máximo de 3 segundos.	Pruebas de carga con herramientas Postman o JMeter para medir tiempos de respuesta.
RNF-03	Seguridad	Las contraseñas se procesan exclusivamente en la capa de aplicación (Backend en ASP.NET Core) y se almacenan con hash Argon2id (salt único ≥ 16 bytes, pepper en gestor de secretos; parámetros mínimos: memoria 64–128 MiB, iteraciones ≥ 3 , paralelismo 1–2). Las credenciales de servicio se protegen con AES-256-GCM mediante envelope encryption con KMS y rotación periódica. Todo tránsito se realiza sobre TLS 1.2/1.3.	Revisión de configuración (parámetros Argon2id en código). Inspección de almacenamiento (hash + salt; ausencia de contraseñas en texto claro). Prueba de rotación y envelope encryption en KMS (logs de clave/cambio). Pen test básico: intento de downgrade de TLS y verificación de cabeceras seguras.
RNF-04	Disponibilidad	El sistema debe estar disponible al menos el 96% del tiempo durante el horario hábil (08:00 – 20:00 horas.)	Monitoreo del sistema mediante herramientas como UptimeRobot o logs del servidor.
RNF-05	Escalabilidad	El sistema debe tener la capacidad de escalar horizontalmente (agregar más instancias) o verticalmente ante un	Prueba de Estrés: Se utilizará JMeter para inyectar una carga progresiva que supere la

		aumento de la demanda. Específicamente, debe ser capaz de soportar un aumento de carga del 200% sobre la concurrencia base (pasando de 50 a 150 usuarios) manteniendo los tiempos de respuesta por debajo de 4 segundos.	línea base. Se verificará que la infraestructura active reglas de auto escalado (cuando CPU > 70%) y que el sistema no colapse, manteniendo la operatividad.
RNF-06	Mantenibilidad	El sistema debe estar desarrollado con buenas prácticas que permitan realizar cambios o correcciones en menos de 8 horas por requerimiento.	Inspección del código, documentación técnica y tiempos de atención de incidencias.
RNF-07	Usabilidad	El sistema debe poseer una interfaz gráfica intuitiva y responsiva que permita a los colaboradores del CAR ejecutar los flujos críticos de operación (consultar disponibilidad, reservar, cancelar y reasignar estacionamientos) de forma autónoma y sin requerir capacitación técnica especializada. El diseño de la plataforma debe minimizar la curva de aprendizaje y la carga cognitiva, asegurando que las tareas	Para dar por cumplido este requerimiento, el sistema debe garantizar una tasa de finalización de tareas sin asistencia igual o superior al 90%. Esta métrica será validada en un entorno de preproducción mediante la aplicación de la escala estandarizada SUS (System Usability Scale) a una muestra representativa de usuarios del CAR, exigiendo la obtención de un puntaje igual o superior a 80,0 (calidad "excelente").

		principales se completen de manera ágil, eficiente y con un mínimo de errores de interacción.	
RNF-08	Compatibilidad	El sistema debe funcionar correctamente en los navegadores Chrome 120+, Edge 120+, Firefox 115 ESR+ y Safari 16+, así como en Android 13+ e iOS 16+ para uso móvil. No se asegura compatibilidad con versiones anteriores a las mínimas.	Ejecución de casos de prueba en una matriz de dispositivos/navegadores mínimos (físicos o emulados), cubriendo autenticación, consulta de disponibilidad, reservar/modificar/cancelar, notificaciones y accesibilidad básica. Se aprueba si todas las funciones críticas pasan en cada combinación mínima definida.
RNF-09	Fiabilidad	El sistema debe garantizar alta fiabilidad y tolerancia a fallos transitorios (tales como micro cortes de red o tiempos de espera agotados en la base de datos). Para lograr que el sistema se recupere automáticamente ante fallos menores sin pérdida de información, se implementarán políticas de reintento (retry policies con retardo exponencial) en la Capa de Servicios. De este	Se validará mediante la simulación de fallos controlados (cortes de conexión temporales y respuestas HTTP 500 simuladas). Se inspeccionarán las trazas de logs del sistema para verificar que la Capa de Servicios captura la excepción y ejecuta correctamente la política de reintento configurada, logrando la recuperación automática de la transacción sin intervención manual ni pérdida de datos.

		modo, ante una falla temporal, el backend reintentará la transacción un número de 3 veces antes de abortar la operación y devolver un error definitivo	
RNF-10	Trazabilidad de acciones	Toda acción crítica (crear, cancelar, modificar) debe quedar registrada en un log con fecha, hora y usuario.	Inspección de logs de auditoría durante pruebas funcionales.
RNF-11	Capacidad de concurrencia	El sistema debe permitir al menos 50 usuarios operando simultáneamente sin degradar el rendimiento.	Pruebas de concurrencia con herramientas como Apache JMeter.
RNF-12	Actualización sin interrupción	Las actualizaciones del sistema deben poder realizarse sin necesidad de suspender el servicio por más de 5 minutos.	Pruebas de despliegue continuo y monitoreo post-actualización

Fuente: Elaboración propia.

2.4 Condicionantes del Diseño Tecnológico

El éxito del desarrollo del sistema propuesto depende del cumplimiento de ciertos condicionantes tecnológicos que aseguran su funcionalidad, estabilidad y sostenibilidad. A continuación, se detallan los principales factores que condicionan el diseño tecnológico del proyecto.

- **Lenguajes y frameworks adecuados:** Se requiere experiencia en lenguajes como JavaScript, Python o C#, junto con frameworks modernos como React, Angular o ASP.NET, que faciliten el desarrollo multiplataforma del sistema (web y móvil).
- **Base de datos relacional robusta:** Es necesario implementar un motor de base de datos como PostgreSQL o SQL Server, que permita manejar múltiples usuarios y transacciones simultáneas con integridad y consistencia de los datos.
- **Integración con sistemas externos (OCR y pagos):** El sistema debe poder integrarse con herramientas de reconocimiento óptico de caracteres (OCR), así como plataformas de pasarelas de pago online como WebPay u otras compatibles.
- **Aplicaciones y herramientas gratuitas o de código abierto:** Siempre que sea posible, se utilizarán herramientas open source o gratuitas, como Apache JMeter para pruebas de carga, con el fin de reducir costos sin afectar la calidad del desarrollo.
- **Seguridad y control de acceso:** Se deben aplicar prácticas de seguridad como cifrado de contraseñas, conexiones bajo protocolo HTTPS, y roles de usuario bien definidos para proteger la información.
- **Disponibilidad de conectividad estable:** Para que el sistema funcione correctamente, es indispensable contar con infraestructura de red estable en el entorno del CAR que asegure la disponibilidad continua del servicio.
- **Tiempos de respuesta aceptables:** Se espera que el sistema responda a acciones críticas en menos de 3 segundos, lo que condiciona el diseño a ser eficiente a nivel de consultas y uso de recursos.

- **Patrón de diseño MVC (Modelo-Vista-Controlador):** La solución adoptará el patrón de diseño MVC, que separa la lógica de negocio, la interfaz del usuario y el control de la aplicación, lo que permite una gestión más eficiente del código, mejora la escalabilidad, facilita el mantenimiento y la implementación de nuevas funcionalidades.

2.5 Estructura funcional del sistema (Caso de Uso General)

En esta sección se describen los casos de uso a nivel general. Además, se describirán los actores identificados que interactúan con el sistema. La sintaxis de los casos de uso será CU – nn, donde nn representa un numero correlativo.

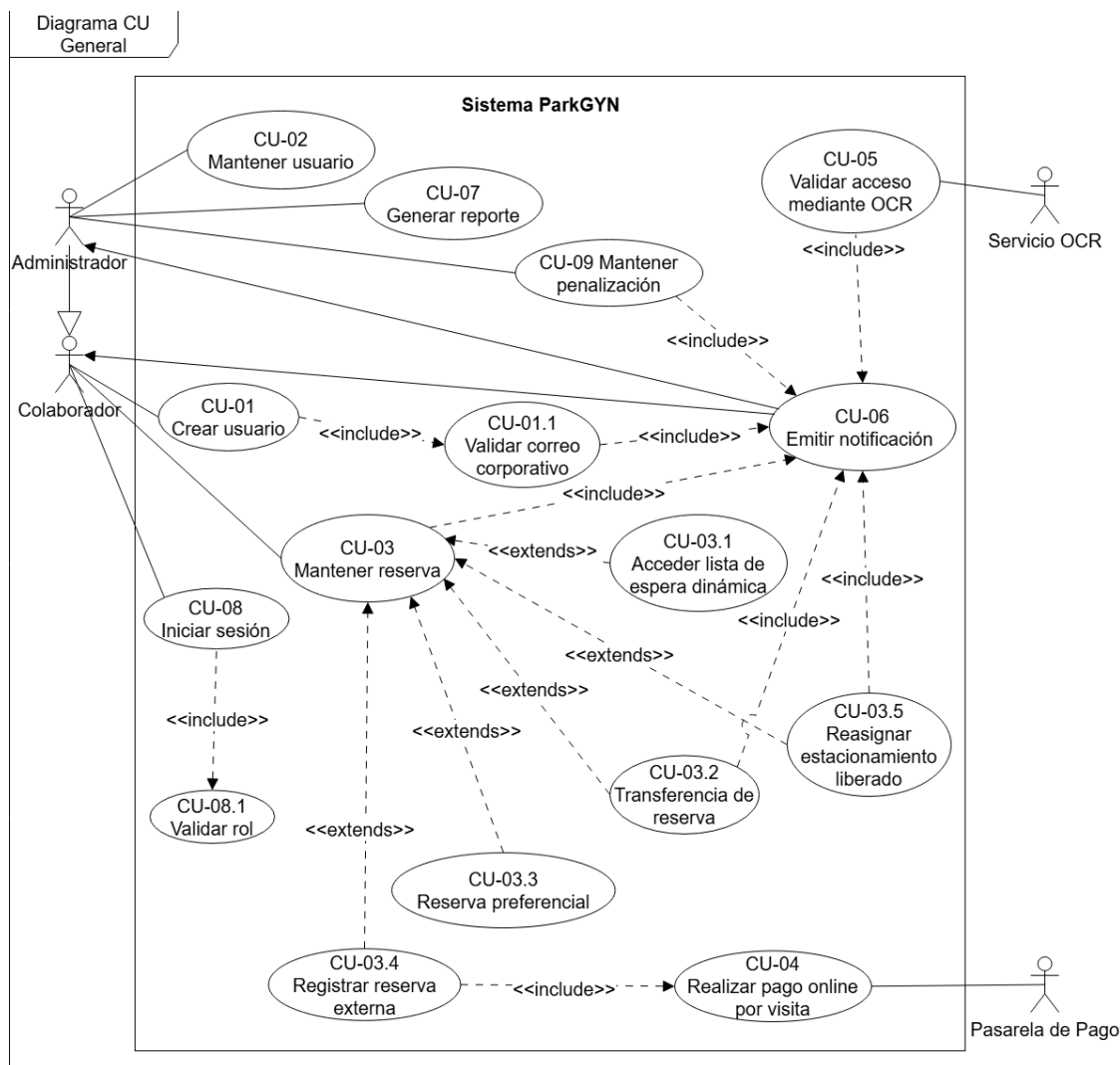


Figura 2-2 Diagrama de caso de uso general

Fuente: Elaboración propia

La identificación de los perfiles de usuario y las fronteras del sistema se realizó mediante un diagrama de casos de uso y actores [6].

2.5.1 Actores involucrados

Se listan todos los actores involucrados que interactúan o se ven afectados por el Sistema propuesto.

- **Colaborador:** Usa el sistema, primero para crear su cuenta de usuario, luego para poder hacer reservas y todo lo que implica las reservas.
- **Administrador:** Es el administrador del sistema, hereda todos los casos de uso del Colaborador y más funciones.
- **Servicio OCR:** Actor sistémico externo que interactúa con el sistema principal. Representa el hardware y software de reconocimiento óptico de caracteres encargado de leer las patentes de los vehículos y enviar la información para validar el acceso o registrar la salida.
- **Pasarela de Pago:** Actor sistémico externo (sistema bancario o financiero) que permite procesar las transacciones monetarias para las reservas de visitas. El sistema se comunica con este actor para iniciar el cobro y recibir la confirmación del pago exitoso o fallido.

2.5.2 Casos de uso

A continuación, se describen los casos de uso a modo de resumen para un mayor entendimiento del diagrama de casos de uso general presentado anteriormente. Para un mayor entendimiento y menos redundancia entiéndase “usuario” como los roles de Colaborador y Administrador.

- ✓ **CU-01 Crear usuario:** El usuario tiene como objetivo crear una cuenta de usuario en el sistema.
 - CU-01.1 Validar correo corporativo:** Funcionalidad de inclusión obligatoria que verifica que el correo ingresado pertenezca al dominio institucional del CAR.
- ✓ **CU-02 Mantener usuario:** Permite al Administrador gestionar las cuentas existentes, incluyendo la visualización, modificación de atributos o desactivación de usuarios.
- ✓ **CU-03 Mantener reserva:** Caso de uso central donde el usuario gestiona el ciclo de vida de una reserva (crear, modificar o cancelar). De este caso de uso se extienden varias funcionalidades específicas:
 - CU-03.1 Acceder lista de espera dinámica:** Permite al Colaborador inscribirse en una cola de espera cuando no hay estacionamientos disponibles.
 - CU-03.2 Transferencia de reserva:** Permite al usuario transferir su reserva confirmada a otro colaborador autorizado.
 - CU-03.3 Reserva preferencial:** Habilita la selección de estacionamientos exclusivos para usuarios con perfil preferencial (ej. movilidad reducida, gerencia).
 - CU-03.4 Registrar reserva externa:** Permite reservar un espacio para un externo. Esta funcionalidad incluye obligatoriamente el proceso de pago (CU-04).
 - CU-03.5 Reasignar estacionamiento liberado:** Proceso automático que se activa al cancelarse una reserva en el CU-03. El sistema asigna el cupo liberado al siguiente usuario en la lista de espera.
- ✓ **CU-04 Realizar pago online por visita:** Permite procesar el pago de una reserva externa mediante la integración con una Pasarela de Pago. Este caso es invocado por el registro de visita.
- ✓ **CU-05 Validar acceso mediante OCR:** Proceso automático donde el Servicio OCR envía la patente leída al sistema para verificar si existe una reserva activa y permitir el acceso.
- ✓ **CU-06 Emitir notificación:** Servicio transversal del sistema que envía alertas automáticas (correo o push) a los usuarios ante eventos relevantes. Es incluido

por la creación de usuarios, gestión de reservas, reasignaciones, validaciones de acceso y penalizaciones.

- ✓ **CU-07 Generar reporte:** Permite al Administrador obtener informes detallados y consolidados sobre el uso de estacionamientos, ocupación y penalizaciones.
- ✓ **CU-08 Iniciar sesión:** Permite a los usuarios autenticarse en el sistema mediante sus credenciales.

CU-08.1 Validar rol: Funcionalidad de inclusión que determina los permisos y la interfaz que se mostrará al usuario según su perfil (Administrador o Colaborador).

- ✓ **CU-09 Mantener penalización:** Permite al Administrador crear, modificar o eliminar penalizaciones aplicadas a los usuarios por mal uso del servicio. El sistema notifica automáticamente al usuario afectado (CU-06).

2.6 Matriz de Trazabilidad

La matriz de trazabilidad es una herramienta analítica fundamental en la ingeniería de software utilizada para asegurar la integridad y consistencia del diseño del sistema. Su propósito es mapear de forma bidireccional los Requerimientos Funcionales (RF) con los Casos de Uso (CU). Mediante este cruce de información, se verifica técnicamente que cada requerimiento levantado esté cubierto por al menos un escenario de interacción en el diseño y se garantiza que no existan funciones modeladas que carezcan de justificación formal en los requerimientos.

Tabla 2-3 Matriz de trazabilidad.

CU/RF	CU-01	CU-01.1	CU-02	CU-03	CU-03.1	CU-03.2	CU-03.3	CU-03.4	CU-03.5	CU-04	CU-05	CU-06	CU-07	CU-08	CU-08.1	CU-09
RF-01	X	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					
RF-14	X			X		X			X		X					X
RF-15						X										
RF-16													X			
RF-17							X									
RF-18														X	X	
RF-19																X
RF-20																X
RF-21																X
RF-22																X

Fuente: Elaboración propia.

2.7 Modelo Conceptual

Las entidades principales del negocio y sus relaciones semánticas fueron abstraídas mediante un modelo conceptual [7], estableciendo el vocabulario común del sistema. Previo a su visualización, se describen las entidades de manera breve para un mejor entendimiento del dominio.

Tabla 2-4 Descripción de entidades del modelo conceptual

Entidad	Descripción
Persona	Representa a cualquier individuo en el sistema. Puede ser un trabajador o visitante. Contiene datos generales como nombre e identificador.
Trabajador	Subclase de Persona que corresponde a usuarios internos del sistema, con credenciales y rol.
EstadoTrabajador	Define los posibles estados del trabajador (activo, suspendido, etc.).
DetalleTrabajador	Registro histórico de los cambios de estado de un trabajador.
Penalizacion	Registro de las penalizaciones aplicadas a un trabajador, incluyendo duración y motivo.
EstadoPenalizacion	Define los posibles estados de la penalización (activa, finalizada, etc.)
DetallePenalizacion	Registro histórico de los cambios de estado de una penalización.
Reserva	Entidad central que registra las reservas de estacionamiento. Incluye tipo, estado, fechas, y datos del visitante si aplica.
DetalleReserva	Registro histórico de los cambios de estado de una reserva (trazabilidad).
EstadoReserva	Define los estados posibles de una reserva (activa, cancelada, finalizada, etc.).
Estacionamiento	Representa los espacios físicos disponibles para reserva. Contiene ubicación y estado.
TipoEstacionamiento	Clasificación de los estacionamientos: normal, preferencial, etc.

Pago	Información de la transacción asociada a una reserva externa. Incluye monto, método y estado.
TransferenciaReserva	Registro la transferencia de una reserva activa de un trabajador a otro.
Notificacion	Registro de los mensajes enviados automáticamente por el sistema a los usuarios.
ListaEspera	Cola de solicitudes pendientes cuando no hay disponibilidad inmediata. Permite reasignaciones automáticas.
ValidacionOCR	Registra los eventos de lectura de patente para validar acceso.
Reporte	Representa los reportes generados por el sistema, incluyendo tipo, formato y fecha.

Fuente: Elaboración propia.

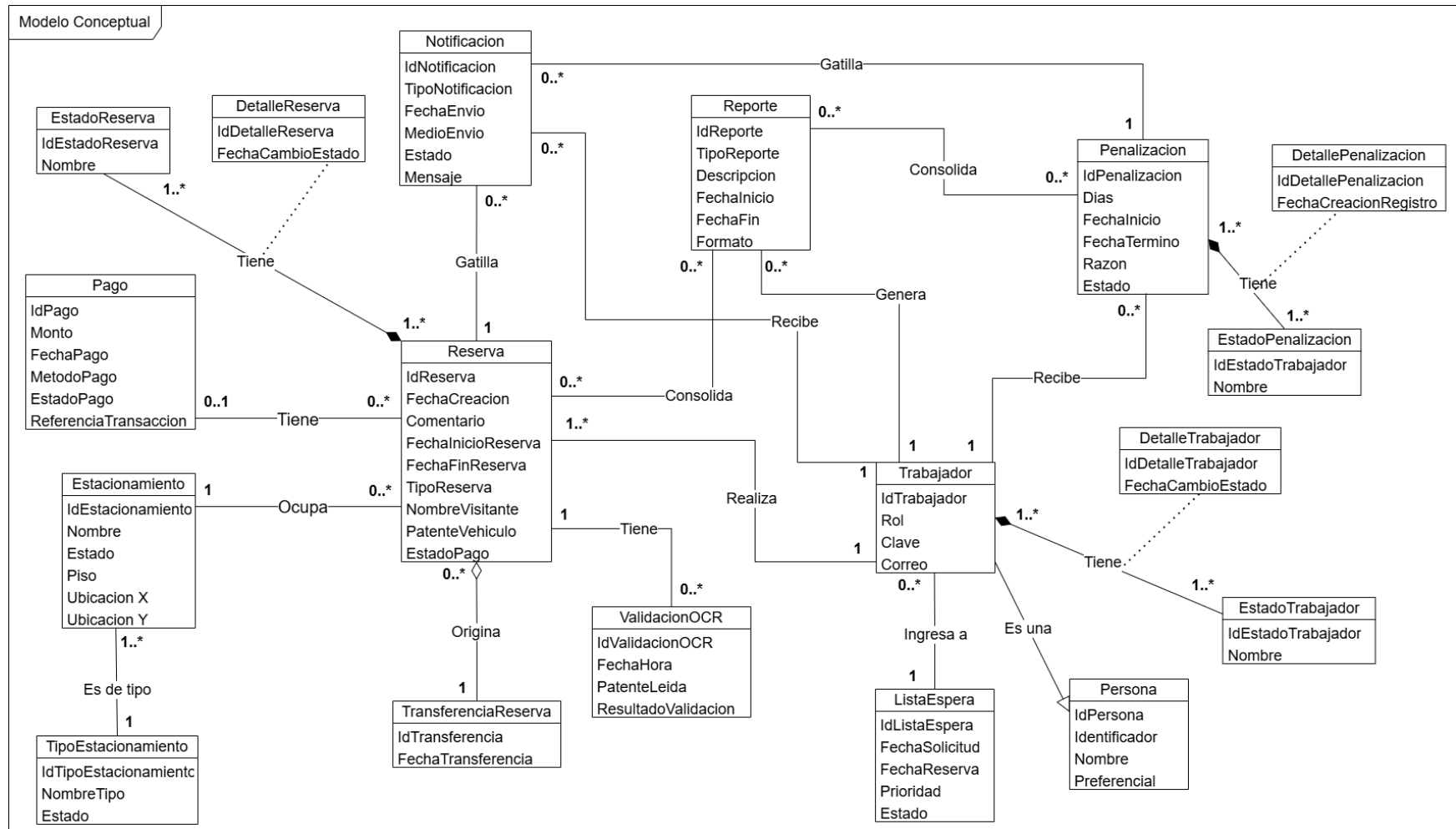


Figura 2-3 Modelo conceptual

Fuente: Elaboración propia.

2.8 **Modelo de Casos de Uso**

En esta sección se presentan los modelos de casos de uso del sistema, los cuales permiten representar gráficamente las interacciones entre los distintos actores y sus funcionalidades.

Cada requerimiento funcional fue desglosado utilizando el formato de casos de uso narrativos, lo que permite detallar los flujos principales y alternativos de la interacción entre el actor y el sistema [7].

Para modelar la interacción dinámica de los componentes durante la ejecución de los casos de uso críticos, se elaboraron diagramas de secuencia [8].

2.8.1 Caso de Uso CU-01: Crear usuario

Tabla 2-5 Descripción narrativa CU-01 Crear usuario

Caso de Uso	CU-01: Crear usuario
Actor	Usuario (Administrador y colaborador).
Descripción	Permite a un Colaborador crear su cuenta utilizando su correo corporativo, o bien el Administrador registrar un nuevo usuario en el sistema. El sistema valida el correo y crea el perfil con los permisos básicos asignados.
Referencias Cruzadas	RF-01 Crear nuevo usuario, CU-01.1 Validar correo corporativo, RF-14 Notificaciones automáticas.
Precondición	El colaborador debe contar con un correo corporativo valido. El administrador debe tener sesión iniciada y los privilegios de gestión de usuarios.
Postcondición	El nuevo usuario quedó registrado en la base de datos en estado activo y con acceso al sistema mediante sus credenciales.
Flujo Normal	
Acciones del Actor	Sistema
1. Inicia el proceso de registro.	2. Muestra formulario de creación de usuario.
3. Completa el formulario con los datos de	5. Valida el formulario. 6. Validar correo corporativo.

registro (nombre, correo, contraseña, preferencial). 4. Envía el formulario para la creación del usuario.	7. Verifica que el correo no este previamente registrado. 8. Crea el usuario en la base de datos. 9. Muestra mensaje "Usuario creado correctamente". 10. Redirige al inicio.
Flujo Alternativo: 1.- Correo invalido	
Acciones del Actor	Sistema
5A. Envía formulario para la creación del usuario.	6A. Validar correo corporativo. 7A. Muestra mensaje "El correo ingresado no se encuentra dentro del dominio". 8A. Redirige al formulario de creación de usuario.
Flujo Alternativo: 2.- Correo ya registrado	
Acciones del Actor	Sistema
5B. Envía formulario para la creación del usuario.	6B. Validar correo corporativo. 7B. Verifica que el correo no este previamente registrado. 8B. Muestra mensaje "El correo ya se encuentra registrado". 9B. Redirige al formulario de creación de usuario.
Flujo Alternativo: 3.- Error de conexión	
Acciones del Actor	Sistema
	9C. Error de conexión con la BD. 10C. Graba log del error. 11C. Muestra mensaje "Error en la conexión a la base de datos, intente nuevamente". 12C. Redirige al inicio.

Fuente: Elaboración propia.

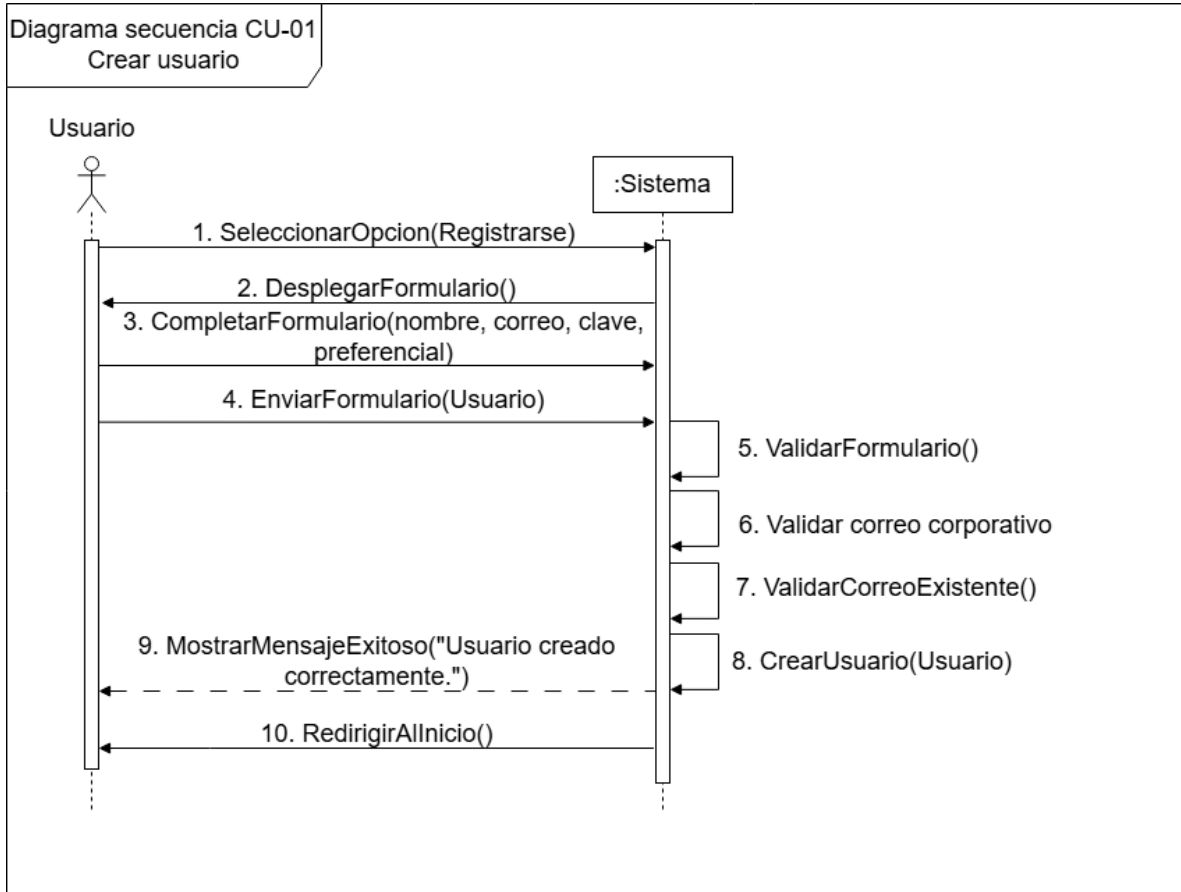


Figura 2-4 Diagrama de secuencia CU-01 Crear usuario

Fuente: Elaboración propia.

Tabla 2-6 Contrato CU-01 Crear usuario.

Operación	CrearUsuario
Parámetros	Nombre, correo corporativo, contraseña, preferencial.
Referencias cruzadas	RF-01: Crear nuevo usuario, CU-01.1 Validar correo corporativo.
Responsabilidad	Verificar que el correo no esté registrado previamente, incluir la validación de correo corporativo (CU-01.1), y registrar al nuevo usuario con estado activo y sus credenciales.
Salida	Confirmación del registro exitoso del usuario.
Excepciones	El sistema fracasa en la creación del usuario. Correo ya registrado. Correo fuera del dominio.

	Campos obligatorios incompletos.
Precondiciones	El correo ingresado debe pertenecer al dominio corporativo y no estar registrado previamente.
Postcondiciones	El usuario quedó almacenado con estado activo en la tabla Trabajador, vinculado a una Persona, con rol asignado y credenciales disponibles.

Fuente: Elaboración propia.

2.8.2 Caso de Uso CU-02: Mantener usuario

Tabla 2-7 Descripción narrativa CU-02 Mantener usuario.

Caso de Uso	CU-02: Mantener usuario
Actor	Administrador.
Descripción	Permite al administrador gestionar usuarios registrados. Esto incluye listar usuarios, modificar atributos del usuario y deshabilitar usuarios (que sería como el Delete). El sistema valida los datos ingresados y actualiza el registro mostrando mensajes de confirmación por cada acción realizada.
Referencias Cruzadas	RF-02: Listar usuarios, RF-03: Modificar usuario, RF-04: Eliminar usuario.
Precondición	El administrador debe tener sesión iniciada en el sistema.
Postcondición	El usuario seleccionado fue actualizado correctamente o deshabilitado (eliminación lógica).
Flujo Normal	
Acciones del Actor	Sistema
1. Selecciona la opción "Gestionar usuarios".	2. Ejecuta ListarUsuarios () y retorna los datos.
3. Visualiza usuarios. 4. Selecciona el usuario.	5. Despliega formulario de modificación.
6. Completa el formulario con los atributos a modificar (ej. preferencial, estado). 7. Envía el formulario.	8. Valida los datos del formulario. 9. Actualiza el usuario. 10. Muestra mensaje: "Usuario modificado exitosamente". 11. Redirige al inicio.
Flujo Alternativo: 1.- Deshabilitar usuario	

Acciones del Actor	Sistema
4B. Selecciona al usuario.	5B. Despliega formulario de modificación.
6B. Modifica el estado a "deshabilitado" 7B. Envía el formulario.	8B. Valida los datos del formulario. 9B. Actualiza el usuario (estado). 10B. Muestra mensaje "Usuario deshabilitado exitosamente." 11B. Redirige al inicio.
Flujo Alternativo: 2.- Error de conexión	
Acciones del Actor	Sistema
	9C. Error de conexión con la BD. 10C. Graba log del error. 11C. Muestra mensaje "Error en la conexión a la base de datos, intente nuevamente". 12C. Redirige al inicio.

Fuente: Elaboración propia.

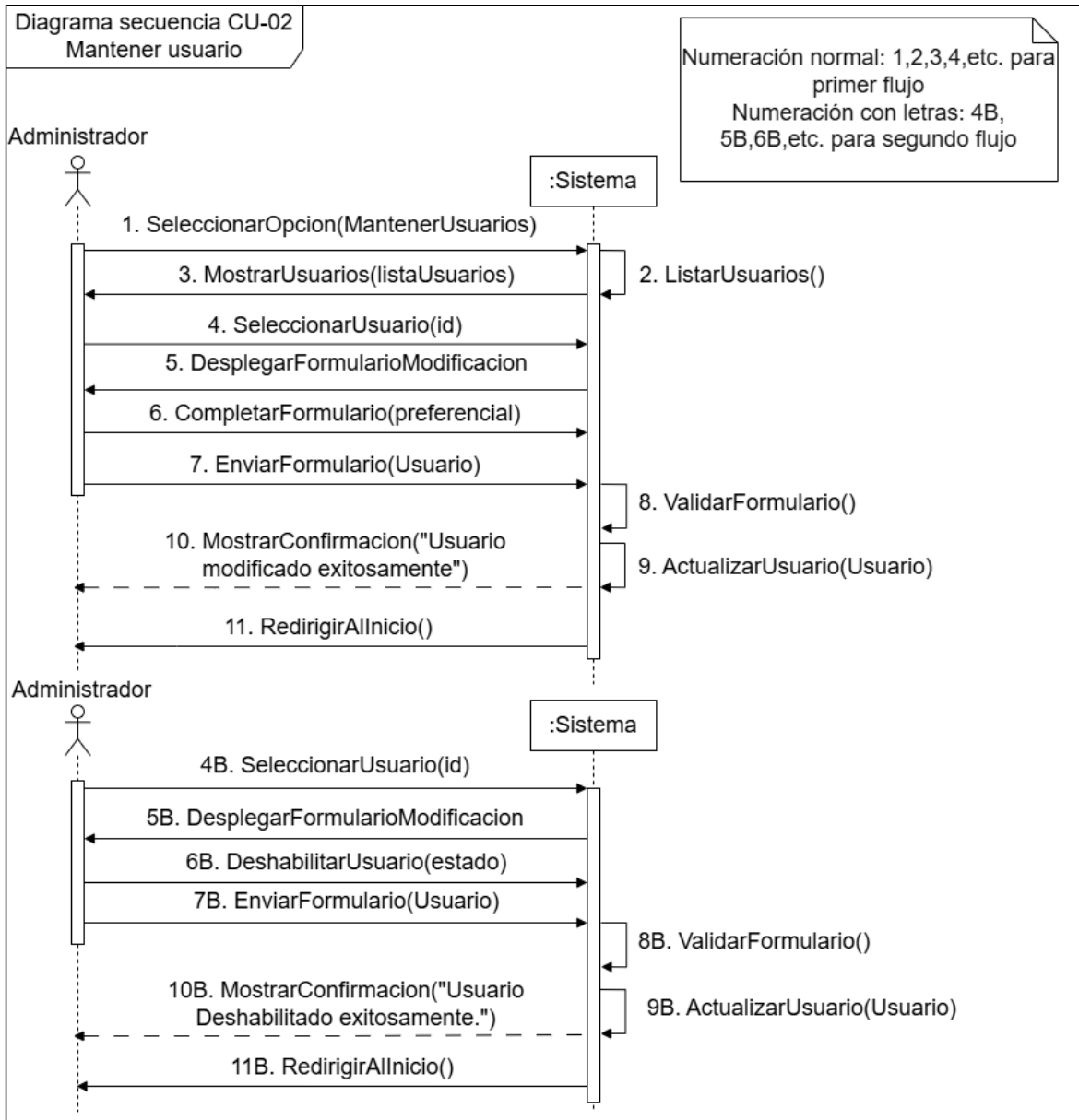


Figura 2-5 Diagrama de secuencia CU-02 Mantener usuario.

Fuente: Elaboración propia.

Tabla 2-8 Contrato CU-02 Mantener usuario

Operación	MantenerUsuario
Parámetros	idTrabajador, Nombre, Correo, Rol, Preferencial, Estado, Clave.
Referencias cruzadas	RF-02: Listar usuarios, RF-03: Modificar usuario, RF-04: Eliminar usuario.

Responsabilidad	Listar usuarios registrados, recibir datos actualizados para modificar un usuario, deshabilitar un usuario.
Salida	Confirmación de la operación exitosa.
Excepciones	El sistema fracasa en la operación.
Precondiciones	El Administrador debe tener sesión iniciada en el sistema.
Postcondiciones	El usuario quedó actualizado y los cambios se reflejaron inmediatamente.

Fuente: Elaboración propia.

2.8.3 Caso de Uso CU-03: Mantener reserva

Tabla 2-9 Descripción narrativa CU-03 Mantener reserva

Caso de Uso	CU-03: Mantener reserva
Actor	Usuario (Administrador y Colaborador).
Descripción	Permite al usuario gestionar sus reservas de estacionamiento, incluyendo la creación de nuevas, modificación de existente y cancelación de éstas. Extiende funcionalidades: lista de espera dinámica, transferencia de reserva, registro de reservas externas y reservas preferenciales.
Referencias Cruzadas	RF-05: Crear reserva, RF-06: Listar reserva, RF-07: Modificar reserva, RF-08: Cancelar reserva, RF-12: Pago online de visitas, RF-09: Reasignación automática, RF-10: Lista de espera dinámica (CU-03.1), RF-11: Registrar visita externa (CU-03.4), RF-14 Notificaciones automáticas, RF-15: Transferencia de reserva (CU-03.2), RF-17: Reserva preferencial (CU-03.3), RF-09 Reasignación automática (CU-03.5).
Precondición	El usuario debe tener sesión iniciada en el sistema.
Postcondición	Se registró, modificó o canceló una reserva. El sistema actualizó la disponibilidad y ejecutó los procesos automáticos de reasignación o notificación correspondientes.

Flujo Normal	
Acciones del Actor	Sistema
1. Selecciona opción "Crear reserva".	2. Despliega el selector de fecha
3. Selecciona la fecha para la reserva	4. Consulta disponibilidad interna para esa fecha. 5. Despliega el mapa de disponibilidad mostrando los estacionamientos libres.
6. Selecciona un estacionamiento disponible en el mapa. 7. Confirma la reserva.	8. Crea la reserva. 9. Notifica el evento de creación de reserva. 10. Muestra mensaje "Estacionamiento reservado exitosamente". 11. Redirige al inicio.
Flujo Alternativo: 1.- Modificar Reserva	
Acciones del Actor	Sistema
1B. Seleccionar opción "Mis Reservas".	2B. Consulta las reservas del usuario. 3B. Muestra las reservas.
4B. Selecciona una reserva.	5B. Despliega formulario de modificación de reserva.
6B. Completa el formulario (fecha de reserva). 7B. Envía el formulario.	8B. Valida el formulario. 9B. Valida la disponibilidad. 10B. Modifica la reserva. 11B. Muestra mensaje "Reserva modificada exitosamente". 12B. Notificación automática ("Reserva modificada exitosamente" + Reserva). 13B. Redirige al inicio.
Flujo Alternativo: 2.- Cancelar Reserva	
Acciones del Actor	Sistema
1C. Seleccionar opción "Mis Reservas".	2C. Consulta las reservas del usuario. 3C. Muestra las reservas.
4C. Selecciona una reserva.	5C. Despliega formulario de modificación de reserva.

6C. Selecciona opción "Cancelar Reserva". 7C. Ingresa motivo de cancelación. 8C. Envía el formulario.	9C. Valida el formulario. 10C. Cancela la reserva. 11C. Muestra mensaje "Reserva cancelada exitosamente". 12C. Notificación automática ("Reserva cancelada exitosamente" + Reserva). 13C. Redirige al inicio.
Flujo Alternativo: 3.- Error de conexión	
Acciones del Actor	Sistema
	9D. Error de conexión con la BD. 10D. Graba log del error. 11D. Muestra mensaje "Error en la conexión a la base de datos, intente nuevamente". 12D. Redirige al inicio.
Extensiones: CU-03.1 Acceder a lista de espera dinámica.	
Acciones del Actor	Sistema
4E. Si no hay estacionamientos disponibles puede acceder a una lista de espera.	5E. Registra lista de espera y pone al usuario en prioridad 1. 6E. Muestra mensaje "Estas en la lista de espera con prioridad 1". 7E. Redirige al inicio.
Extensiones: CU-03.2 Transferencia de reserva.	
Acciones del Actor	Sistema
4F. Selecciona una reserva.	5F. Despliega formulario de modificación de reserva.
6F. Selecciona opción "Transferir reserva". 7F. Selecciona usuario de destino. 8F. Envía formulario.	9F. Valida el formulario. 10F. Modifica la reserva actualizando el nuevo id del titular y queda registro de la reserva en ambos usuarios. 11F. Muestra mensaje "Reserva transferida exitosamente al usuario X". 12F. Notificación automática ("Reserva transferida exitosamente" + Reserva). 13F. Redirige al inicio.

Extensiones: CU-03.3 Reserva Preferencial	
Acciones del Actor	Sistema
4G. Selecciona un estacionamiento preferencial disponible.	5G. Valida atributo preferencial del usuario. 6G. Despliega formulario de reserva.
8G. Completa el formulario de reserva. 9G. Envía el formulario.	10G. Valida el formulario. 11G. Valida disponibilidad del estacionamiento. 12G. Crea la reserva. 13G. Muestra mensaje "Estacionamiento preferencial reservado exitosamente". 14G. Notificación automática ("Estacionamiento preferencial reservado exitosamente." + Reserva). 15G. Redirige al inicio.
Extensiones: CU-03.4 Registrar reserva externa	
Acciones del Actor	Sistema
4H. Selecciona un estacionamiento disponible.	5H. Despliega el formulario de reserva.
6H. Selecciona opción "Registrar visita externa". 7H. Completa el formulario con los datos de la visita.	8H. Valida el formulario. 9H. Valida la disponibilidad del estacionamiento. 10H. Crea la reserva. 11H. Despliega pasarela de pago CU-05: Pago online (visita).
12H. Completa el pago de la reserva.	13H. Muestra mensaje "Estacionamiento reservado exitosamente para la visita X". 14H. Notificación automática ("Estacionamiento reservado exitosamente para visita X" + Reserva + Comprobante de pago). 15H. Redirige al inicio.
Extensiones: CU-03.5 Reasignación automática	
Acciones del Actor	Sistema
	1. Detecta cancelación de una reserva existente.

	2. Verifica la coincidencia de horario en la lista de espera. 3. Ordena la lista según prioridad y fecha. 4. Selecciona el primer colaborador elegible. 5. Crea una nueva reserva con los datos del colaborador. 6. Cambia el estado del registro en la lista de espera.
--	--

Fuente: Elaboración propia.

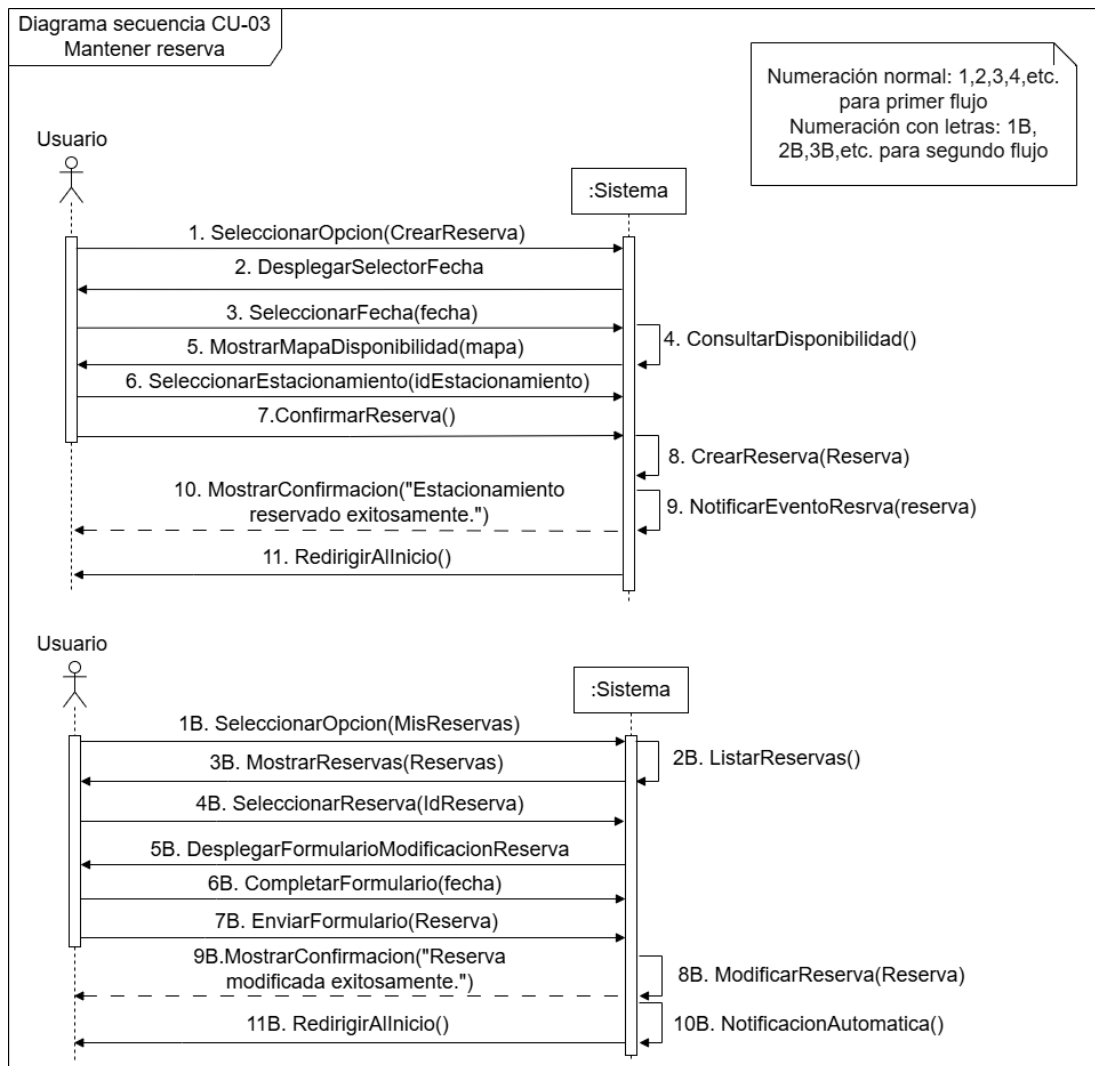


Figura 2-6 Diagrama de secuencia CU-03 Mantener reserva

Fuente: Elaboración propia.

También se diagramo el CU-03.5 Reasignar estacionamiento liberado, ya que es lo que agrega valor a la propuesta.

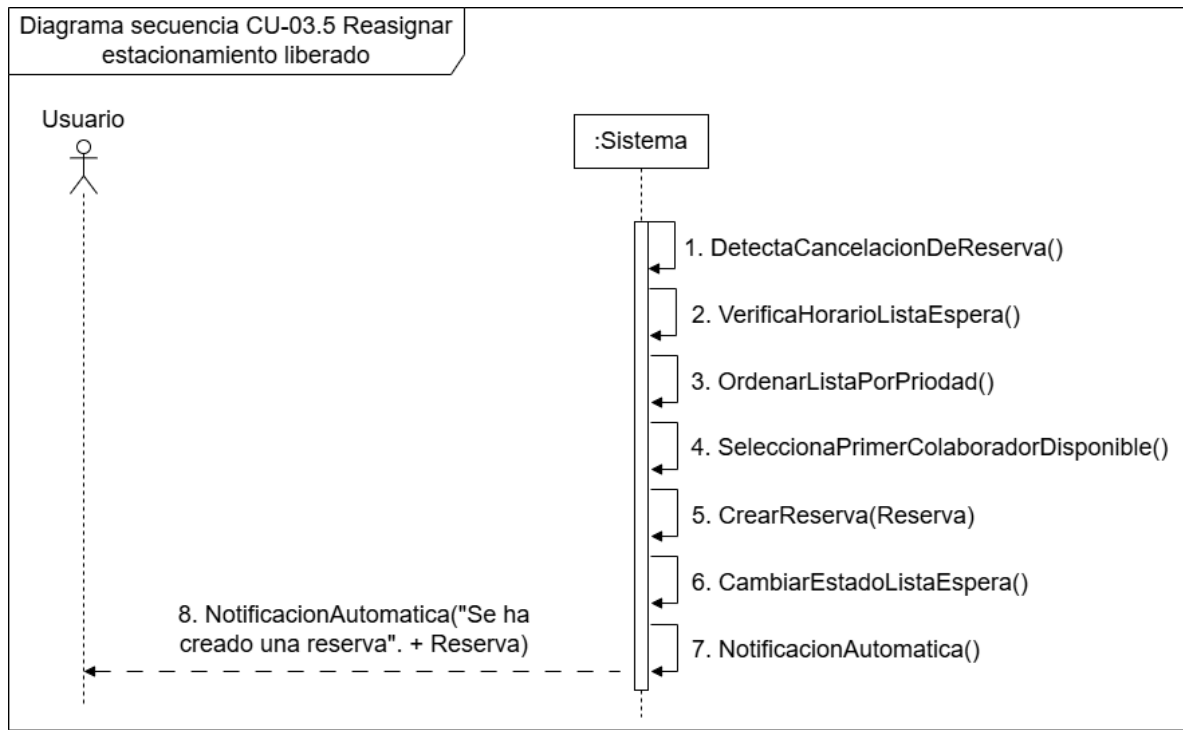


Figura 2-7 Diagrama de secuencia CU-03.5 Reasignar estacionamiento liberado

Fuente: Elaboración propia.

Tabla 2-10 Contrato CU-03 Mantener reserva

Operación	MantenerReserva
Parámetros	IdReserva, IdEstacionamiento, FechaCreacion, FechaInicioReserva, FechaFinReserva, TipoReserva, Comentario, NombreVisita (solo si es visita), PatenteVehiculo, Preferencial.
Referencias cruzadas	RF-05: Crear reserva, RF-06: Listar reserva, RF-07: Modificar reserva, RF-08: Cancelar reserva, RF-12: Pago online de visitas, RF-09: Reasignación automática, RF-10: Lista de espera dinámica (CU-03.1), RF-11: Registrar visita externa (CU-03.4), RF-15: Transferencia de reserva (CU-03.2), RF-17: Reserva preferencial (CU-03.3), RF-09 Reasignación automática (CU-03.5).

Responsabilidad	Permitir al colaborador o administrador registrar, modificar o cancelar una reserva. Validar la disponibilidad, generar notificaciones y disparar procesos automáticos asociados.
Salida	Confirmación visual de la acción realizada (reserva registrada, modificada o cancelada). Notificación enviada al usuario.
Excepciones	No hay disponibilidad en el horario seleccionado. Usuario con penalización activa. No autorizado para reservar estacionamiento preferencial.
Precondiciones	El usuario debe tener sesión iniciada en el sistema. El usuario no debe tener penalizaciones activas. El estacionamiento seleccionado debe estar activo y disponible.
Postcondiciones	Se creó o actualizó un registro en la entidad Reserva. Se generó un DetalleReserva para trazabilidad del estado. Se envió una Notificación automática. En caso de cancelación, se activó el CU-04 Reasignación automática. En caso de no disponibilidad, se creó una entrada en ListaEspera.

Fuente: Elaboración propia.

2.8.4 Caso de uso CU-04: Realizar pago online por visita

Tabla 2-11 Descripción narrativa CU-04 Realizar pago online por visita

Caso de Uso	CU-04: Realizar pago online por visita
Actor	Usuario (Colaborador o Administrador que registra la visita).
Descripción	Este caso de uso permite realizar el pago en línea de una reserva de estacionamiento para un visitante, luego de haber sido registrada por un usuario autorizado.
Referencias Cruzadas	RF-12: Pago online visitas.

Precondición	El usuario debe haber registrado previamente una reserva externa a través del CU.03.4
Postcondición	La reserva quedó registrada con su estado de pago confirmado y se notificó al usuario del éxito del proceso.
Flujo Normal	
Acciones del Actor	Sistema
1. CU-03.4 Registrar reserva externa. 2. Selecciona la opción "Registrar visita externa". 3. Completa el formulario con los datos del visitante y la reserva. 4. Envía el formulario.	5. Valida el formulario. 6. Verifica la disponibilidad del estacionamiento. 7. Despliega la pasarela de pago.
8. Realiza el pago de la reserva.	9. Crea la reserva. 10. Muestra mensaje "Estacionamiento reservado exitosamente para la visita X". 11. Notificación automática ("Estacionamiento reservado exitosamente para visita X" + Reserva + Comprobante de pago). 12. Redirige al inicio.
Flujo Alternativo 1.- No hay disponibilidad	
Acciones del Actor	Sistema
	5A. Valida el formulario. 6A. Verifica la disponibilidad del estacionamiento. 7A. Muestra mensaje "No hay disponibilidad, puede ingresar a una lista de espera". 8A. Redirige al formulario de reserva.
Flujo Alternativo 2.- Pago fallido	
Acciones del Actor	Sistema
8B. Intenta realizar el pago de la reserva.	7B. Despliega la pasarela de pago.

	9B. Pierde conexión con pasarela de pago u otro problema con la pasarela de pago. 10B. Muestra mensaje "Proceso de pago interrumpido, intente nuevamente". 11B. Redirige al formulario de reserva.
Flujo Alternativo 3.- Pago cancelado	
Acciones del Actor	Sistema
8C. Selecciona la opción cancelar	7C. Despliega pasarela de pago.
	9C. Muestra mensaje "Pago cancelado". 10C. Redirige al formulario de la reserva.
Flujo Alternativo 4.- Error de conexión	
Acciones del Actor	Sistema
	6D. Error de conexión con la BD. 7D. Graba log del error. 8D. Muestra mensaje "Error en la conexión a la base de datos, intente nuevamente". 9D. Redirige al inicio.

Fuente: Elaboración propia.

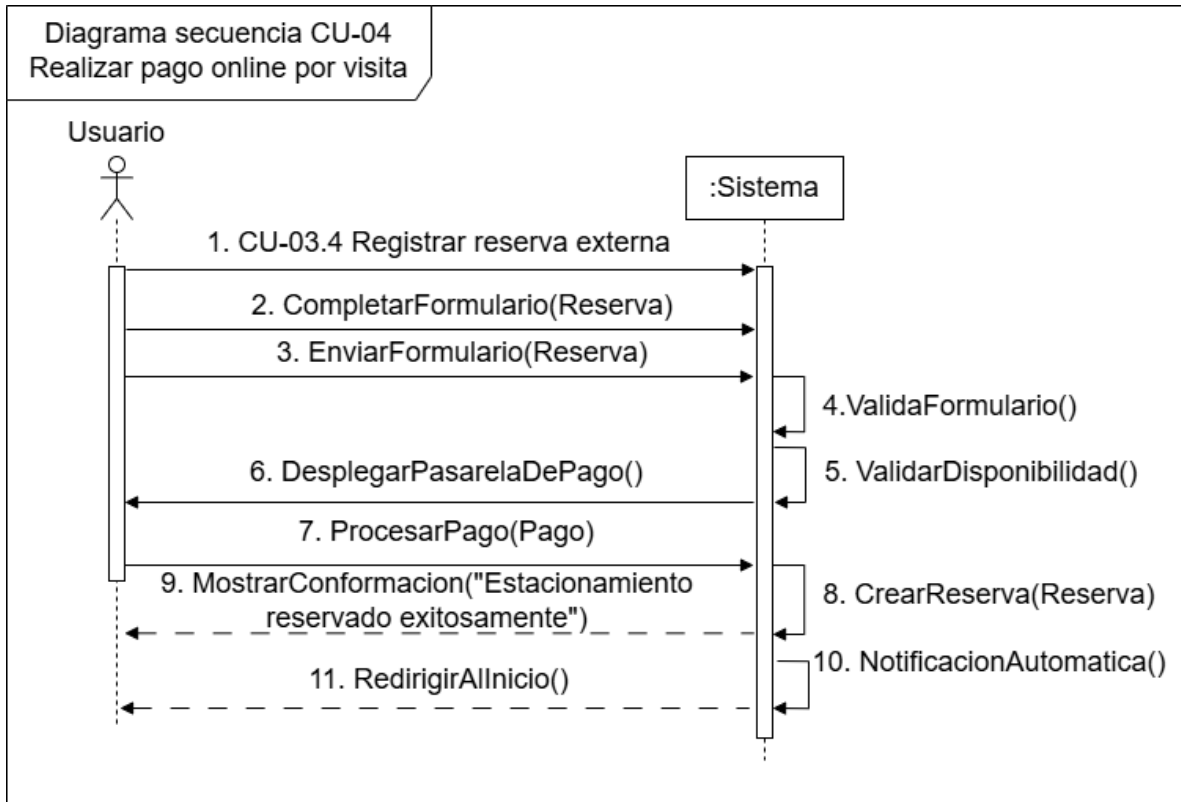


Figura 2-8 Diagrama de secuencia CU-04 Realizar pago online por visita

Fuente: Elaboración propia.

Tabla 2-12 Contrato CU-04 Realizar pago online por visita

Operación	ProcesarPagoReserva
Parámetros	IdReserva, nombreVisitante, patenteVehiculo, tipoReserva, monto, metodoPago, estadoPago,
Referencias cruzadas	RF-12: Pago online visitas
Responsabilidad	Validar datos de la reserva y del formulario, verificar disponibilidad, procesar pago, registrar reserva, notificar resultado.
Salida	Mensaje de confirmación de pago exitoso.
Excepciones	Datos inválidos. Disponibilidad no existente. Fallo en el procesamiento del pago.
Precondiciones	Reserva de tipo visita debe estar registrada (CU-03.4 completado)

Postcondiciones	La reserva quedó registrada con pago exitoso y estado actualizado. La notificación fue enviada.
------------------------	---

Fuente: Elaboración propia.

2.8.5 Caso de uso CU-05: Validar acceso mediante OCR

Tabla 2-13 Descripción narrativa CU-05 Validar acceso mediante OCR

Caso de Uso	CU-05 Validar acceso mediante OCR
Actor	Dispositivo OCR
Descripción	Este caso de uso permite validar el ingreso y uso de un vehículo al estacionamiento mediante la lectura de la patente por parte del sistema OCR. La patente leída se envía al sistema, el cual verifica si existe una reserva activa asociada a dicha patente para la fecha y hora actual. El resultado de la validación se registra y se envía una notificación automática al usuario correspondiente.
Referencias Cruzadas	RF-13: Integración con OCR, RF-14 Notificaciones automáticas.
Precondición	El sistema debe tener al menos una reserva activa para alguna patente y estar conectado al OCR.
Postcondición	El sistema validó el acceso, registró el ingreso en la tabla de validaciones y notificó el resultado.
Flujo Normal	
Acciones del Actor	Sistema
1. Leer la patente. 2. Enviar la patente leída (patenteLeida).	3. Valida si existe una reserva activa con esa patente y horario. 4. Registra la validación de la reserva (patenteLeida, resultado). 5. Notificación automática hacia el Usuario ("Reserva validada exitosamente).
Flujo Alternativo 1.- Lectura patente fallida	
Acciones del Actor	Sistema

1A. Lectura fallida. 2A. Envía mensaje "Lectura fallida".	3A. Notificación automática hacia el Administrador "Lectura fallida" + fecha y hora.
Flujo Alternativo 2.- Validación negativa	
Acciones del Actor	Sistema
	3B. Valida si existe una reserva activa con esa patente y horario. 4B. Registra validación negativa (patenteLeida, resultado). 5B. Notificación automática hacia el Administrador "Ingreso a estacionamiento sin reserva". 6B. Registra penalización al usuario que tenga asociada esa patenteLeida.
Flujo Alternativo 3.- Error de conexión	
Acciones del Actor	Sistema
	3C. Error de conexión con la BD. 4C. Graba log del error.

Fuente: Elaboración propia.

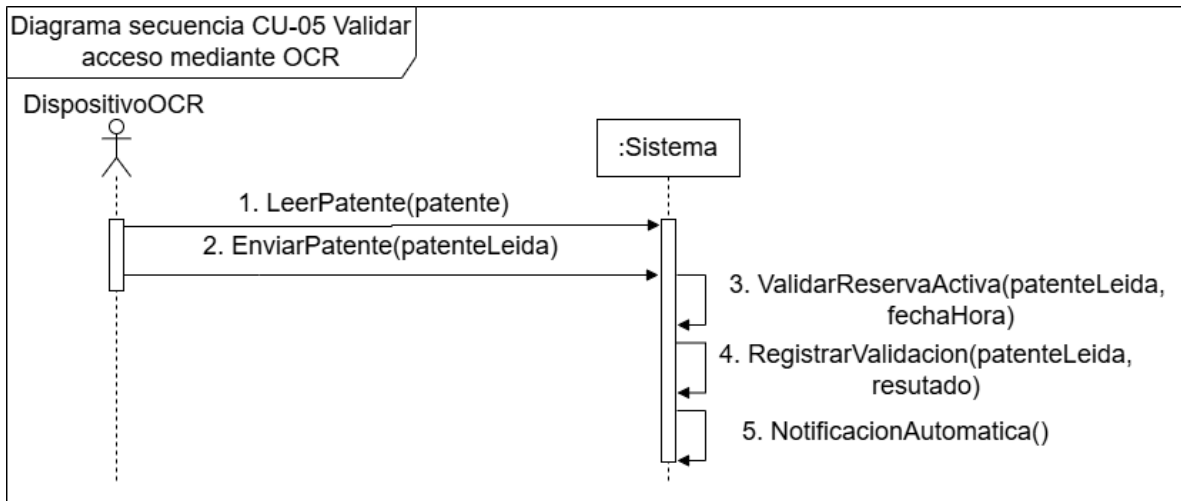


Figura 2-9 Diagrama de secuencia CU-05 Validar acceso mediante OCR

Fuente: Elaboración propia.

Tabla 2-14 Contrato CU-05 Validar acceso mediante OCR

Operación	ValidarAccesoOCR
------------------	------------------

Parámetros	patenteLeida, fechaHora
Referencias cruzadas	RF-13: Integración con OCR, RF-14 Notificaciones automáticas.
Responsabilidad	Verificar si la patente corresponde a una reserva activa en ese momento y registrar el resultado de la validación.
Salida	Resultado de la validación y notificación al usuario.
Excepciones	No hay reservas activas para esa patente. Error en la lectura del OCR. Fallo de conexión con el sistema.
Precondiciones	El sistema debe estar operativo y sincronizado con el OCR, con reservas registradas en la base de datos.
Postcondiciones	El sistema registró el resultado de la validación y envió la notificación automática al usuario correspondiente.

Fuente: Elaboración propia.

2.8.6 Caso de uso CU-06: Emitir notificación

Tabla 2-15 Descripción narrativa CU-06 Emitir notificación

Caso de Uso	CU-06 Emitir notificación.	
Actor	Usuario (Colaborador o Administrador).	
Descripción	Este caso de uso permite que los usuarios (Colaborador y Administrador) reciban notificaciones automáticas generadas por el sistema, relacionadas a eventos como confirmación, cancelación, transferencias, recordatorios previos, penalizaciones u otros cambios.	
Referencias Cruzadas	RF-14: Notificaciones automáticas.	
Precondición	Debe existir un evento registrado por el sistema que active una notificación (por ejemplo, creación de reserva, validación, pago, etc.).	
Postcondición	El usuario fue notificado a través del canal correspondiente según su configuración de preferencia.	
Flujo Normal		
Acciones del Actor		Sistema

5. Recibe la notificación por el medio indicado en sus preferencias.	1. Detecta un evento importante. 2. Genera un mensaje a partir del evento detectado (tipo, datos). 3. Selecciona el medio preferente del usuario(preferenciasUsuario). 4. Envía la notificación (mensaje, medio).
Flujo Alternativo 1.- Error de conexión	
Acciones del Actor	Sistema
	4A. Envía la notificación (mensaje, medio). 5A. Error de conexión. 6A. Registra log con información de la notificación y la excepción asociada.
Flujo Alternativo 2.- Usuario tiene notificaciones desactivadas	
Acciones del Actor	Sistema
	3B. Selecciona el medio preferente del usuario(preferenciasUsuario). 4B. Usuario tiene desactivadas las notificaciones. 5B. Se guarda notificación sin enviar.

Fuente: Elaboración propia.

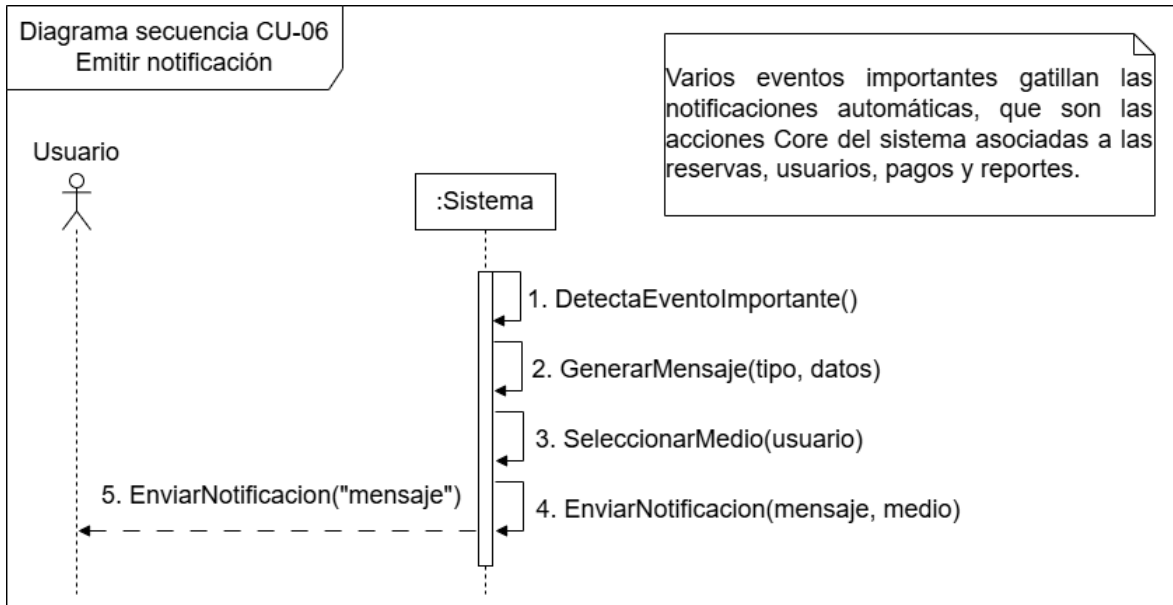


Figura 2-10 Diagrama de secuencia CU-06 Recibir notificaciones

Fuente: Elaboración propia.

Tabla 2-16 Contrato CU-06 Recibir notificaciones

Operación	EnviarNotificacionesAutomaticas
Parámetros	tipoNotificacion, datosEvento (objeto del evento, puede ser una Reserva, Pago, Validación, etc.), usuario, medio
Referencias cruzadas	RF-14: Notificaciones automáticas.
Responsabilidad	Generar el mensaje adecuado y enviarlo por el medio correspondiente.
Salida	Notificación enviada
Excepciones	Fallo en el envío por medio seleccionado. Usuario sin medio de notificación definido. Usuario tiene desactivadas las notificaciones. Fallo de conexión.
Precondiciones	Debe existir un evento del sistema que justifique la notificación
Postcondiciones	Se generó y envió la notificación al usuario correspondiente.

Fuente: Elaboración propia.

2.8.7 Caso de uso CU-07: Generar reporte

Tabla 2-17 Descripción narrativa CU-07 Generar reporte

Caso de Uso	CU-07 Generar reporte	
Actor	Administrador	
Descripción	Permite al Administrador generar reportes basados en tasa de ocupación, cancelaciones, reservas por periodo, uso por trabajador, etc.	
Referencias Cruzadas	RF-16: Generación de reportes.	
Precondición	El Administrador debe tener una sesión activa. Deben existir datos para generar el tipo de reporte seleccionado.	
Postcondición	El reporte solicitado quedó disponible para descarga.	
Flujo Normal		
Acciones del Actor		Sistema
1. Selecciona la opción "Generar Reporte".		2. Despliega formulario de generación de reporte.
3. Completa el formulario (indicando tipo de reporte y rango de fechas). 4. Envía el formulario.		5. Valida el formulario. 6. Consulta los datos requeridos. 7. Genera el reporte el reporte en el sistema. 8. Guarda el reporte en el sistema. 9. Muestra mensaje "Reporte generado exitosamente". 10. Pone a disposición del usuario el reporte para descargar.
Flujo Alternativo 1.- No existen datos		
Acciones del Actor		Sistema
		7A. Consulta los datos requeridos. 8A. No existen datos para los parámetros seleccionados. 9A. Muestra mensaje "No existen datos para los parámetros seleccionados". 10A. Redirige al formulario de reportes.
Flujo Alternativo 2.- Error en la generación		

Acciones del Actor	Sistema
	8B. Error al generar el reporte en el sistema. 9B. Graba log del error. 10B. Muestra mensaje "Error al generar el reporte, intente nuevamente". 11B. Redirige al formulario de reportes.
Flujo Alternativo 3.- Error de conexión	
Acciones del Actor	Sistema
	7D. Error de conexión con la BD. 8D. Graba log del error. 9D. Muestra mensaje "Error en la conexión a la base de datos, intente nuevamente". 10D. Redirige al inicio.

Fuente: Elaboración propia.

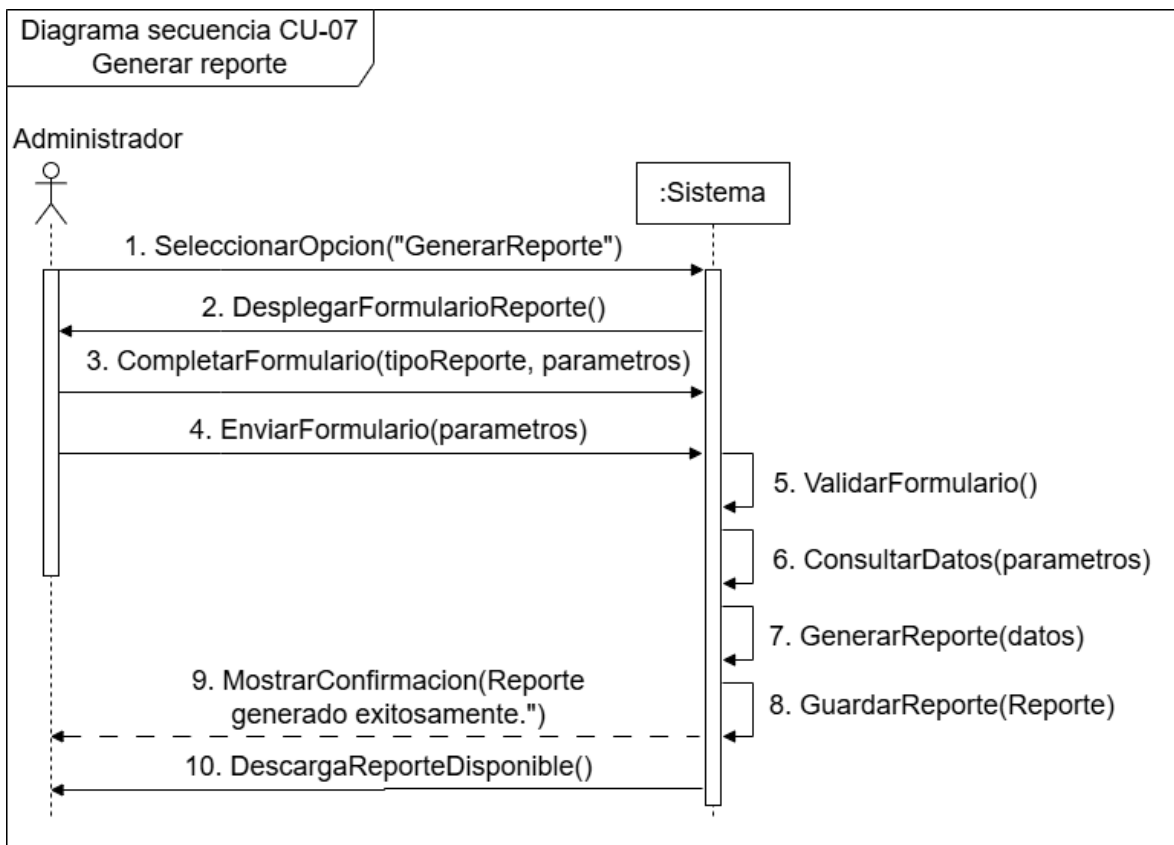


Figura 2-11 Diagrama de secuencia CU-07 Generar reporte

Fuente: Elaboración propia.

Tabla 2-18 Contrato CU-07 Generar reporte

Operación	GenerarReporte(parámetros)
Parámetros	tipoReporte, fechaInicio, fechaFin, Formato
Referencias cruzadas	RF-16: Generación de reportes.
Responsabilidad	El sistema debe validar los parámetros, consultar datos, generar el reporte, guardarlo y dejarlo disponible para descargar.
Salida	Reporte generado y mensaje de confirmación.
Excepciones	Parámetros inválidos. Falta de datos. Error de almacenamiento.
Precondiciones	El Administrador debe tener sesión iniciada en el sistema.
Postcondiciones	El reporte quedó disponible para descargar.

Fuente: Elaboración propia.

2.8.8 Caso de uso CU-08: Inicio de sesión

Tabla 2-19 Descripción narrativa CU-08 Inicio de sesión

Caso de Uso	CU-08 Inicio de sesión	
Actor	Usuario (Colaborador o Administrador)	
Descripción	El usuario accede al sistema ParkGYN ingresando sus credenciales. El sistema valida la información y redirige según su rol. Incluye la validación del rol del usuario.	
Referencias Cruzadas	RF-18: Inicio de sesión.	
Precondición	El usuario debe estar previamente registrado y contar con un estado habilitado.	
Postcondición	El sistema mostró la interfaz correspondiente al rol del usuario.	
Flujo Normal		
Acciones del Actor		Sistema
1. Selecciona opción “Iniciar sesión”.		2. Despliega el formulario de inicio de sesión (Login).

3. Completa el formulario (ingresando correo y clave). 4. Envía el formulario.	5. Valida el formulario. 6. CU-09.1 Validar rol. 7. Muestra mensaje "Sesión iniciada. Bienvenido." 8. Redirige al inicio correspondiente según rol.
Flujo Alternativo: 1.- Credenciales invalidas	
Acciones del Actor	Sistema
4A. Envía el formulario.	5A. Valida el formulario. 6A. Credenciales no coinciden. 7A. Muestra mensaje "Credenciales invalidas". 8A. Redirige al formulario de inicio de sesión.
Flujo Alternativo: 2.- Usuario no registrado	
Acciones del Actor	Sistema
4B. Envía el formulario.	5B. Valida el formulario. 6B. Credenciales no existen. 7B. Muestra mensaje "Usuario no encontrado". 8B. Redirige al formulario de inicio de sesión.
Flujo Alternativo: 3.-Usuario deshabilitado	
Acciones del Actor	Sistema
4C. Envía el formulario.	5C. Valida el formulario. 6C. Usuario estado deshabilitado. 7C. Muestra negación "Usuario deshabilitado por Administrador".
Flujo Alternativo: 4.- Error de conexión	
Acciones del Actor	Sistema
	7D. Error de conexión con la BD. 8D. Graba log del error. 9D. Muestra mensaje "Error en la conexión a la base de datos, intente nuevamente".

	10D. Redirige al inicio.
--	--------------------------

Fuente: Elaboración propia.

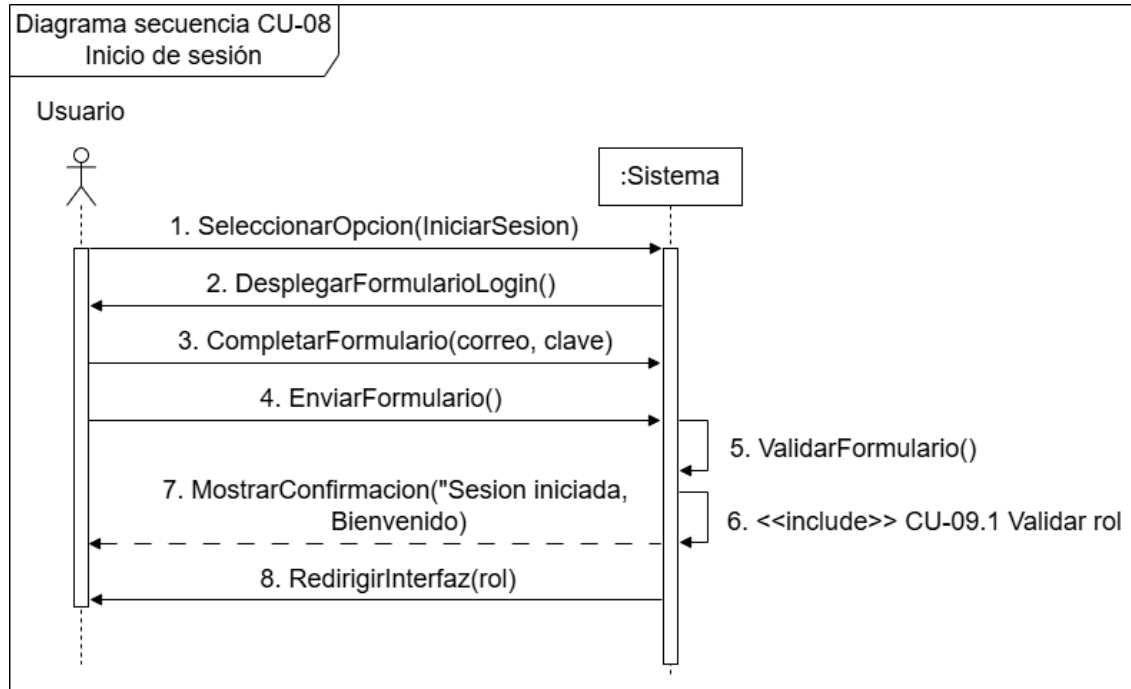


Figura 2-12 Diagrama de secuencia CU-08 Inicio de sesión

Fuente: Elaboración propia.

Tabla 2-20 Contrato CU-08 Inicio de sesión

Operación	IniciarSesion
Parámetros	correo, clave
Referencias cruzadas	RF-18: Inicio de sesión.
Responsabilidad	Validar las credenciales del usuario, determinar el rol y permitir el acceso si es válido.
Salida	Acceso al sistema y redirección según el rol (Colaborador o Administrador).
Excepciones	Credenciales invalidas. Usuario no registrado. Usuario deshabilitado.
Precondiciones	Usuario registrado y con estado habilitado.
Postcondiciones	El usuario quedó autenticado con acceso al sistema y con la vista correspondiente a su rol.

Fuente: Elaboración propia.

2.8.9 Caso de uso CU-09: Mantener penalización

Tabla 2-21 Descripción narrativa CU-09 Mantener penalización

Caso de Uso	CU-09: Mantener penalización		
Actor	Administrador		
Descripción	Permite al Administrador gestionar las penalizaciones dentro del sistema, ya sea la creación de nuevas, modificación de existentes y cancelación de éstas.		
Referencias Cruzadas	RF-19: Crear penalización, RF-20: Listar penalización, RF-21: Modificar penalización, RF-22: Eliminar penalización, RF-14 Notificaciones automáticas.		
Precondición	El usuario debe tener sesión iniciada en el sistema.		
Postcondición	Se registró, modificó o canceló una penalización. El sistema ejecutó los procesos automáticos de asignación y notificaciones automáticas.		
Flujo Normal			
Acciones del Actor		Sistema	
1. Selecciona opción "Crear penalización".		2. Consulta usuarios. 3. Muestra los usuarios.	
4. Selecciona al usuario.		5. Despliega el formulario de penalización.	
6. Completa el formulario. 7. Envía el formulario.		8. Valida el formulario. 9. Crea la penalización. 10. Muestra mensaje "Penalización creada exitosamente al usuario x". 11. Notificación automática al usuario penalizado. 12. Redirige al inicio.	
Flujo Alternativo: 1.- Modificar penalización			
Acciones del Actor		Sistema	
1B. Seleccionar la opción "Modificar penalización".		2B. Consulta y obtiene las penalizaciones. 3B. Muestra la lista de penalizaciones.	
4B. Selecciona la penalización a modificar.		5B. Despliega el formulario de modificación de penalización.	

6B. Completa el formulario (modificando los datos). 7B. Envía el formulario.	8B. Valida el formulario. 9B. Modifica la penalización. 10B. Muestra mensaje "Penalización modificada exitosamente." 11B. Notificación automática ("Penalización modificada exitosamente" + Penalización). 12B. Redirige al inicio.
Flujo Alternativo: 2.- Eliminar penalización	
Acciones del Actor	Sistema
1C. Selecciona la opción "Gestionar penalizaciones."	2C. Consulta y obtiene las penalizaciones. 3C. Muestra la lista de penalizaciones.
4C. Selecciona la penalización a eliminar	5C. Muestra mensaje de confirmación ¿Esta seguro?
6C. Selecciona la opción "Sí".	7C. Deshabilita la penalización (eliminación lógica). 8C. Muestra mensaje "Penalización eliminada al usuario x". 9C. Notificación automática ("Penalización deshabilitada" + Penalización). 10C. Redirige al inicio.
Flujo Alternativo: 3.- Error de conexión	
Acciones del Actor	Sistema
	9D. Error de conexión con la BD. 10D. Graba log del error. 11D. Muestra mensaje "Error en la conexión a la base de datos, intente nuevamente". 12D. Redirige al inicio.

Fuente: Elaboración propia.

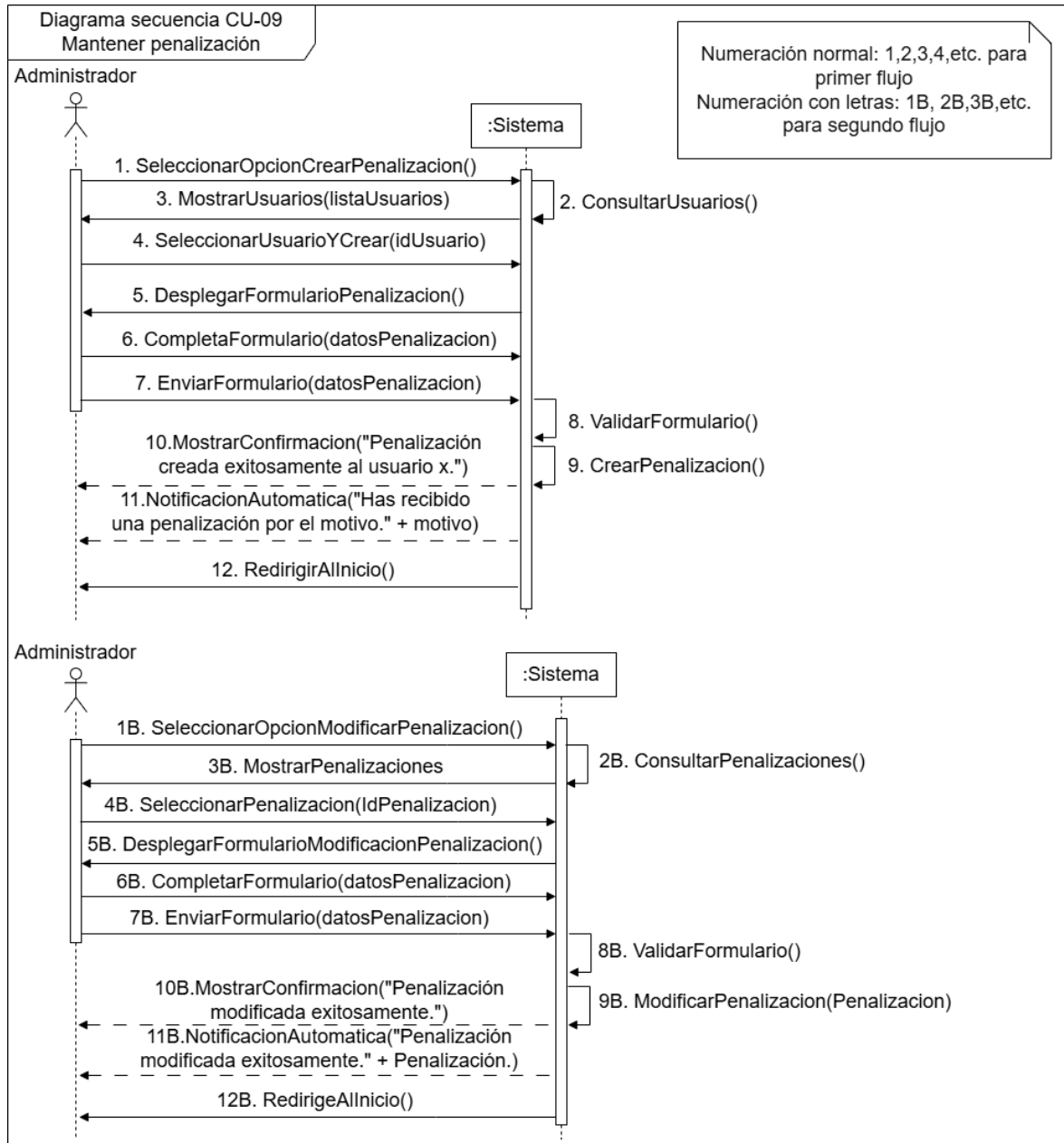


Figura 2-13 Diagrama de secuencia CU-09 Mantener Penalización

Fuente: Elaboración propia.

Tabla 2-22 Contrato CU-09 Mantener Penalización.

Operación	MantenerPenalizacion
Parámetros	idUsuario, idPenalizacion, días, fechaInicio, fechaTermino, razón, estado.

Referencias cruzadas	RF-19: Crear penalización, RF-20: Listar penalización, RF-21: Modificar penalización, RF-22: Eliminar penalización.
Responsabilidad	Validar los datos de entrada, permitir al Administrador crear, modificar o deshabilitar (eliminar lógicamente) una penalización a un usuario. El sistema debe registrar la acción y notificar al usuario afectado.
Salida	Confirmación de la operación exitosa (creada, modificada o eliminada) y envío de notificación al usuario afectado.
Excepciones	Datos del formulario inválidos. Usuario no registrado. Usuario deshabilitado.
Precondiciones	El usuario (Administrador) debe tener una sesión activa en el sistema.
Postcondiciones	La penalización quedó creada, modificada o deshabilitada en la base de datos. La notificación fue enviada al usuario afectado. El sistema mostró el mensaje de confirmación y redirigió al administrador.

Fuente: Elaboración propia.

CAPITULO 3. ASPECTOS RELEVANTES DISEÑO

3 ASPECTOS RELEVANTES DEL DISEÑO

A partir del análisis desarrollado en el capítulo anterior, en el presente capítulo se diseña la solución propuesta.

3.1 Arquitectura del software

La solución propuesta se basa en una arquitectura de software de tipo n-capas, esta opción es la más adecuada debido a sus numerosas ventajas en términos de modularidad, mantenibilidad, escalabilidad y separación de responsabilidades [9]. Al dividir el sistema en capas independientes, cada una con una responsabilidad específica, se logra un alto grado de aislamiento entre las distintas funcionalidades del sistema. Este aislamiento reduce las dependencias entre capas, lo que facilita el mantenimiento y permite realizar modificaciones en una capa sin afectar a las demás [10].

Además, se usará el patrón de diseño Modelo-Vista-Controlador (MVC) que mejorará aún más la organización del sistema. El MVC separa claramente la lógica de negocio, la interfaz de usuario y el controlador de flujo, permitiendo que cada componente evolucione de manera independiente. Esto resulta en mayor flexibilidad y escalabilidad, permitiendo la implementación de nuevas funcionalidades o modificaciones en la interfaz sin comprometer la lógica de negocio.

La abstracción entre capas es otro de los puntos fuertes de esta arquitectura. Al utilizar capas bien definidas, cada una se comunica con la siguiente capa de manera controlada, lo que garantiza que las interacciones sean claras y que cada capa se enfoque exclusivamente en su propósito específico. Eso contribuye a una mejor separación de responsabilidades, lo que también mejora la calidad del código, facilita la detección de errores y reduce la complejidad del sistema a medida que crece.

Por último, la arquitectura n-capas, al estar estructurada de forma modular, permite una escalabilidad eficiente. A medida que la aplicación crece, se pueden añadir nuevas capas o servicios sin alterar las capas existentes, lo que asegura que el sistema pueda adaptarse a futuros requerimientos sin comprometer su rendimiento o estabilidad [11].

A continuación, se presenta el diagrama de la arquitectura de software propuesta.

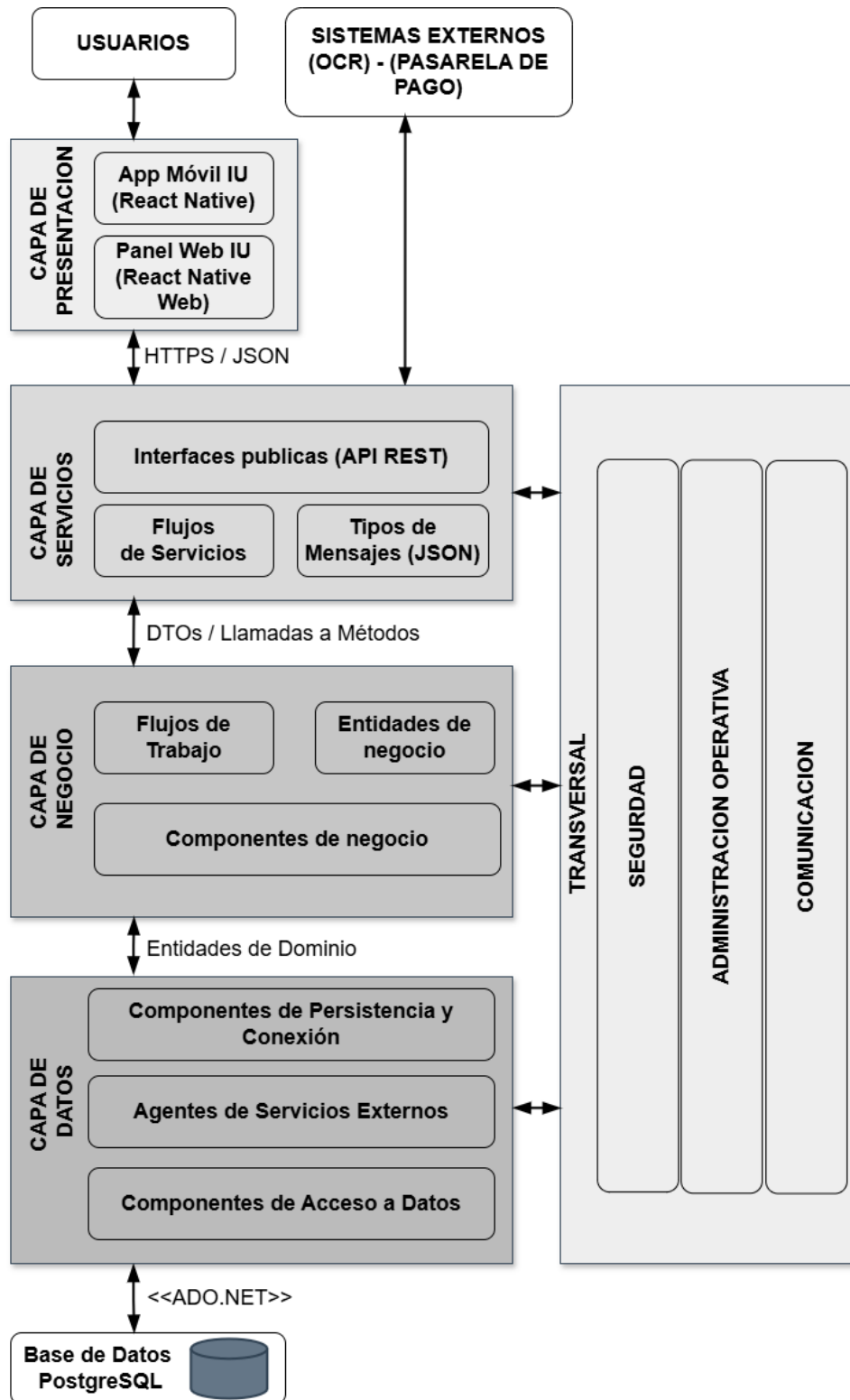


Figura 3-1 Diagrama de Arquitectura del Software

Fuente: Elaboración propia.

Capa de Presentación (corresponde a la Vista del patrón MVC): Esta capa se encarga de entregar la interfaz gráfica al usuario final. Provee los medios para que los colaboradores y administradores interactúen con el sistema de forma clara, amigable y desde múltiples dispositivos. Toda la comunicación entre esta capa y el sistema se realiza mediante solicitudes HTTP a las interfaces REST del backend.

- ✓ **App Móvil IU (React Native):** Aplicación destinada a dispositivos móviles para que los colaboradores realicen y gestionen sus reservas.
- ✓ **Panel Web IU (React Native Web):** Interfaz accesible vía navegador con las funcionalidades Core de la solución, por ejemplo, no cuenta con notificaciones automáticas.

Capa de Servicios (corresponde al Controlador del patrón MVC): Esta capa actúa como intermediaria entre la presentación y la lógica de negocio. Su función es recibir solicitudes externas (móvil, web, OCR, pasarela de pagos), validarlas y redirigirlas al componente correspondiente de la capa de negocio. Además, define el formato en que la información viaja por el sistema.

- ✓ **Interfaces Públicas (API REST):** Conjunto de endpoints accesibles externamente que exponen las funcionalidades del sistema de manera controlada. Que se llama públicas no quiere decir que estén disponibles para todo internet, sino que pueden acceder a ellas actores externos como el OCR.
- ✓ **Flujos de Servicios:** Lógica que orquesta el paso de datos entre capa de presentación, negocio y persistencia.
- ✓ **Tipos de Mensajes (JSON):** Estructuras estandarizadas de entrada/salida de información, utilizadas para mantener coherencia y trazabilidad en las solicitudes y respuestas.

Capa de Negocio (corresponde al Modelo del patrón MVC): Esta capa contiene la lógica central del sistema, aquella que representa las reglas reales del dominio. Es el núcleo funcional donde se ejecutan validaciones, cálculos, decisiones y flujos estructurados que determinan el comportamiento del sistema frente a cada evento.

- ✓ **Flujos de Trabajo:** Representan procesos completos (como crear o cancelar una reserva), definidos paso a paso.

- ✓ **Entidades de Negocio:** Modelos lógicos que representan objetos del dominio como Reserva, Usuario, Estacionamiento.
- ✓ **Componentes de Negocio:** Módulos que encapsulan funcionalidades específicas reutilizables, como aplicar penalizaciones o reasignar estacionamientos.

Capa de Datos: Responsable de persistir y recuperar la información del sistema. Se comunica directamente con la base de datos PostgreSQL utilizando ADO.NET, y también se encarga de integrar información proveniente de servicios externos como podrían ser el OCR o la pasarela de pago. Aunque esta capa no forma parte directa del patrón MVC, sirve como base para el Modelo, permitiendo que se mantenga sincronizado con los datos persistidos.

- ✓ **Componentes de Persistencia y Conexión:** Encargados de establecer y manejar las conexiones con la base de datos, controlando también las transacciones.
- ✓ **Componentes de Acceso a Datos (DAO):** Ejecutan consultas SQL y operaciones CRUD sobre las distintas tablas de la base de datos.
- ✓ **Agentes de Servicios Externos:** Gestionan la comunicación con sistemas como OCR o pasarela de pagos, transformando sus datos al formato interno y registrando los eventos en la base de datos.

Capa Transversal: Capa no jerárquica que entrega servicios de soporte al sistema completo. Aporta funcionalidad global relacionada con la seguridad, la administración operativa y la comunicación con el usuario.

- ✓ **Seguridad:** Mecanismos de autenticación, autorización por roles, cifrado de credenciales.
- ✓ **Administración operativa:** Registro de logs, auditoría, monitoreo y métricas de salud del sistema.
- ✓ **Comunicación:** Envío de notificaciones automáticas (push o correo electrónico) en base a eventos del sistema.

Gracias a esta estructura, la solución garantiza un desarrollo modular, con alta capacidad de evolución, integración con servicios externos y un diseño que respeta los principios de responsabilidad única, reutilización de código y separación de preocupaciones.

Una vez definida la estructura lógica del sistema, corresponde detallar la vista física de la solución. A continuación, se presenta el Diagrama de Despliegue, el cual ilustra la arquitectura física de 3 Niveles (3-Tier Architecture) seleccionada para el proyecto. Este esquema representa la distribución de los artefactos de software sobre los nodos de hardware (dispositivos y servidores), especificando los entornos de ejecución y los protocolos de comunicación que permiten la interacción entre el cliente, la lógica de negocio y la persistencia de datos.

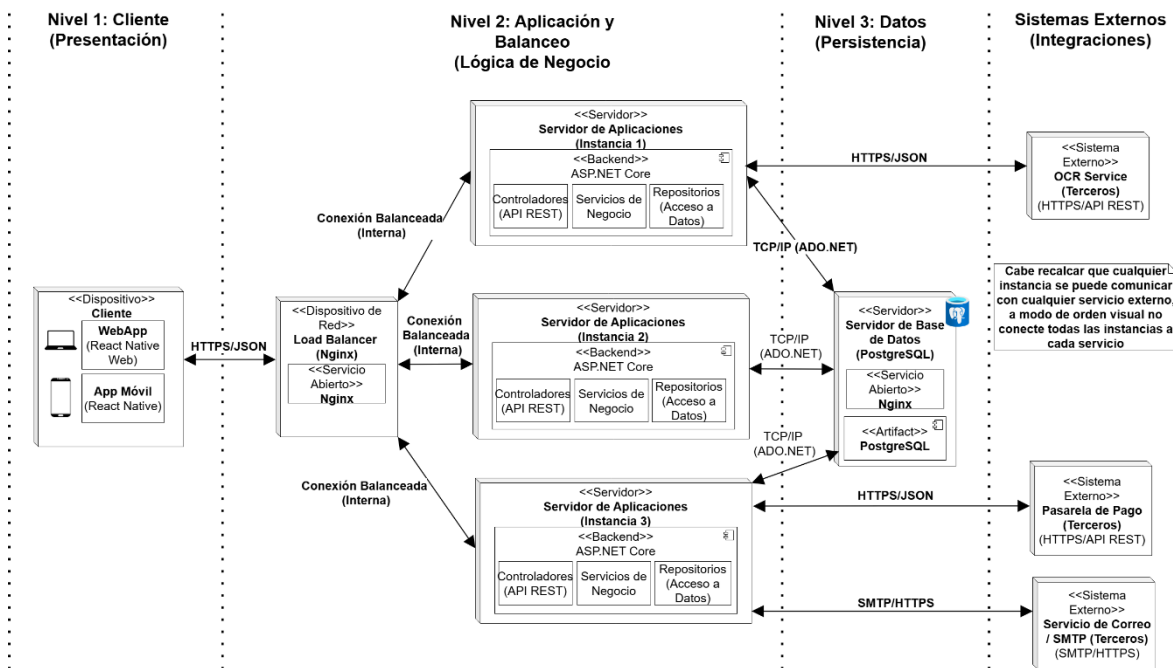


Figura 3-2 Diagrama de despliegue
Fuente: Elaboración propia

La arquitectura de despliegue se organiza en tres niveles físicos claramente segregados, cada uno con responsabilidades específicas de ejecución.

Nivel 1(Nivel de cliente): Este nivel representa el entorno de ejecución en el lado del usuario. Se compone de los dispositivos físicos (teléfonos inteligentes y computadores) que alojan y ejecutan la interfaz de usuario.

- ✓ **Componentes:** Se incluye la aplicación móvil (compilada en React Native) y el cliente web (React Native Web).
- ✓ **Función:** La responsabilidad de este nivel se limita a la presentación visual y la captura de datos. No se procesan reglas de negocio complejas ni se accede

directamente a la base de datos; actúa como un cliente ligero que consume los servicios del sistema mediante peticiones HTTP seguras.

Nivel 2 (Nivel de aplicación): Constituye el núcleo del procesamiento del sistema y la distribución del tráfico del sistema. Este nivel reside en una infraestructura escalable que aloja el Backend desarrollado en .NET (C#).

- ✓ **Componentes:** Se incorpora un Load Balancer (Balanceador de Carga) como punto de entrada de la red interna, seguido por múltiples instancias del servidor de aplicaciones donde se ejecutan los Controladores (API REST), los Servicios de Negocio y los Repositorios de acceso a datos
- ✓ **Función:** El Load Balancer recibe las solicitudes JSON provenientes del Nivel 1 y las distribuye equitativamente entre las instancias activas del backend. Esto sustenta la política de auto escalado y evita la saturación de los nodos durante los picos de demanda. Posteriormente, la instancia asignada procesa la lógica de negocio (validaciones, cálculos de disponibilidad, gestión de reservas) y coordina las transacciones hacia el nivel de datos.

Nivel 3 (Nivel de datos): Este nivel está conformado por un servidor independiente dedicado exclusivamente a la persistencia de la información.

- ✓ **Componentes:** Se ejecuta el motor de base de datos PostgreSQL.
- ✓ **Función:** Se garantiza la integridad y el almacenamiento seguro de los datos del sistema. Por motivos de seguridad, este nodo se encuentra aislado del acceso público directo y solo acepta conexiones vía protocolo TCP/IP provenientes del Nivel de Aplicación, utilizando drivers nativos.

Protocolos de Comunicación: La interacción entre el Nivel 1 y el Nivel 2 se realiza mediante el protocolo HTTPS intercambiando archivos en formato JSON, lo cual garantiza una comunicación estándar y segura a través de internet. Por su parte, la comunicación entre el Nivel 2 y el Nivel 3 es interna y utiliza el protocolo TCP/IP, optimizado para la transferencia rápida de datos entre el servidor de backend y el motor de base de datos.

3.1.1 Plataforma tecnológica y herramientas de desarrollo

Para el desarrollo del sistema propuesto, se ha seleccionado un conjunto de herramientas, tecnologías y plataformas que permiten cumplir con los requisitos funcionales y no funcionales definidos durante el análisis del proyecto. Estas decisiones fueron tomadas considerando criterios como la compatibilidad multiplataforma, la robustez, la facilidad de mantenimiento y la escalabilidad futura del sistema.

Sistema operativo: El desarrollo y despliegue del sistema se llevará a cabo sobre sistemas operativos compatibles con los entornos de ejecución de .NET y Node.js, siendo Windows 10 o superior el entorno principal para el desarrollo local. En cuanto a la ejecución en servidores, se contempla el uso de Linux (Ubuntu Server) por su estabilidad, bajo consumo de recursos y amplia compatibilidad con contenedores Docker, en caso de ser necesarios en etapas posteriores.

Características del recurso computacional: Hardware mínimo que tendrá cada miembro del equipo.

- ✓ Procesador: i5 10400.
- ✓ RAM: 16GB.
- ✓ Almacenamiento: 512GB de SSD.
- ✓ Tarjeta Gráfica: Integrada del procesador.
- ✓ Pantalla: Full HD (1920 x 1080) de 14" o superior.
- ✓ Sistema Operativo: Windows 10 Pro u Windows 11 Pro.

Herramientas necesarias para el desarrollo del proyecto: Para la ejecución y construcción del sistema, se determinó el uso de las siguientes herramientas tecnológicas:

- **Visual Studio 2022/Visual Studio Code:** Entorno de desarrollo integrado (IDE) para implementar el backend en C# utilizando ASP.NET Core.
- **Node.js y npm:** Para gestionar dependencias y scripts del entorno React Native.
- **PostgreSQL:** Sistema de gestión de bases de datos relacional elegido para almacenar toda la información del sistema.
- **DBeaver/pgAdmin:** Clientes para administración, diseño y consultas sobre la base de datos.

- **Draw.io:** Herramienta utilizada para el diseño de diagramas UML, arquitectura y modelos conceptuales.
- **Figma:** Plataforma de diseño colaborativo utilizada para la creación de wireframes, diseño de la interfaz de usuario (UI) y validación de prototipos visuales previo a la fase de codificación.
- **Postman:** Utilizado para probar endpoints y validar el comportamiento de las API REST expuestas.
- **Git y GitHub:** Para el control de versiones y colaboración entre los miembros del equipo.
- **Nginx:** Servidor web y proxy inverso de código abierto (Open Source). En la arquitectura de este proyecto, se utilizará como Balanceador de Carga (Load Balancer) para distribuir equitativamente el tráfico de red entrante hacia las múltiples instancias del servidor de aplicaciones, garantizando la alta disponibilidad y sustentando la política de auto escalamiento del backend sin incurrir en costos de licenciamiento.

Lenguajes de programación y tecnologías a usar:

- **C# con ASP.NET Core:** Lenguaje y frameworks a utilizar para implementar el backend del sistema. Se eligió C# por su robustez, orientación a objetos, soporte para programación asíncrona y su integración natural con ASP.NET Core, que permite crear servicios REST eficientes, seguros y escalables. Además ASP.NET Core es multiplataforma, lo cual otorga flexibilidad al momento de desplegar el sistema en distintos entornos (Windows o Linux).
- **React Native:** Utilizado para construir tanto la aplicación móvil como el panel web mediante React Native Web. Esta tecnología fue seleccionada por su capacidad de reutilización de código entre plataformas (iOS, Android y Web), su amplia comunidad de soporte, y su integración fluida con bibliotecas modernas para construcción de interfaces de usuario reactivas.
- **JavaScript/TypeScript:** Lenguajes principales utilizados en el desarrollo de capa de presentación. TypeScript, en particular, permite trabajar con tipado estático, facilitando la mantenibilidad del código y reduciendo errores en tiempo de desarrollo.
- **PostgreSQL:** Sistema gestor de base de datos relacional seleccionado por su rendimiento, integridad transaccional, escalabilidad y compatibilidad con herramientas open source. Además, permite trabajar fácilmente con datos

estructurados y ofrece extensiones útiles como pgAudit para auditoria, pgcrypto para cifrado, entre otras que podrían requerirse en el futuro.

- **ADO.NET:** Conjunto de clases del ecosistema .NET utilizadas para realizar operaciones CRUD sobre la base de datos desde el backend. Se optó por ADO.NET en lugar de un ORM como Entity Framework para tener un control más preciso sobre las consultas, maximizar el rendimiento y reducir la complejidad del modelo de datos.

3.2 **Diseño de Datos**

Esta sección presenta el diseño de datos propuesto para el sistema, el cual define la estructura lógica que permite representar, almacenar y gestionar la información requerida por los distintos procesos del negocio. El diseño se construyó en base a los RF identificados y los CU definidos previamente, asegurando coherencia entre las entidades del modelo y las operaciones del sistema.

3.2.1 **Modelo de Clases**

El modelo de clases representa la estructura fundamental de los datos dentro del sistema. Describe las entidades principales del dominio, sus atributos y las relaciones entre ellas, proporcionando una base sólida para la implementación de la lógica de negocio. Se listan todas las clases del modelo.

- ReservaController – ReservaService – ReservaRepository – Reserva – TransferenciaReserva – EstadoReserva – DetalleReserva
- EstacionamientoController – EstacionamientoService – EstacionamientoRepository – Estacionamiento – TipoEstacionamiento –
- OCRController – OCRService – OCRRepository – ValidacionOCR.
- PagoController – PagoService – PagoRepository – Pago.
- NotificacionService – NotificacionRepository – Notificacion.
- ReporteController – ReporteService – ReporteRepository – Reporte.
- PenalizacionController – PenalizacionService – PenalizacionRepository – Penalizacion – DetallePenalizacion – EstadoPenalizacion.
- TrabajadorController – TrabajadorService – TrabajadorRepository – Trabajador – DetalleTrabajador – EstadoTrabajador – Persona – ListaEsperaController – ListaEsperaService – ListaEsperaRepository – ListaEspera.

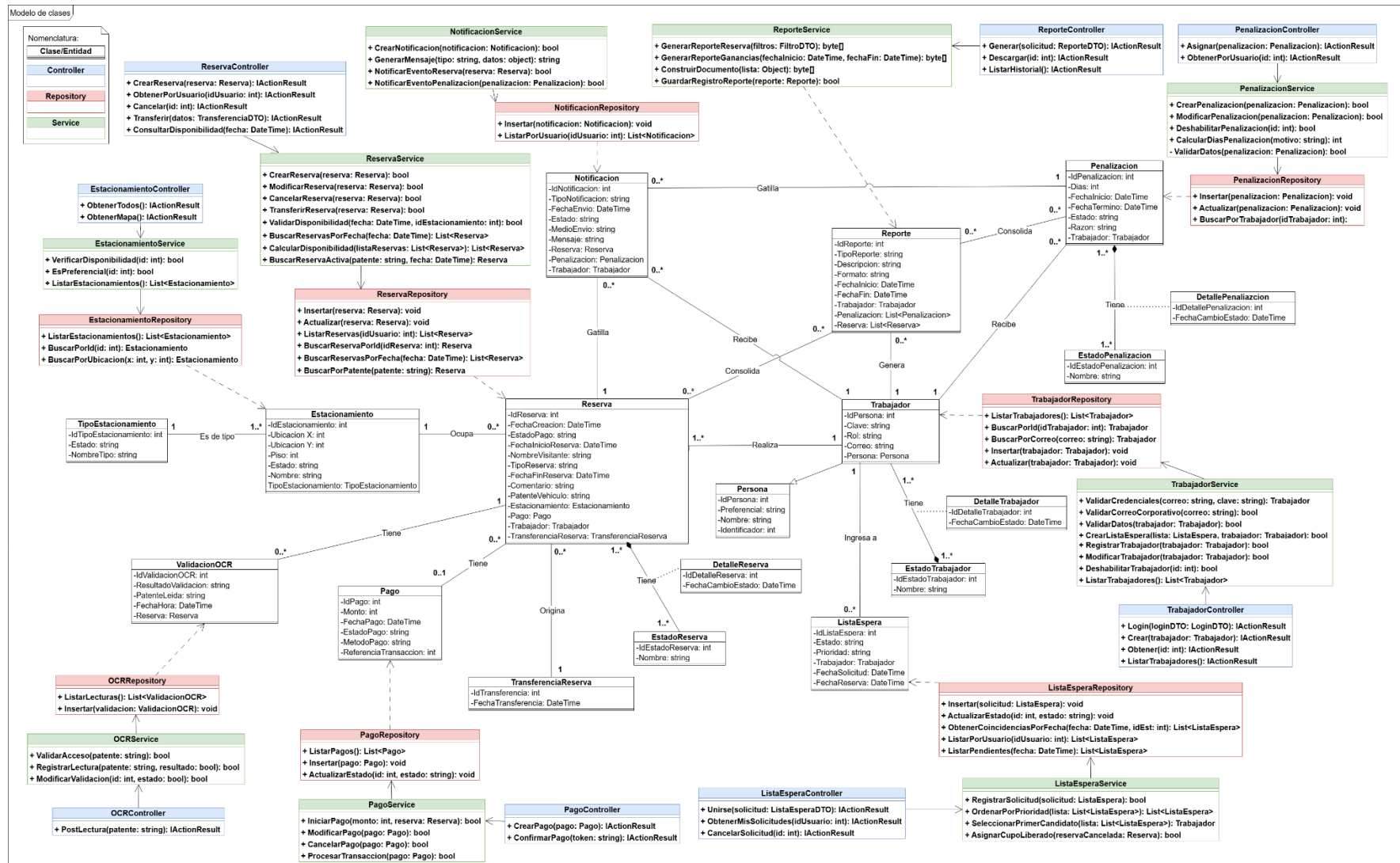


Figura 3-3 Modelo de Clases

Fuente: Elaboración propia.

3.2.2 Modelo Relacional

El modelo relacional del sistema describe cómo se estructuran y se interrelacionan las tablas dentro de la base de datos para almacenar la información de manera eficiente [12], a continuación, se presenta dicho modelo. Se listan las entidades del modelo.

1. EstadoReserva
2. DetalleReserva
3. Reserva
4. Pago
5. Estacionamiento
6. TipoEstacionamiento
7. TransferenciaReserva
8. Notificacion
9. ValidacionOCR
10. ReporteReserva
11. Reporte
12. ReportePenalizacion
13. Penalizacion
14. DetallePenalizacion
15. EstadoPenalizacion
16. Trabajador
17. DetalleTrabajador
18. EstadoTrabajador
19. Persona
20. ListaEspera

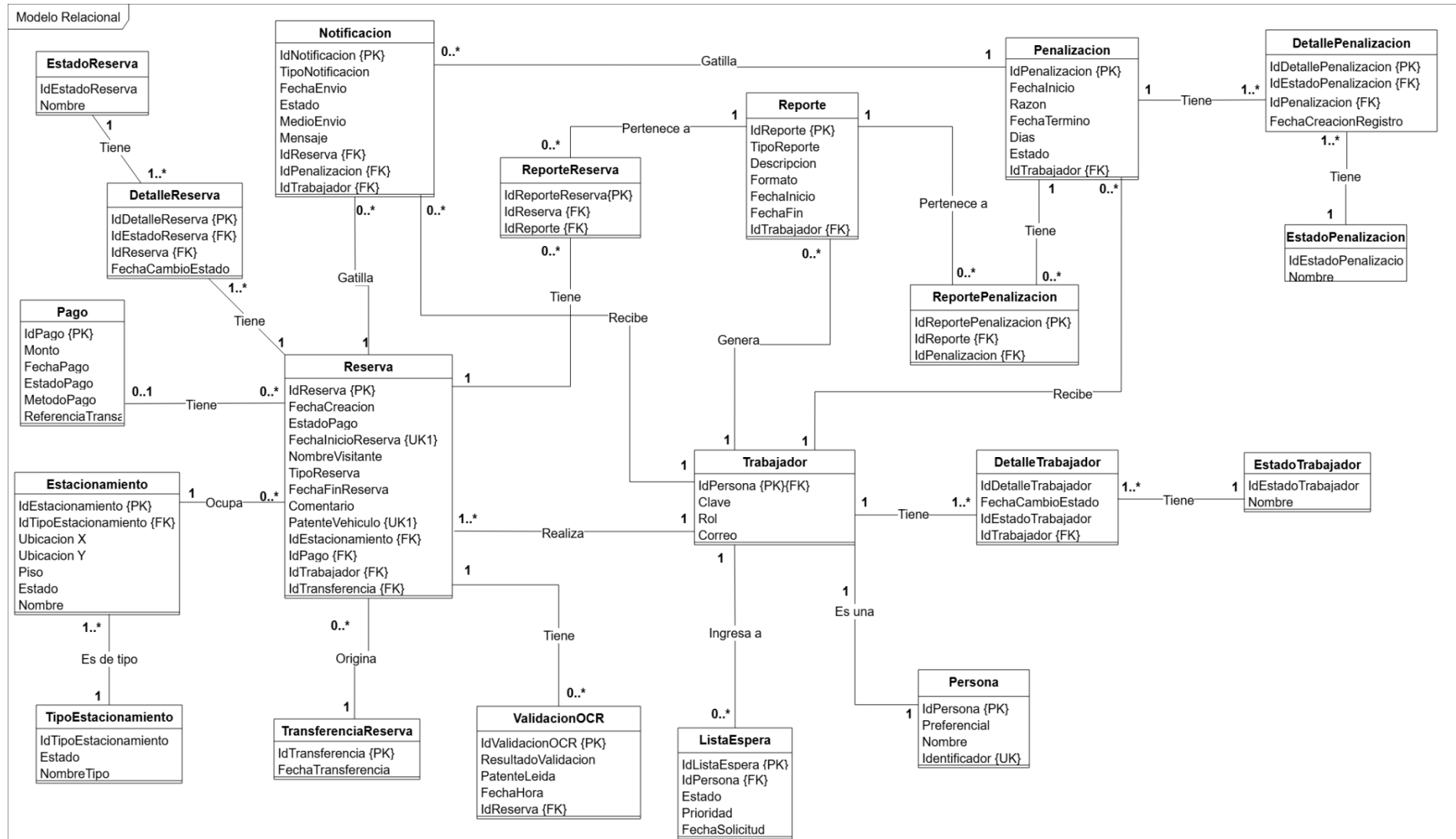


Figura 3-4 Modelo Relacional

Fuente: Elaboración propia.

3.2.3 Diccionario de Datos

Esta sección proporciona una lista detallada de las entidades, atributos, tipos de datos y las relaciones entre ellos. Además, facilita la consistencia y la integridad del modelo, sirviendo como referencia para los desarrolladores y administradores durante la implementación y mantenimiento del sistema.

Tabla 3-1 Tabla TipoEstacionamiento

NOMBRE	TipoEstacionamiento		
DESCRIPCIÓN	Tabla que almacena los tipos de estacionamientos disponibles, como Uso Común y Uso Preferencial, con la posibilidad de agregar más tipos en el futuro.		
CLAVE PRINCIPAL	IdTipoEstacionamiento		
CLAVES FORÁNEAS	N/A		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdTipoEstacionamiento	Int	N/A	Identificador único de cada tipo de estacionamiento en el sistema.
Estado	Varchar	50	Indica el estado del tipo de estacionamiento, como activo o inactivo.
NombreTipo	Varchar	50	Nombre que describe el tipo de estacionamiento como Común, Preferencial, etc.

Fuente: Elaboración propia.

Tabla 3-2 Tabla Estacionamiento

NOMBRE	Estacionamiento
DESCRIPCIÓN	Tabla que almacena la información de los estacionamientos, incluyendo su tipo, ubicación, estado. Esta tabla está vinculada a la tabla TipoEstacionamiento para identificar el tipo de cada Estacionamiento (Común o Preferencial).
CLAVE PRINCIPAL	IdEstacionamiento
CLAVES FORÁNEAS	IdTipoEstacionamiento

ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdEstacionamiento	Int	N/A	Identificador único de cada estacionamiento en el sistema.
IdTipoEstacionamiento	Int	N/A	Clave foránea que vincula el estacionamiento con un tipo específico de la tabla TipoEstacionamiento.
Ubicación X	Int	N/A	Coordenada horizontal X que define la ubicación del estacionamiento en el plano.
Ubicación Y	Int	N/A	Coordenada vertical Y que define la ubicación del estacionamiento en el plano.
Piso	Int	N/A	Número de piso o nivel donde se encuentra ubicado el estacionamiento.
Estado	Varchar	50	Estado del estacionamiento, podría ser Habilitado, Mantenimiento, Deshabilitado o Exclusivo
Nombre	Varchar	10	Nombre compuesto por letras y números. Ejemplo A01.

Fuente: Elaboración propia.

Tabla 3-3 Tabla Pago

NOMBRE	Pago		
DESCRIPCIÓN	Tabla que almacena los registros de pagos realizados, asociados a las reservas externas de estacionamiento. Cada registro contiene información sobre el monto pagado, la fecha, el estado del pago y el método utilizado.		
CLAVE PRINCIPAL	IdPago		
CLAVES FORÁNEAS	N/A		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN

IdPago	Int	N/A	Identificador único para cada pago realizado en el sistema.
Monto	Int	N/A	Monto total pagado en la transacción.
FechaPago	Datetime	N/A	Fecha en la que se realizó el pago.
EstadoPago	Varchar	50	Estado del pago, puede ser pendiente, completado, fallido, etc.
MetodoPago	Varchar	50	Método utilizado para el pago, como Webpay, Efectivo o Transferencia.
ReferenciaTransaccion	Varchar	100	Número de referencia o identificación de la transacción proporcionado por la pasarela de pago.

Fuente: Elaboración propia.

Tabla 3-4 Tabla DetalleReserva

NOMBRE	DetalleReserva		
DESCRIPCIÓN	Tabla que registra los cambios de estado asociados a una reserva. Cada entrada en esta tabla corresponde a un cambio en el estado de una reserva, y mantiene un historial de transacciones de estado con la fecha en que ocurrió cada cambio.		
CLAVE PRINCIPAL	IdDetalleReserva		
CLAVES FORÁNEAS	IdEstadoReserva -> EstadoReserva IdReserva -> Reserva		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdDetalleReserva	Int	N/A	Identificador único para cada registro de cambio de estado de una reserva.
IdEstadoReserva	Int	N/A	Clave foránea que hace referencia a la tabla EstadoReserva, indicando el nuevo estado de la reserva. Por ejemplo: confirmada, cancelada, pendiente.

IdReserva	Int	N/A	Clave foránea que hace referencia a la tabla Reserva, asociando el cambio de estado con una reserva específica.
FechaCambioEstado	Datetime	N/A	Fecha y hora en que ocurrió el cambio de estado de la reserva.

Fuente: Elaboración propia.

Tabla 3-5 Tabla EstadoReserva

NOMBRE	EstadoReserva		
DESCRIPCIÓN	Tabla que define los posibles estados de una reserva. Cada estado representa una fase del ciclo de vida de una reserva, como pendiente, confirmada o cancelada, y se utiliza para gestionar el flujo de la reserva en el sistema.		
CLAVE PRINCIPAL	IdEstadoReserva		
CLAVES FORÁNEAS	N/A		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdEstadoReserva	Int	N/A	Identificador único para cada estado de reserva en el sistema.
Nombre	Varchar	50	Nombre que describe el estado de la reserva, como Pendiente, Confirmada, Cancelada, etc.

Fuente: Elaboración propia.

Tabla 3-6 Tabla TransferenciaReserva

NOMBRE	TransferenciaReserva
DESCRIPCIÓN	Tabla que registra las transferencias de reservas entre usuarios. Cada entrada de esta tabla refleja una acción en la que una reserva es transferida de un usuario a otro, incluyendo la fecha en la que se realizó dicha transferencia.
CLAVE PRINCIPAL	IdTransferencia
CLAVES FORÁNEAS	N/A
ESTRUCTURA DE REGISTRO	

CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdTransferencia	Int	N/A	Identificador único para cada transferencia de reserva.
FechaTransferencia	Datetime	N/A	Fecha en la que se realizó la transferencia de reserva

Fuente: Elaboración propia.

Tabla 3-7 Tabla Reserva

NOMBRE	Reserva		
DESCRIPCIÓN	Tabla que almacena la información relacionada con las reservas de estacionamiento. Cada registro en esta tabla representa una reserva realizada por un usuario, incluyendo detalles como la fecha de inicio y fin de la reserva, el estado del pago, vehículo asociado y comentarios adicionales.		
CLAVE PRINCIPAL	IdReserva		
CLAVES FORÁNEAS	IdEstacionamiento -> Estacionamiento IdPago -> Pago IdTrabajador -> Trabajador IdValidacionOCR -> ValidacionOCR, IdTransferencia.		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdReserva	Int	N/A	Identificador único para cada reserva.
FechaCreacion	Datetime	N/A	Fecha en la que se creó la reserva.
EstadoPago	Varchar	256	Estado del pago de la reserva. Por ejemplo: pendiente, completado, cancelado.
FechaInicioReserva	Datetime	N/A	Fecha y hora de inicio de la reserva del estacionamiento.
FechaFinReserva	Datetime	N/A	Fecha y hora de fin de la reserva del estacionamiento.
NombreVisitante	Varchar	100	Nombre del visitante si la reserva es para un externo.

TipoReserva	Varchar	50	Tipo de la reserva. Por ejemplo, Común, Preferencial.
PatenteVehiculo	Varchar	20	Numero de la matrícula del vehículo asociado a la reserva.
Comentario	Varchar	256	Comentarios adicionales que pueda tener el usuario sobre la reserva. Por ejemplo, instrucciones especiales.
IdEstacionamiento	Int	N/A	Clave foránea que referencia a la tabla Estacionamiento, indicando el estacionamiento reservado.
IdPago	Int	N/A	Clave foránea que hace referencia a la tabla Pago. Asociando el pago con la reserva.
IdTrabajador	Int	N/A	Clave foránea que hace referencia a la tabla Trabajador. Asociando la reserva al trabajador.
IdTransferencia	Int	N/A	Clave foránea que hace referencia a la transferencia de reserva (si aplica).

Fuente: Elaboración propia.

Tabla 3-8 Tabla Notificación

NOMBRE	Notificación		
DESCRIPCIÓN	Tabla que almacena las notificaciones enviadas a los usuarios del sistema. Cada registro representa una notificación específica relacionada con eventos del sistema, como reservas, penalizaciones o acciones relacionadas con los trabajadores.		
CLAVE PRINCIPAL	IdNotificacion		
CLAVES FORÁNEAS	IdReserva -> Reserva IdPenalizacion -> Penalización IdTrabajador -> Trabajador		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN

IdNotificacion	Int	N/A	Identificador único por cada notificación.
TipoNotificacion	Varchar	50	Tipo de notificación, como reserva, penalización, trabajador, recordatorio, aviso.
FechaEnvio	Datetime	N/A	Fecha y hora en la que se envió la notificación.
Estado	Varchar	50	Estado de la notificación, como pendiente, enviada, fallida.
MedioEnvio	Varchar	50	Medio utilizado para enviar la notificación, como correo electrónico, notificación push.
Mensaje	Varchar	500	Contenido del mensaje enviado en la notificación.
IdReserva	Int	N/A	Clave foránea que hace referencia a la tabla Reserva. Asociando la reserva a la notificación (si aplica).
IdPenalizacion	Int	N/A	Clave foránea que hace referencia a la tabla Penalización. Asociando la penalización a la notificación (si aplica).
IdTrabajador	Int	N/A	Clave foránea que hace referencia a la tabla Trabajador. Asociando al trabajador con la notificación (si aplica).

Fuente: Elaboración propia.

Tabla 3-9 Tabla ValidacionOCR

NOMBRE	ValidacionOCR
DESCRIPCIÓN	Tabla que almacena los registros de validación de acceso al estacionamiento mediante el sistema OCR. Cada registro corresponde a la validación de la matrícula de un vehículo al momento del acceso, registrando el resultado, la patente leída y la fecha y hora de la validación.
CLAVE PRINCIPAL	IdValidacionOCR
CLAVES FORÁNEAS	IdReserva -> Reserva

ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdValidacionOCR	Int	N/A	Identificador único para cada registro de validación de OCR.
ResultadoValidacion	Varchar	50	Resultado de la validación, como exitoso, o fallido.
PatenteLeida	Varchar	20	Matrícula del vehículo que fue leída por el sistema OCR.
FechaHora	Datetime	N/A	Fecha y hora en que se realizó la validación del vehículo.
IdReserva	Int	N/A	Referencia a la reserva validada (puede ser nulo si fallo o no existe reserva).

Fuente: Elaboración propia.

Tabla 3-10 Tabla Reporte

NOMBRE	Reporte		
DESCRIPCIÓN	Tabla que almacena los reportes generados por el sistema, contienen métricas relacionadas con las reservas y penalizaciones.		
CLAVE PRINCIPAL	IdReporte		
CLAVES FORÁNEAS	IdTrabajador -> Trabajador		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdReporte	Int	N/A	Identificador único de cada reporte generado.
TipoReporte	Varchar	50	Tipo de reporte, como Uso de estacionamiento, Ingresos, Penalizaciones.
Descripción	Varchar	256	Descripción del contenido del reporte o propósito
Formato	Varchar	10	Formato en el que se genera el reporte, como PDF, CSV, Excel.

FechaInicio	Datetime	N/A	Fecha de inicio del periodo al que se refiere el reporte, como inicio del mes.
FechaFin	Datetime	N/A	Fecha de finalización del periodo al que se refiere el reporte.
IdTrabajador	Int	N/A	Clave foránea que hacer referencia a la tabla Trabajador. Especifica quien genero el reporte.

Fuente: Elaboración propia.

Tabla 3-11 Tabla ListaEspera

NOMBRE	ListaEspera		
DESCRIPCIÓN	Tabla que almacena las solicitudes de usuarios para ocupar un espacio de estacionamiento cuando no hay disponibilidad inmediata. Cada registro representa la entrada en la lista de espera, incluyendo información sobre el estado de la solicitud, su prioridad y la fecha en que fue solicitada.		
CLAVE PRINCIPAL	IdListaEspera		
CLAVES FORÁNEAS	N/A		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdListaEspera	Int	N/A	Identificador único para cada solicitud en la lista de espera.
Estado	Varchar	50	Estado de la solicitud en la lista de espera, como pendiente, asignada, cancelada.
Prioridad	Varchar	10	Prioridad de la solicitud en la lista de espera, que puede influir en el orden de asignación de un estacionamiento, como alta, media, baja.
FechaReserva	Datetime	N/A	Fecha para la cual el usuario solicita el estacionamiento.
FechaSolicitud	Datetime	N/A	Fecha en que el usuario solicitó ser añadido a la lista de espera.

Fuente: Elaboración propia.

Tabla 3-12 Tabla Trabajador

NOMBRE	Trabajador		
DESCRIPCIÓN	Tabla que almacena la información relacionada con los trabajadores del sistema, incluyendo su rol, clave de acceso, correo, y las relaciones con otras entidades como la persona asociada y la lista de espera en caso de que el trabajador este esperando un espacio de estacionamiento.		
CLAVE PRINCIPAL	IdTrabajador		
CLAVES FORÁNEAS	IdListaEspera -> ListaEspera IdPersona -> Persona		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdTrabajador	Int	N/A	Identificador único para cada trabajador.
Clave	Varchar	256	Clave de acceso del trabajador al sistema.
Rol	Varchar	50	Rol asignado al trabajador, como administrador, colaborador, supervisor, gerente, etc.
Correo	Varchar	100	Correo electrónico del trabajador.
IdPersona	Int	N/A	Clave foránea que hacer referencia a la tabla Persona vinculado los datos personales del trabajador.
IdListaEspera	Int	N/A	Clave foránea que hacer referencia a la tabla ListaEspera, en caso de que el trabajador este esperando un estacionamiento.

Fuente: Elaboración propia.

Tabla 3-13 Tabla Penalización

NOMBRE	Penalización
DESCRIPCIÓN	Tabla que almacena las penalizaciones aplicadas a los trabajadores, registrando información sobre la duración,

	razón y estado de cada penalización, así como su relación con el trabajador afectado.		
CLAVE PRINCIPAL	IdPenalizacion		
CLAVES FORÁNEAS	IdTrabajador -> Trabajador		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdPenalizacion	Int	N/A	Identificador único para cada penalización
FechaInicio	Datetime	N/A	Fecha en la que comienza la penalización
Razón	Varchar	256	Motivo por el cual se aplica la penalización, como retraso, violación de las reglas, etc.
FechaTermino	Datetime	N/A	Fecha en la que finaliza la penalización.
Días	Int	N/A	Número de días que dura la penalización.
Estado	Varchar	256	Estado de la penalización, como activa, terminada, suspendida.
IdTrabajador	Int	N/A	Clave foránea que hace referencia al trabajador afectado por la penalización.

Fuente: Elaboración propia.

Tabla 3-14 Tabla Persona

NOMBRE	Persona		
DESCRIPCIÓN	Tabla que almacena la información básica de las personas registradas en el sistema como su nombre, su identificación, junto con un indicador de si son personas que pueden usar los estacionamientos preferenciales.		
CLAVE PRINCIPAL	IdPersona		
CLAVES FORÁNEAS	N/A		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdPersona	Int	N/A	Identificador único para cada persona.

Preferencial	Boolean	N/A	Identificador que determina si la persona puede usar un estacionamiento preferencial, donde 1 es igual a true y 0 es igual a false.
Nombre	Varchar	100	Nombre completo de la persona.
Identificador	Varchar	20	Identificación única de la persona, como el RUT.

Fuente: Elaboración propia.

Tabla 3-15 Tabla DetalleTrabajador

NOMBRE	DetalleTrabajador		
DESCRIPCIÓN	Tabla que registra los cambios de estado asociados a los trabajadores. Cada registro representa un cambio en el estado de un trabajador, junto con la fecha en la que ocurrió el cambio.		
CLAVE PRINCIPAL	IdDetalleTrabajador		
CLAVES FORÁNEAS	IdTrabajador -> Trabajador IdEstadoTrabajador -> EstadoTrabajador		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdDetalleTrabajador	Int	N/A	Identificador único para cada registro de cambios de estado del trabajador.
FechaCambioEstado	Datetime	N/A	Fecha en la que ocurrió el cambio de estado del trabajador.
IdEstadoTrabajador	Int	N/A	Clave foránea que hacer referencia a la tabla EstadoTrabajador, indicando un nuevo estado del trabajador.
IdTrabajador	Int	N/A	Clave foránea que hace referencia a la tabla Trabajador. Indicando el trabajador al que se le aplico el cambio de estado.

Fuente: Elaboración propia.

Tabla 3-16 Tabla EstadoTrabajador

NOMBRE	EstadoTrabajador		
DESCRIPCIÓN	Tabla que define los posibles estados de los trabajadores en el sistema, como activo, suspendido, etc.		
CLAVE PRINCIPAL	IdEstadoTrabajador		
CLAVES FORÁNEAS	N/A		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdEstadoTrabajador	Int	N/A	Identificador único para cada estado del trabajador.
Nombre	Varchar	50	Nombre del estado, como Activo, Suspendido, etc.

Fuente: Elaboración propia.

Tabla 3-17 EstadoPenalizacion

NOMBRE	EstadoPenalizacion		
DESCRIPCIÓN	Tabla que define los posibles estados de las penalizaciones en el sistema, como activa, finalizada o suspendida. Esta tabla se utiliza para gestionar el estado de cada penalización aplicada a un trabajador.		
CLAVE PRINCIPAL	IdEstadoPenalizacion		
CLAVES FORÁNEAS	N/A		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdEstadoPenalizacion	Int	N/A	Identificador único para cada estado de penalización.
Nombre	Varchar	50	Nombre del estado de la penalización, como activa, finalizada, suspendida, etc.

Fuente: Elaboración propia.

Tabla 3-18 Tabla DetallePenalizacion

NOMBRE	DetallePenalizacion
--------	---------------------

DESCRIPCIÓN	Tabla que registra los cambios de estado asociados a una penalización. Cada registro refleja un cambio en el estado de una penalización, junto con su fecha de registro.		
CLAVE PRINCIPAL	IdDetallePenalizacion		
CLAVES FORÁNEAS	IdEstadoPenalizacion -> EstadoPenalizacion IdPenalizacion -> Penalizacion		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdDetallePenalizacion	Int	N/A	Identificador único para cada registro de cambio de estado de la penalización.
IdEstadoPenalizacion	Int	N/A	Clave foránea que hace referencia a la tabla EstadoPenalizacion, indicando el estado actual de la penalización.
IdPenalizacion	Int	N/A	Clave foránea que hace referencia a la tabla Penalizacion. Asociando el cambio de estado con la penalización a la que pertenece.
FechaCreacionRegistro	Datetime	N/A	Fecha en la que se registró el cambio de estado de la penalización.

Fuente: Elaboración propia.

Tabla 3-19 Tabla ReporteReserva

NOMBRE	ReporteReserva		
DESCRIPCIÓN	Tabla intermedia que resuelve la relación de muchos a muchos entre los reportes y las reservas. Permite consolidar múltiples reservas dentro de un único reporte generado para fines de análisis y trazabilidad.		
CLAVE PRINCIPAL	IdReporteReserva		
CLAVES FORÁNEAS	IdReporte -> Reporte IdReserva -> Reserva		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN

IdReporteReserva	Int	N/A	Identificador único para cada registro de reporte reserva.
IdReporte	Int	N/A	Clave foránea que hace referencia a la tabla Reporte.
IdReserva	Int	N/A	Clave foránea que hace referencia a la tabla Reporte.

Fuente: Elaboración propia.

Tabla 3-20 Tabla ReportePenalizacion

NOMBRE	ReportePenalizacion		
DESCRIPCIÓN	Tabla intermedia que resuelve la relación de muchos a muchos entre los reportes y las penalizaciones. Permite agrupar múltiples sanciones dentro de un mismo reporte para el control de gestión del administrador.		
CLAVE PRINCIPAL	IdReportePenalizacion		
CLAVES FORÁNEAS	IdReporte -> Reporte IdPenalizacion -> Penalizacion		
ESTRUCTURA DE REGISTRO			
CAMPO	TIPO	LARGO	DESCRIPCIÓN
IdReporteReserva	Int	N/A	Identificador único para cada registro de reporte reserva.
IdReporte	Int	N/A	Clave foránea que hace referencia a la tabla Reporte.
IdPenalizacion	Int	N/A	Clave foránea que hace referencia a la tabla Penalización.

Fuente: Elaboración propia.

3.3 Diagramas de secuencia extendido

En esta sección se muestran los diagramas de secuencia extendidos que se utilizan para modelar los aspectos dinámicos del sistema.

Observaciones:

- ✓ Para simplificar la representación y evitar redundancias, se utiliza la clase Usuario para englobar tanto a Administrador como a Trabajador, dejando claro que las operaciones y diagramas que involucran a Usuario pueden ser ejecutadas por cualquiera de estos roles según corresponda.
- ✓ Los métodos o mensajes utilizados en los diagramas de secuencia y colaboración corresponden a las operaciones principales asociadas a las entidades del dominio. Aquellos que dependen de la lógica de negocio o integración (servicios, DAOs, controladores) no necesariamente se reflejan en el modelo de clases principal.
- ✓ Para facilitar la lectura y comprensión de los diagramas, se representa explícitamente la interacción con servicios (por ejemplo, ServicioUsuario) y DAOs (por ejemplo, UsuarioDAO), aunque estos componentes no forman parte del modelo entidad-relación ni del modelo de clases del dominio, ya que su objetivo es evidenciar el flujo lógico y la arquitectura en capas.

3.3.1 Diagrama de Secuencia Extendido: CU-01 Crear usuario

El diagrama de secuencia extendido del CU-01: Crear usuario representa la secuencia de interacciones necesarias para registrar un nuevo usuario en el sistema, destacando la comunicación entre cada capa involucrada.

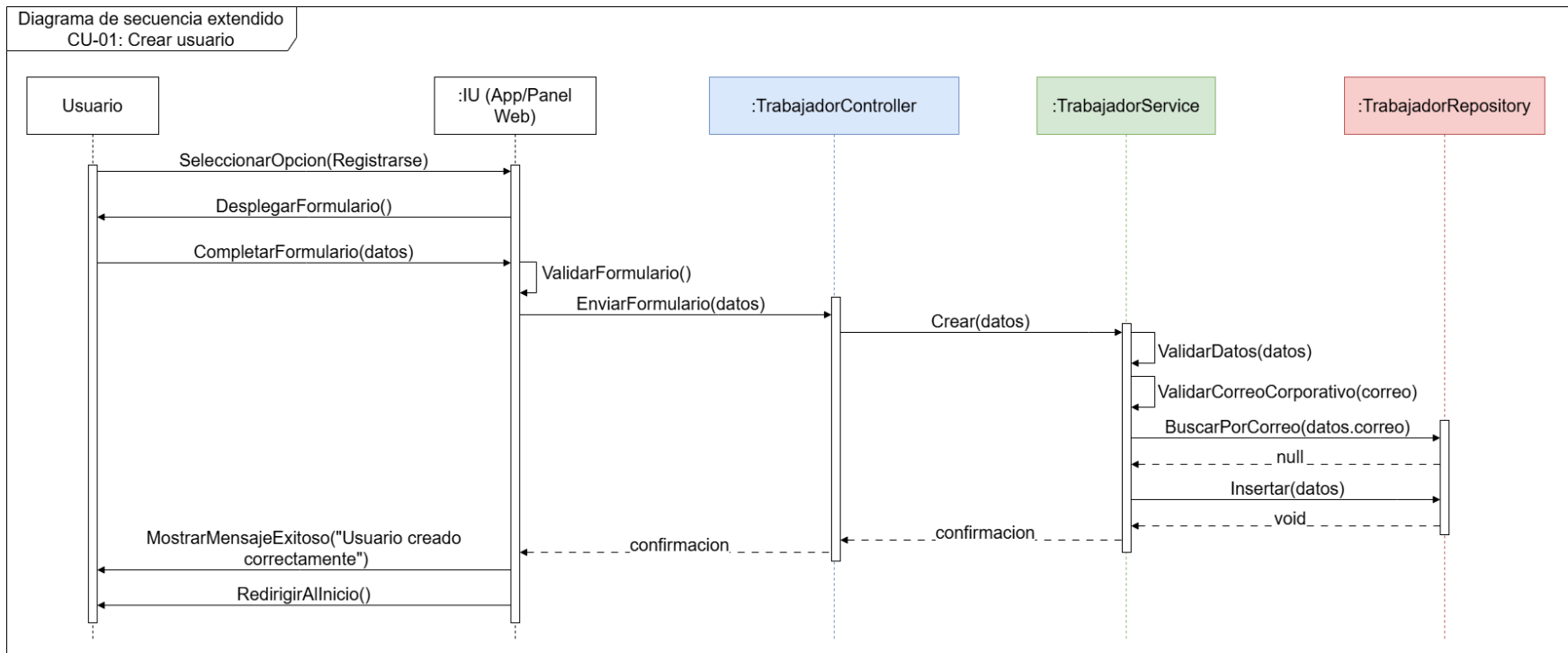


Figura 3-5 Diagrama de secuencia extendido CU-01 Crear Usuario

Fuente: Elaboración propia.

3.3.2 Diagrama de Secuencia Extendido: CU-02 Mantener usuario

El diagrama de secuencia extendido del CU-02: Mantener usuario muestra la interacción entre el usuario y el sistema para consultar y modificar información de usuarios, detallando el flujo de mensajes entre las distintas capas.

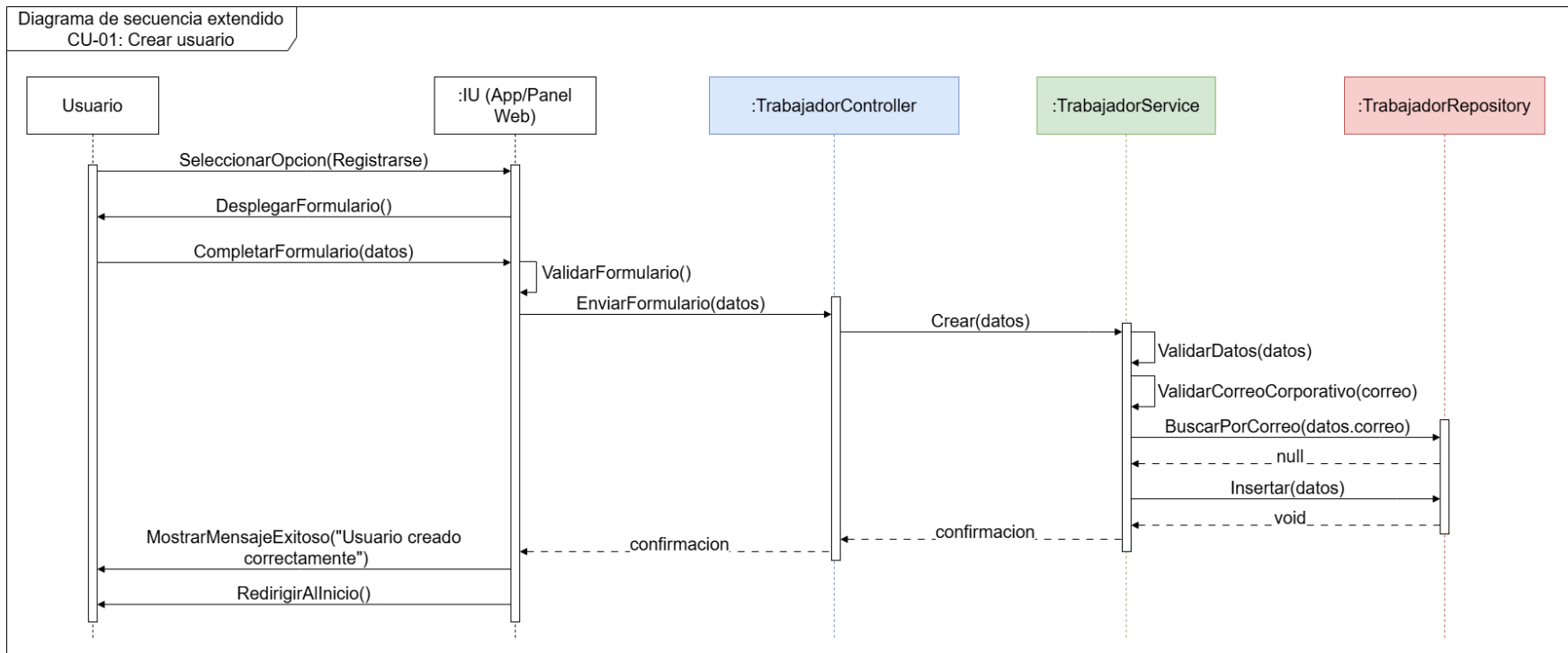


Figura 3-6 Diagrama de secuencia extendido CU-02 Mantener usuario

Fuente: Elaboración propia.

3.3.3 Diagrama de Secuencia Extendido: CU-03 Mantener reserva

El diagrama de secuencia extendido del CU-03: Mantener reserva describe cómo el usuario interactúa con el sistema para crear, modificar o cancelar reservas de estacionamiento, mostrando la secuencia de mensajes entre cada capa del sistema.

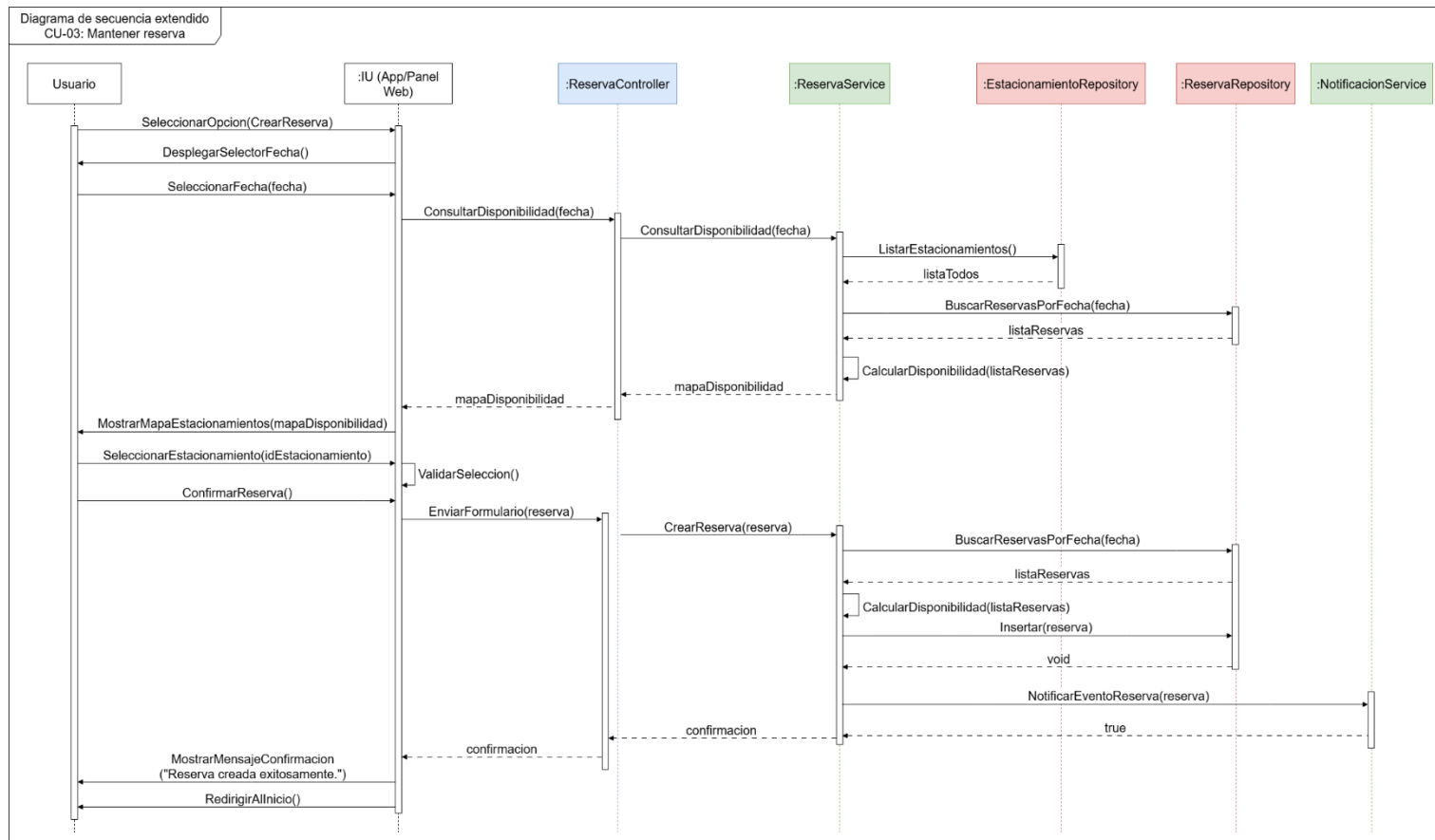


Figura 3-7 Diagrama de secuencia extendido CU-03 Mantener reserva

Fuente: Elaboración propia.

3.3.4 Diagrama de Secuencia Extendido: CU-03.5 Reasignar estacionamiento liberado

El diagrama de secuencia extendido del CU-03.5: Reasignar estacionamiento liberado muestra el proceso automático mediante el cual el sistema reasigna un estacionamiento cancelado al primer usuario elegible en lista de espera, actualizando los registros y notificando al colaborador correspondiente.

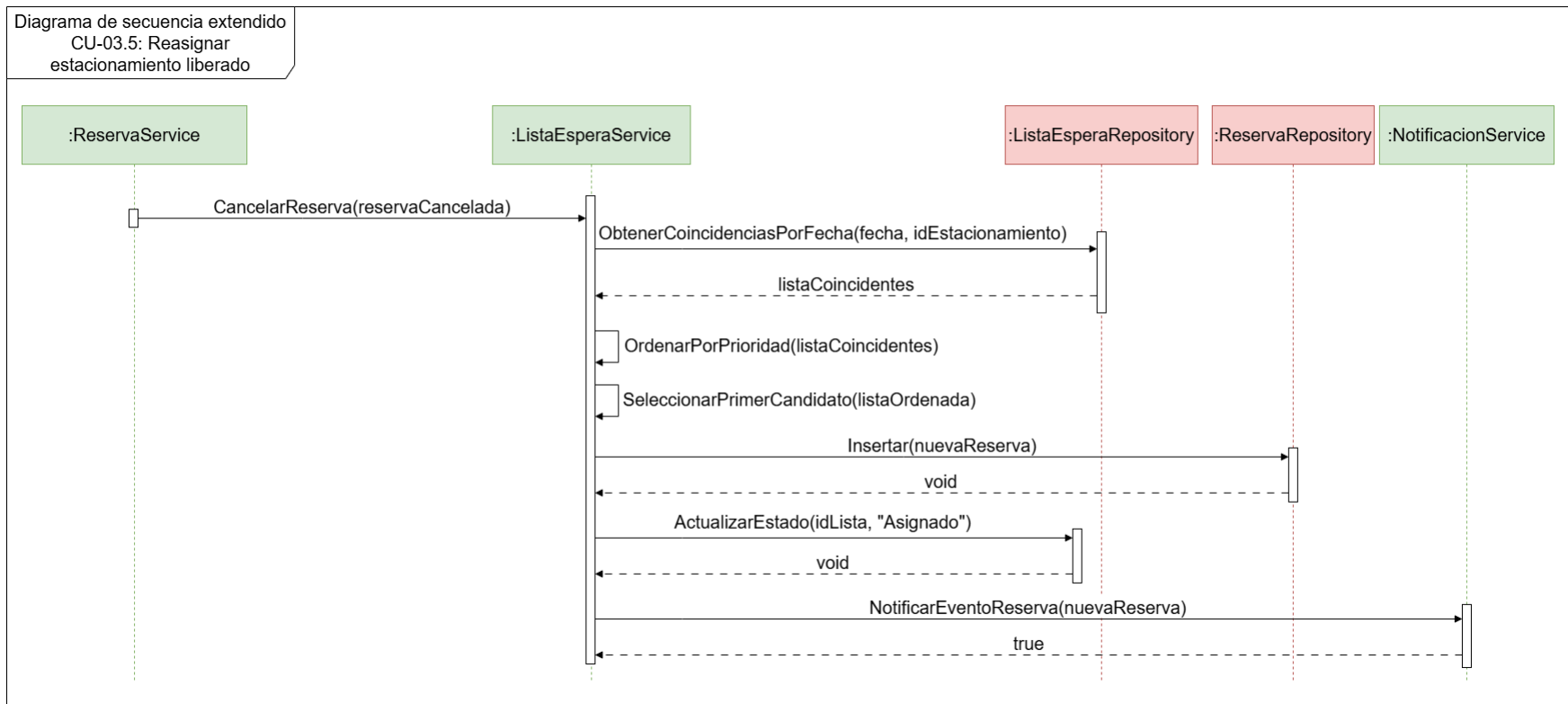


Figura 3-8 Diagrama de secuencia extendido CU-03.5 Reasignar estacionamiento liberado

Fuente: Elaboración propia.

3.3.5 Diagrama de Secuencia Extendido: CU-04 Realizar pago online por visita

El diagrama de secuencia extendido del CU-04: Realizar pago online por visita representa el proceso mediante el cual el usuario efectúa el pago electrónico de una reserva para un visitante, integrando la validación de datos, la interacción con la pasarela de pago y la confirmación automática de la reserva y el pago.

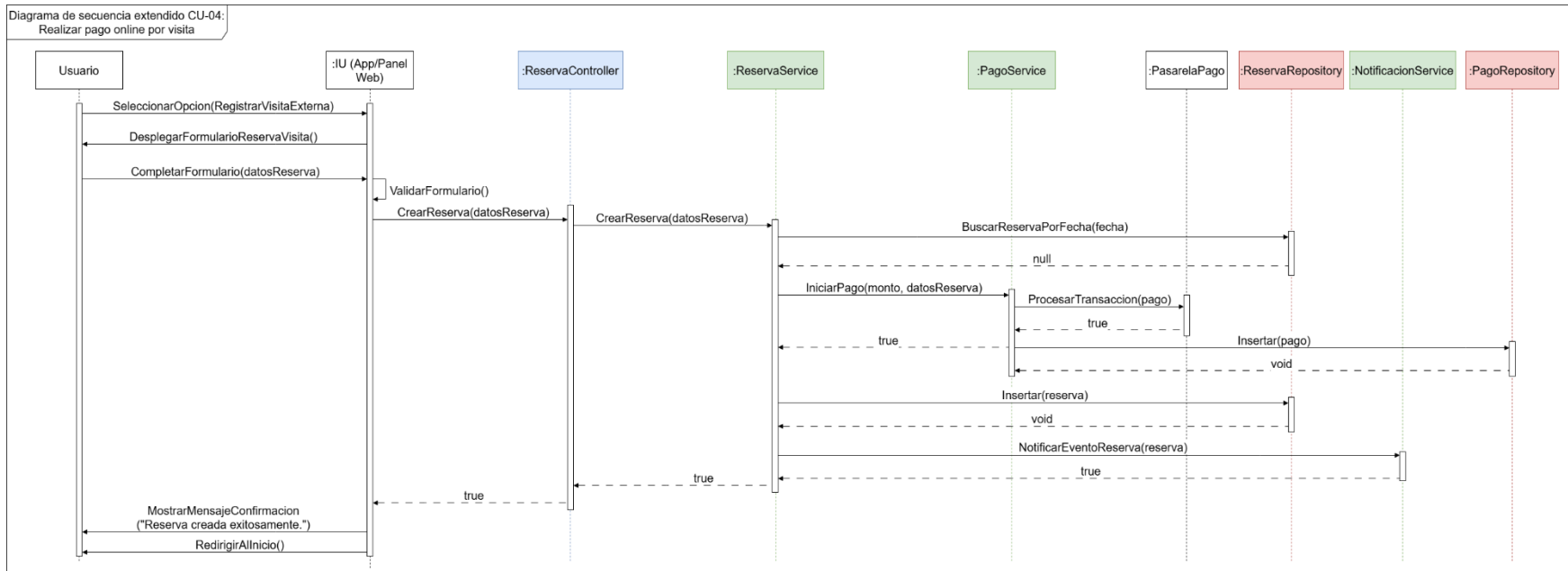


Figura 3-9 Diagrama de secuencia extendido CU-04 Realizar pago online por visita

Fuente: Elaboración propia.

3.3.6 Diagrama de Secuencia Extendido: CU-05 Validar acceso mediante OCR

El diagrama de secuencia extendido del CU-05: Validar acceso mediante OCR ilustra el proceso por el cual el sistema valida el ingreso de un vehículo a través de la lectura automática de la patente, notificando al usuario o administrador según el resultado de la validación.

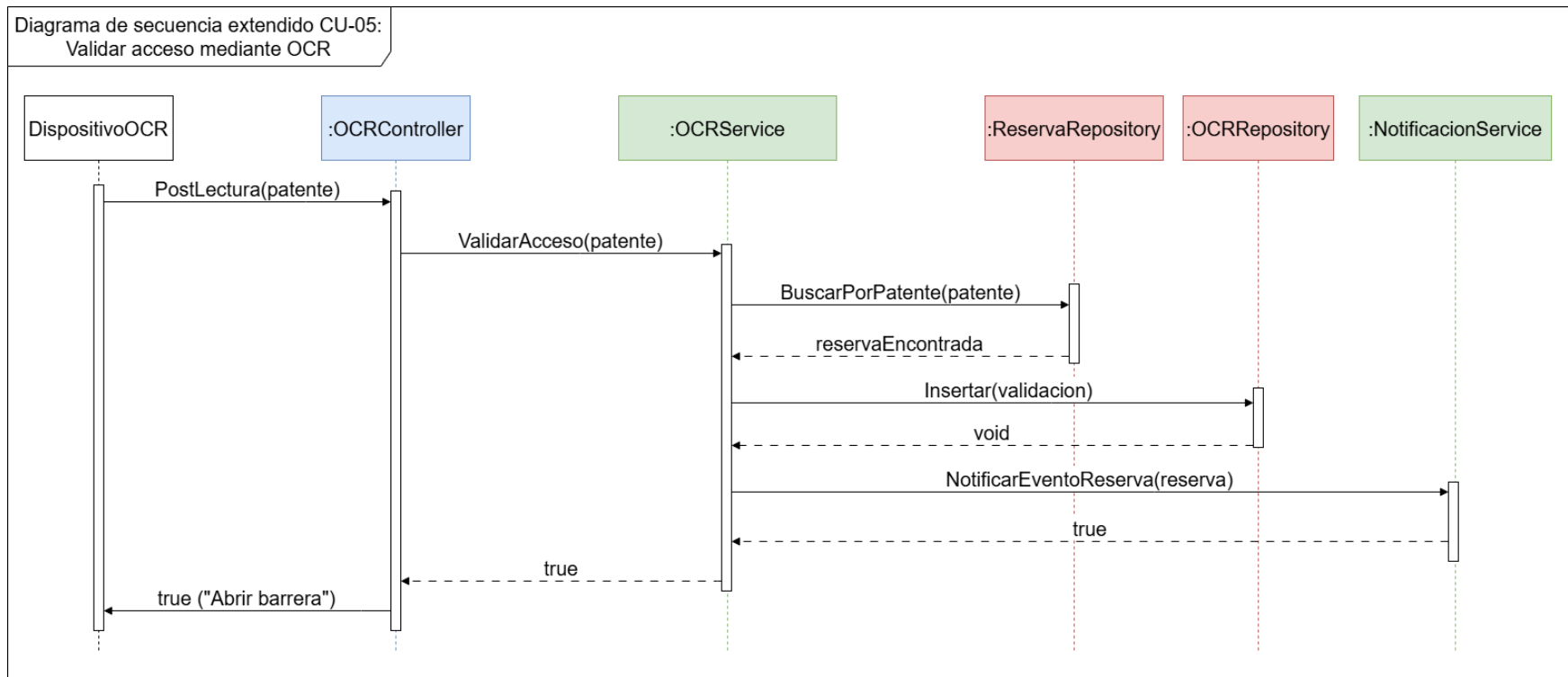


Figura 3-10 Diagrama de secuencia extendido CU-05 Validar acceso mediante OCR

Fuente: Elaboración propia.

3.3.7 Diagrama de Secuencia Extendido: CU-06 Recibir notificaciones

El diagrama de secuencia extendido del CU-06: Recibir notificaciones representa cómo el sistema detecta eventos relevantes y envía notificaciones automáticas a los usuarios, según sus preferencias de comunicación.

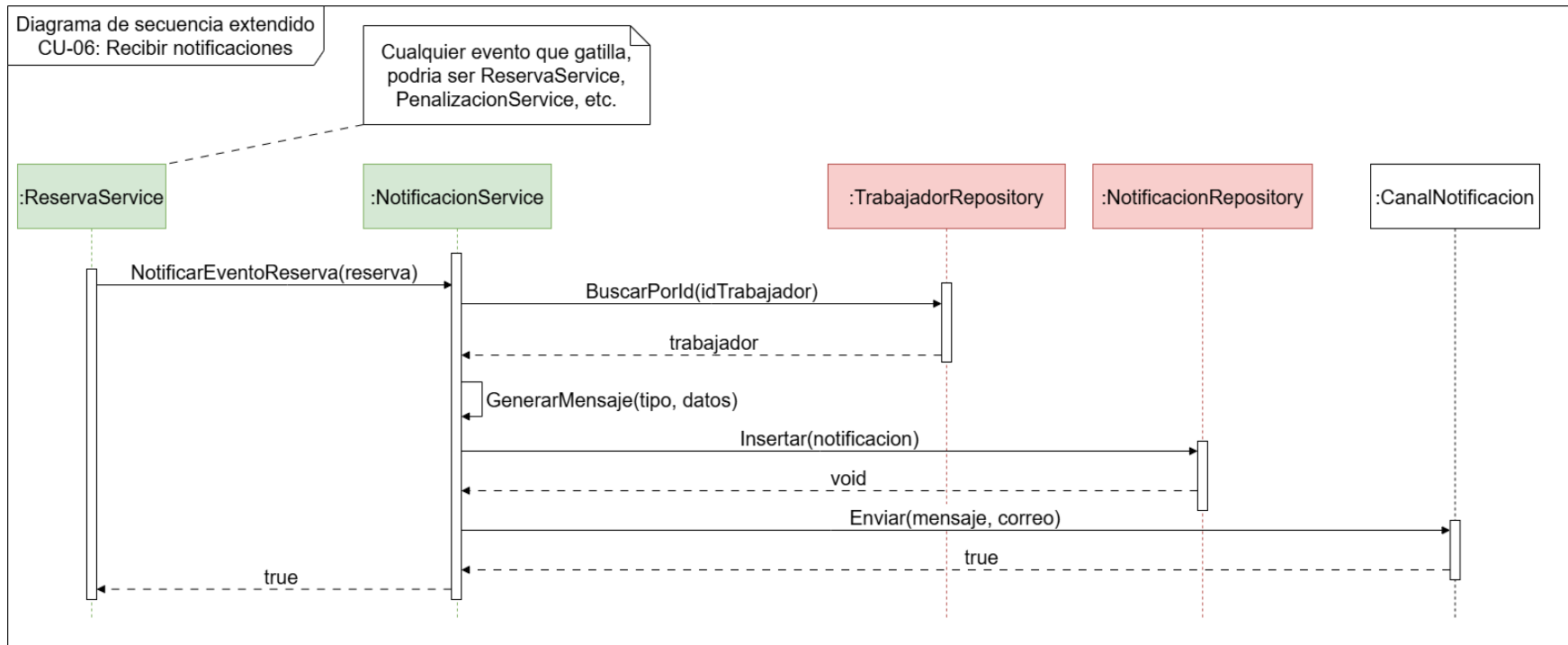


Figura 3-11 Diagrama de secuencia extendido CU-06 Recibir notificaciones

Fuente: Elaboración propia.

3.3.8 Diagrama de Secuencia Extendido: CU-07 Generar reporte

El diagrama de secuencia extendido del CU-07: Generar reporte muestra cómo el administrador solicita y descarga reportes del sistema, evidenciando la interacción entre cada capa y la validación de datos para la generación del informe.

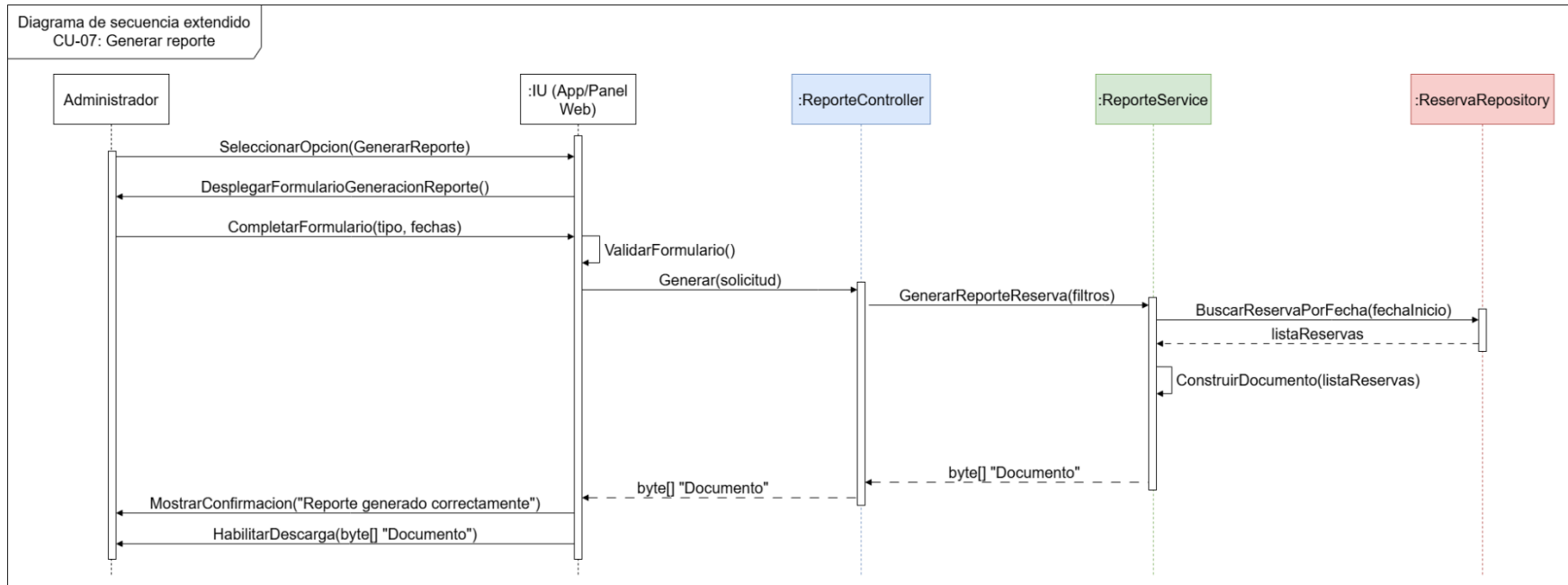


Figura 3-12 Diagrama de secuencia extendido CU-07 Generar reporte

Fuente: Elaboración propia.

3.3.9 Diagrama de Secuencia Extendido: CU-08 Inicio de sesión

El diagrama de secuencia extendido del CU-08: Inicio de sesión describe cómo el usuario accede al sistema ingresando sus credenciales, y muestra la validación de usuario, clave y rol antes de permitir el ingreso.

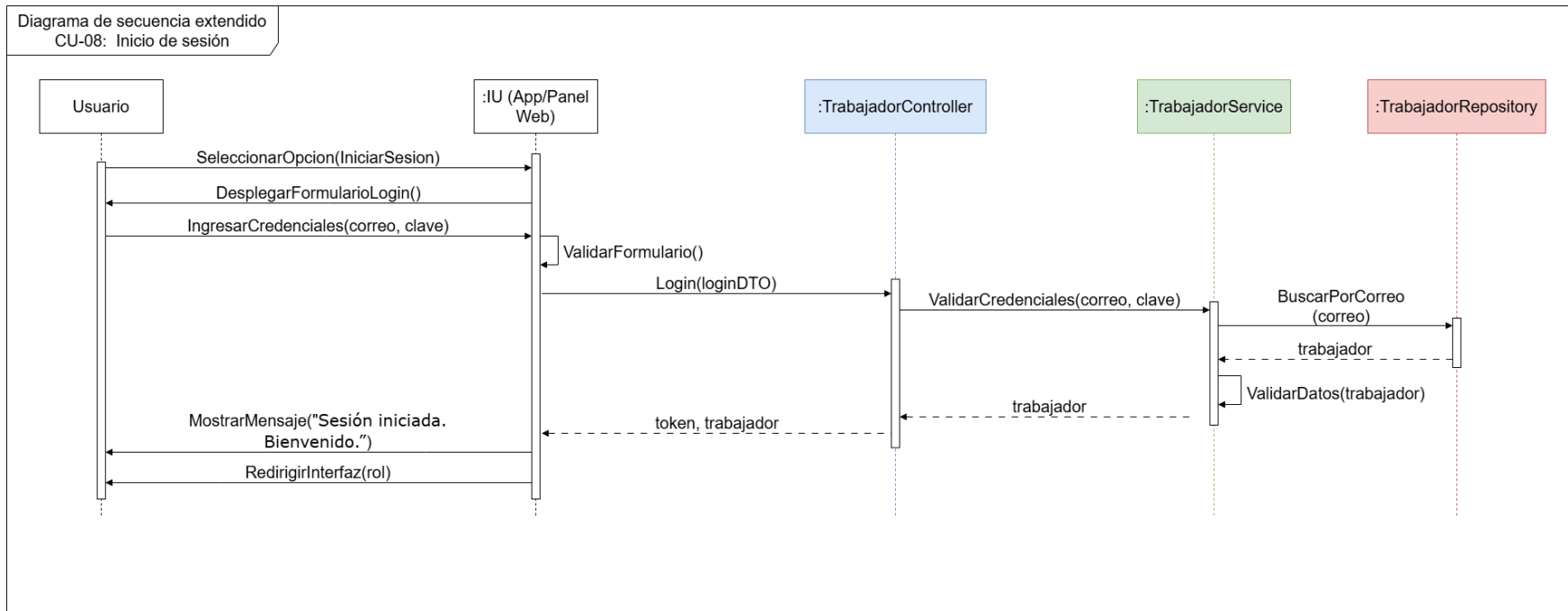


Figura 3-13 Diagrama de secuencia extendido CU-08 Inicio de sesión

Fuente: Elaboración propia.

3.3.10 Diagrama de Secuencia Extendido: CU-09 Mantener penalización

El diagrama de secuencia extendido del CU-09: Mantener penalización muestra cómo el administrador crea, modifica o elimina penalizaciones, detallando la interacción entre capas y el envío automático de notificaciones.

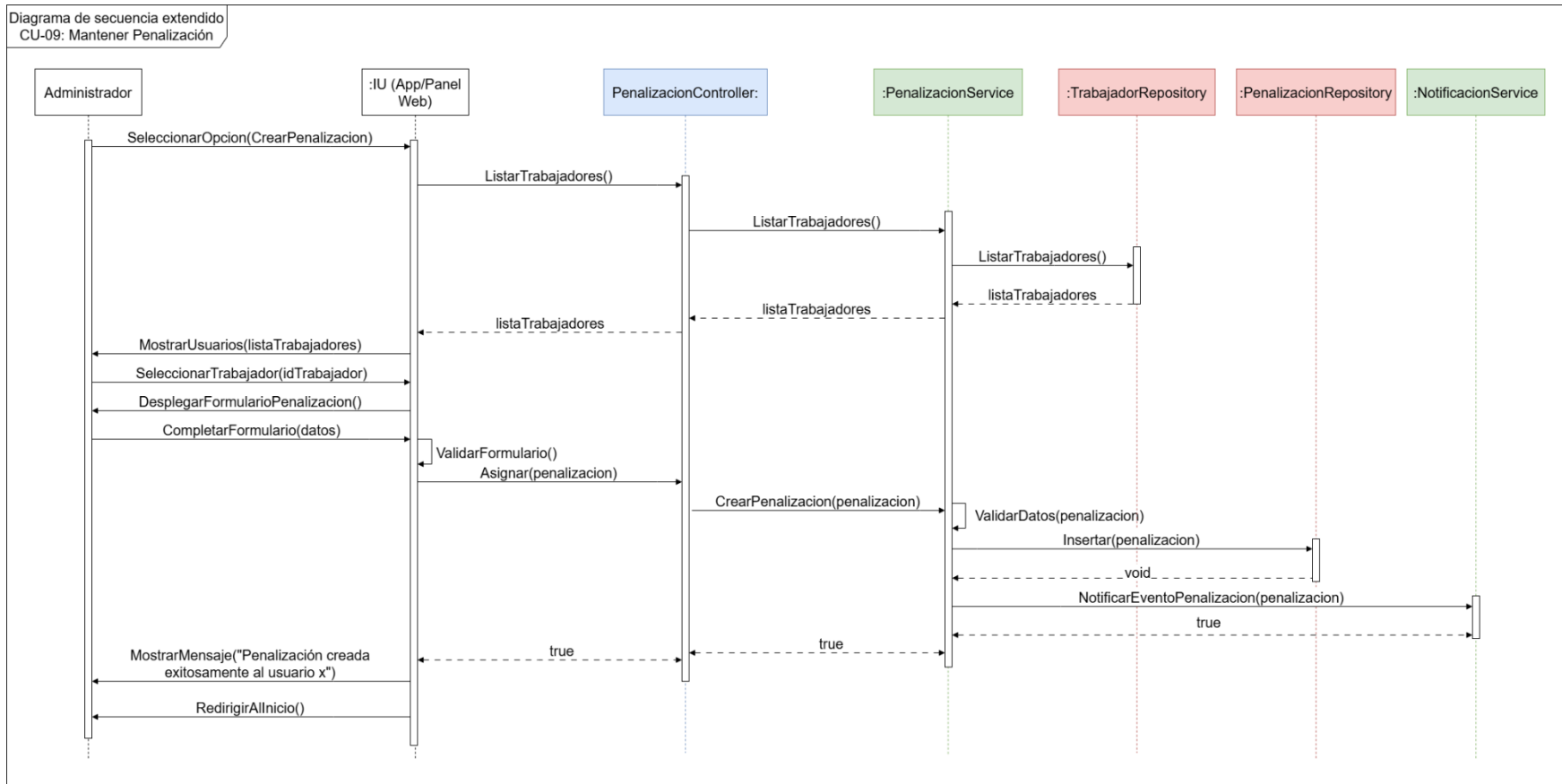


Figura 3-14 Diagrama de secuencia extendido CU-09 Mantener penalización

Fuente: Elaboración propia.

3.4 Diseño de Interfaz

Esta sección presenta el diseño de la interfaz del sistema, describiendo la estructura, funcionalidad y principales elementos visuales de las pantallas que permiten la interacción del usuario con la solución desarrollada. Se busca garantizar una experiencia intuitiva y eficiente, alineada con los requerimientos y flujos definidos en los casos de uso. Para la construcción de estos diseños de interfaz se utilizó la herramienta Figma, lo cual permitió validar la navegación y la disposición de los elementos previo a la etapa de codificación.

Para garantizar la calidad técnica y visual de la experiencia de usuario (UX), en cada diseño de interfaz se detallará el análisis de usabilidad basado en las heurísticas de Jakob Nielsen. Estas heurísticas corresponden a principios generales de diseño de interacción reconocidos como el estándar de la industria para evaluar y mejorar la usabilidad de interfaces informáticas.

Para proporcionar contexto sobre los criterios de evaluación utilizados en las siguientes pantallas, a continuación se describe brevemente cada una de las diez heurísticas de Nielsen:

1. **Visibilidad del estado del sistema:** El sistema debe mantener siempre informado al usuario sobre lo que está ocurriendo, proporcionando retroalimentación adecuada en un tiempo razonable (por ejemplo, indicadores de carga o mensajes de éxito).
2. **Relación entre el sistema y el mundo real:** El sistema debe utilizar el lenguaje de los usuarios, con palabras, frases y conceptos que le sean familiares, siguiendo convenciones del mundo real para que la información aparezca en un orden natural y lógico.
3. **Control y libertad del usuario:** El usuario elige funciones del sistema por error y necesita una "salida de emergencia" claramente marcada para abandonar el estado no deseado sin tener que pasar por un diálogo extendido (opciones de deshacer o rehacer).
4. **Consistencia y estándares:** Los usuarios no deberían tener que preguntarse si diferentes palabras, situaciones o acciones significan lo mismo. Se deben seguir las convenciones y estándares de la plataforma.
5. **Prevención y manejo de errores:** Más importante que un buen mensaje de error es un diseño cuidadoso que prevenga la ocurrencia de problemas. Se deben eliminar las condiciones propensas a errores o, en su defecto, verificarlas y presentar a los usuarios una opción de confirmación de que se comprometan a una acción.

6. **Reconocer antes que recordar:** Se debe minimizar la carga de memoria del usuario haciendo visibles los objetos, acciones y opciones. El usuario no debe tener que recordar información de una parte del dialogo a otra.
7. **Flexibilidad y eficiencia de uso:** La presencia de aceleradores (atajos no vistos por el usuario novato) permite la interacción para el usuario experto, de tal forma que el sistema pueda atender tanto a usuarios inexpertos como experimentados.
8. **Estética y diseño minimalista:** Los diálogos no deben contener información irrelevante o que rara vez se necesite. Cada unidad adicional de información compite con las unidades relevantes y disminuye la visibilidad relativa.
9. **Ayudar a los usuarios a reconocer, diagnostica y recuperarse de errores:** Los mensajes de error deben expresarse en lenguaje claro (sin códigos internos), indicando con precisión el problema y sugiriendo constructivamente una solución.
10. **Ayuda y documentación:** Aunque es mejor si el sistema puede ser utilizado sin documentación, puede ser necesario proveer ayuda. Cualquier información de este tipo debe ser fácil de buscar, estar enfocada en la tarea del usuario, listar todos los pasos concretos a llevar a cabo y no ser demasiado extensa.

Prototipo de interfaz

Para el desarrollo de la solución se elaboró un prototipo de interfaz que permite visualizar y validar la estructura, navegación y funcionalidad principal del sistema. El prototipo fue diseñado aplicando las heurísticas de usabilidad de Jakob Nielsen (detalladas previamente), haciendo especial énfasis en la prevención de errores, la visibilidad del estado del sistema y el diseño estético y minimalista.

Adicionalmente, se integraron principios fundamentales de la experiencia de usuario (UX), tales como la jerarquía visual para destacar los flujos críticos de operación (como los botones de reserva) y el enfoque de diseño responsivo, garantizando una interacción fluida y adaptable a través de múltiples dispositivos definidos previamente.

El prototipo contempla las pantallas clave del sistema, tales como inicio de sesión, menú principal, gestión de reservas, generación de reportes y administración de usuarios. Su construcción facilitó la detección temprana de posibles mejoras en los flujos de interacción, distribución de elementos y disposición de controles.

3.4.1 Diagrama de Menú o de Navegación

El diagrama de menús ilustra la estructura jerárquica de navegación del sistema, organizando las principales funciones en categorías y submenús para facilitar el acceso rápido e intuitivo a cada módulo de la aplicación. Cabe recalcar que el rol de Administrador vera este menú, en cambio el rol de Usuario solo vera Reservas y Penalizaciones (solo listara sus penalizaciones).

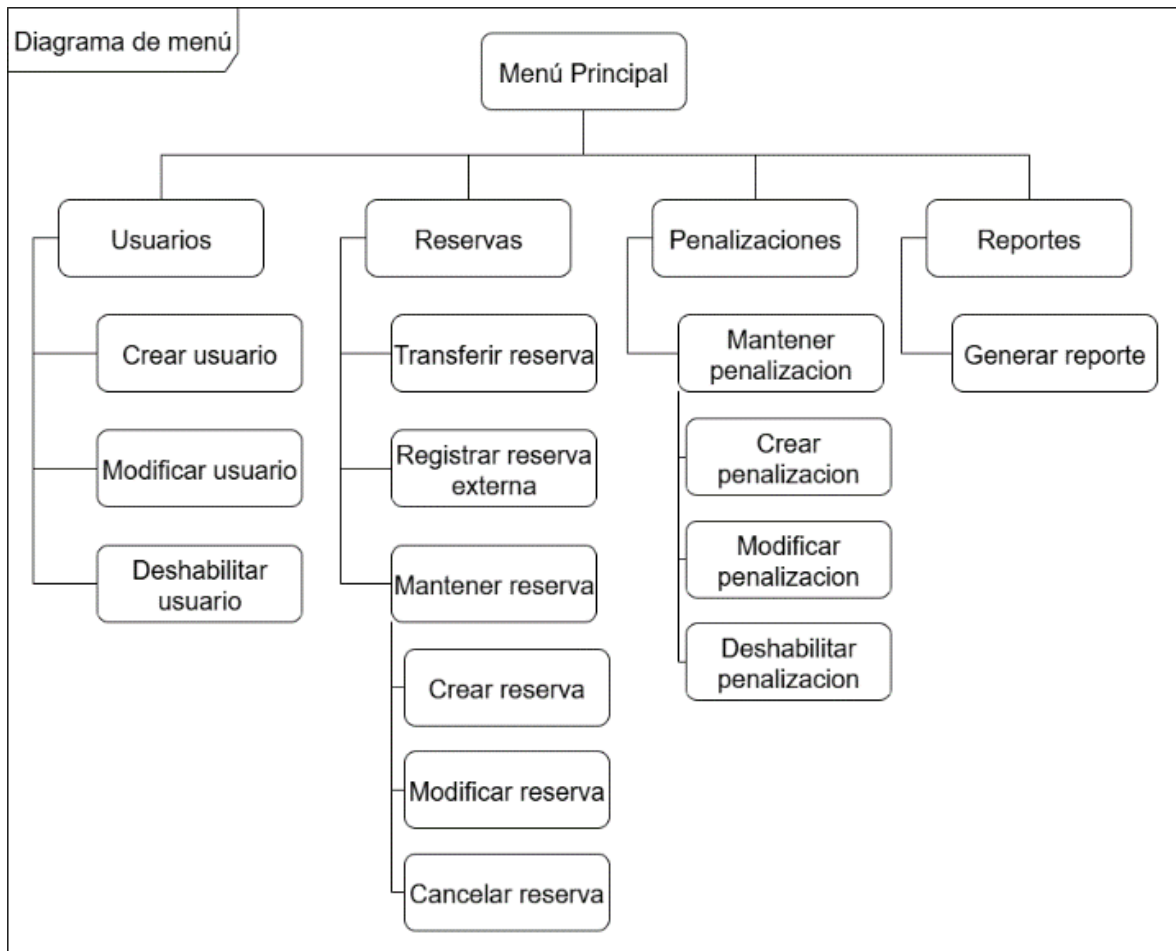


Figura 3-15 Diagrama de menú del sistema

Fuente: Elaboración propia.

3.4.2 Diseño de interfaz de la página principal

La pantalla de inicio presentada en la Figura 3-26 corresponde a la vista principal para el perfil de Administrador. Muestra un menú lateral con acceso a todas las funcionalidades del sistema, incluyendo usuarios, reservas, penalizaciones y reportes. En el área central se despliega un resumen visual con los principales indicadores de uso (reservas del día, reservas pendientes y penalizaciones activas), además de accesos rápidos a las tareas más frecuentes. El diseño prioriza la facilidad de navegación y visualización clara de la información relevante desde el primer ingreso. Cabe recalcar que el estilo del menú en el caso del rol Usuario es el mismo, la única diferencia es la cantidad de ítems que dispone, además que algunos menú ítem están colapsados para que encajen bien en el diseño como Mantener reservas.

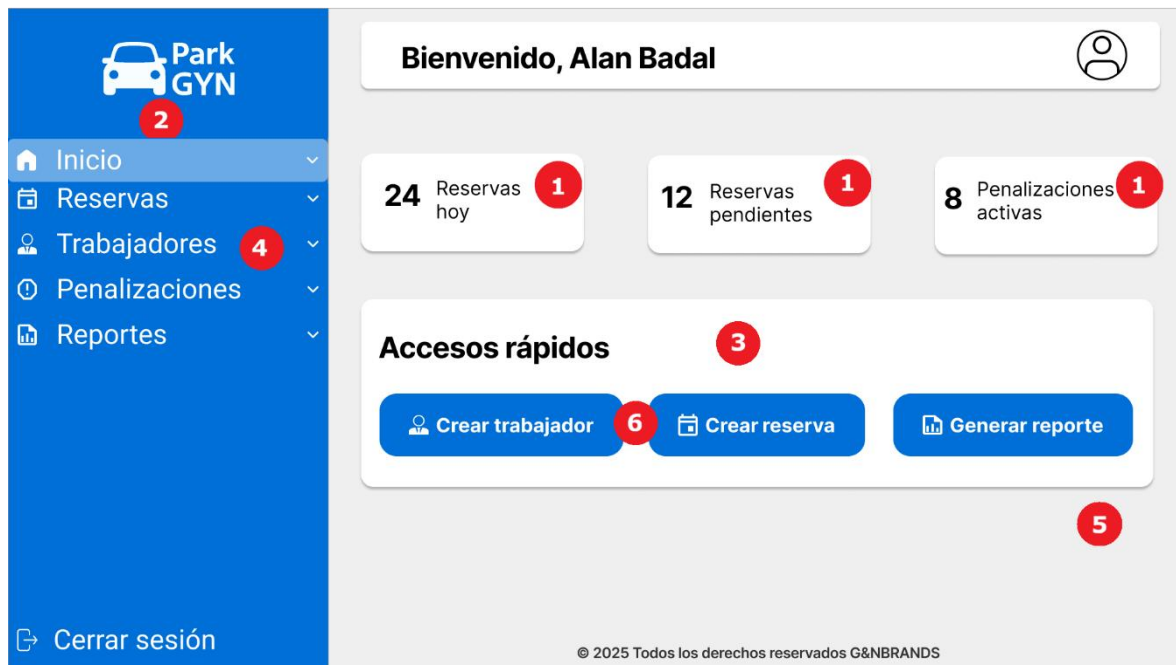


Figura 3-16 Diseño de interfaz de la página principal del sistema

Fuente: Elaboración propia.

Análisis de heurísticas:

1. **Visibilidad del estado del sistema:** Los indicadores de reservas y penalizaciones muestran información clave en tiempo real, permitiendo al administrador conocer el estado actual del sistema de un vistazo.
2. **Reconocer antes que recordar:** Las opciones del menú están claramente etiquetadas y acompañadas de iconos, facilitando la identificación de las funciones sin que el usuario deba recordar la estructura interna del sistema.

3. **Consistencia y estándares:** La disposición del menú, el uso de iconografía y los accesos rápidos son consistentes con interfaces web modernas, siguiendo patrones reconocidos.
4. **Control y libertad del usuario:** El menú permite al usuario acceder rápidamente a cualquier módulo y regresar al inicio en cualquier momento, manteniendo siempre el control de la navegación.
5. **Estética y diseño minimalista:** El diseño utiliza espacios en blanco, agrupación visual y tipografía clara para evitar sobrecargar la pantalla, enfocando la atención en la información más importante.
6. **Flexibilidad y eficiencia de uso:** Los accesos rápidos optimizan el flujo de trabajo para tareas frecuentes, reduciendo la cantidad de clics y mejorando la eficiencia operativa.

3.4.3 Diseño de Interfaz CU-09 Iniciar sesión

A continuación, se detalla el diseño de interfaz del CU-09 Iniciar sesión, desde la perspectiva del panel web, ya que la app móvil será un ajuste responsivo y contendrá los mismos elementos.

La interfaz de usuario de la Figura 3-27 muestra el formulario de inicio de sesión para acceder al sistema ParkGYN. Esta pantalla permite al usuario ingresar sus credenciales, activar la opción de recordar usuario y acceder a la recuperación de contraseña en caso necesario. Además, proporciona mensajes de error claros cuando los datos son incorrectos.

G&NBRANDS
GOOD FOOD, GOOD MOOD.

© 2025 Todos los derechos reservados G&NBRANDS

Figura 3-17 Diseño interfaz CU-09 Inicio de sesión

Fuente: Elaboración propia.

Análisis de heurísticas:

1. **Visibilidad del estado del sistema:** El sistema informa al usuario cuando las credenciales son incorrectas, mostrando un mensaje en color rojo debajo del botón de inicio de sesión.
2. **Consistencia y estándares:** Los campos de correo y contraseña, el checkbox "Recordar mi usuario", y el botón principal siguen convenciones comunes en sistemas web, permitiendo al usuario reconocer fácilmente el propósito de cada elemento.
3. **Prevención y manejo de errores:** Cuando los datos ingresados son incorrectos, la interfaz despliega inmediatamente un mensaje de error en rojo. Además, ofrece la opción de "¿Olvidaste tu contraseña?" para recuperación.
4. **Reconocer antes que recordar:** El formulario utiliza etiquetas claras en los campos, ayudando al usuario a identificar fácilmente que información debe ingresar.
5. **Estética y diseño minimalista:** La interfaz mantiene un diseño limpio, con solo los elementos necesarios, facilitando la concentración en la tarea principal.

3.4.4 Diseño de interfaz CU-03 Mantener reserva

La interfaz de usuario de la Figura 3-28 muestra la opción Crear reserva del CU-03 Mantener reserva, la cual permite al usuario seleccionar visualmente un estacionamiento disponible utilizando un esquema tipo "cine". Cada espacio está identificado por un código y color, e incluye iconos diferenciadores para espacios de uso común y preferencial. En la parte derecha se despliega una leyenda clara con la simbología y un área de ayuda que informa sobre la lista de espera. El flujo se completa mediante los botones de acción "Confirmar selección" y "Lista de espera", facilitando la toma de decisión y navegación.

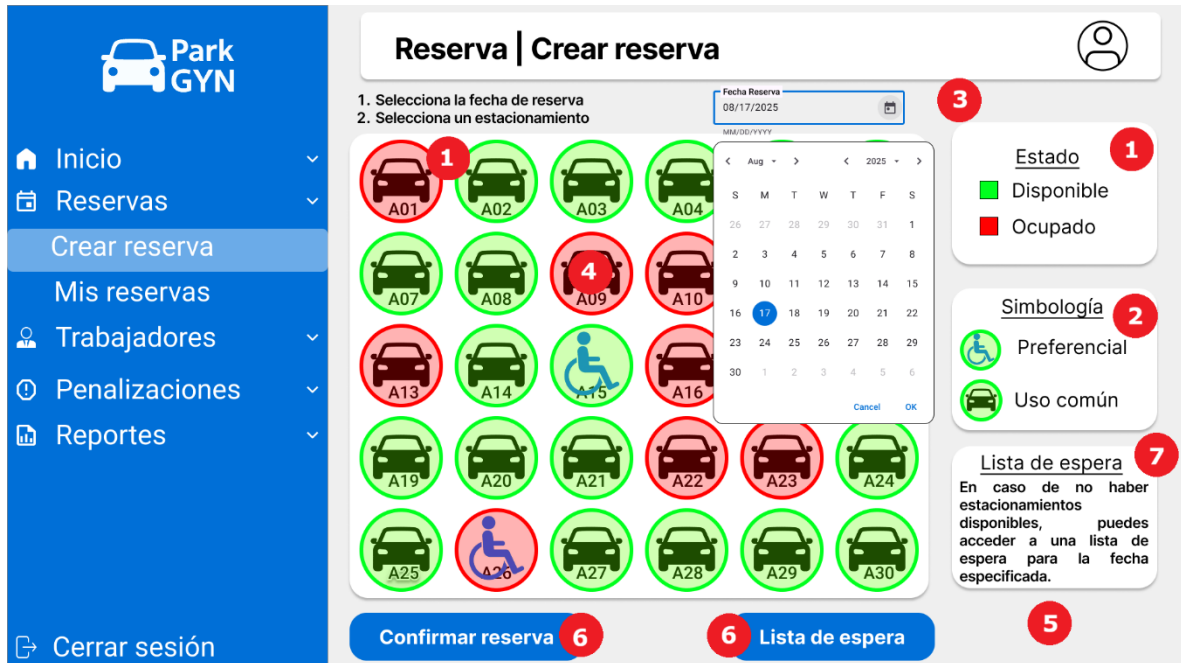


Figura 3-18 Diseño de interfaz CU-03 Mantener reserva

Fuente: Elaboración propia.

Análisis de heurísticas:

1. **Visibilidad del estado del sistema:** El usuario puede ver claramente que estacionamiento están disponibles u ocupados gracias a la codificación de colores y la leyenda explicativa.
2. **Reconocer antes que recordar:** El uso de iconos y colores, junto con la simbología lateral, permite que los usuarios identifiquen rápidamente los tipos de estacionamiento sin necesidad de memorizar significados.
3. **Consistencia y estándares:** La disposición de botones, leyendas y colores sigue patrones conocidos, facilitando el uso incluso para quienes acceden por primera vez.
4. **Prevención y manejo de errores:** Al intentar seleccionar un estacionamiento ocupado, el sistema previene la acción y permite al usuario acceder fácilmente a la lista de espera.
5. **Estética y diseño minimalista:** La interfaz presenta solo los elementos esenciales, utilizando espacios en blanco y agrupaciones lógicas para enfocar la atención en la sección principal.
6. **Flexibilidad y eficiencia de uso:** La inclusión de botones para confirmar selección y acceder a la lista de espera optimiza el flujo para distintos escenarios de uso.

7. **Ayuda y documentación:** La sección de “Lista de espera” entrega información contextual relevante para orientar al usuario ante la falta de disponibilidad.

Cabe destacar que el diseño de estos prototipos no fue un proceso aislado, sino que contó con la participación y asesoría constante de los usuarios finales (personal del CAR). A modo de validación empírica, se realizaron pruebas de percepción y usabilidad con los colaboradores, cuya retroalimentación fue crucial para ajustar el alcance, validar la disposición de los elementos y asegurar que los flujos de trabajo reflejaran fielmente las operaciones diarias, resolviendo así de manera efectiva los “dolores de cabeza” asociados a la gestión manual.

4 CONCLUSIONES

El desarrollo de este proyecto representó un desafío que involucró desde el análisis profundo de la problemática existente en el Centro de Apoyo Restaurant (CAR), hasta el diseño detallado de una solución tecnológica alineada a las necesidades reales de los usuarios. El ciclo de investigación, levantamiento de requerimientos y diseño arquitectónico se extendió a lo largo de aproximadamente 2 semestres (10 meses), logrando establecer bases sólidas para la futura construcción del sistema.

Durante el proceso fue necesario realizar investigación adicional, especialmente para comparar alternativas tecnológicas (software comercial, código abierto, y desarrollo a medida) y para aprender sobre integración con sistemas de OCR, manejo de notificaciones automáticas y diseño UX/UI multiplataforma. Este aprendizaje me ayudó en mi formación profesional y me permitió adquirir competencias en la selección de tecnologías apropiadas y mejores prácticas de arquitectura de software.

En relación con el cumplimiento de los objetivos específicos planteados, se logró analizar con éxito el proceso manual actual, formalizando los requerimientos críticos del negocio. Posteriormente, se diseñó una arquitectura de software robusta y se planificó estratégicamente el desarrollo bajo un enfoque incremental (MVP), definiendo las reglas de negocio para la gestión de reservas y reasignación de cupos. Finalmente, se establecieron los mecanismos de validación y prototipado, cumpliendo con el propósito de diseñar una herramienta que mejore la eficiencia y transparencia del proceso.

Conté con la asesoría constante tanto de usuarios finales (personal del CAR) como de profesores en la universidad. Sus aportes y retroalimentación fueron cruciales para definir funcionalidades prioritarias, validar diseños y ajustar el alcance del sistema para que sea factible. La participación de los usuarios también facilitó el diseño de flujos de

trabajo que reflejaran fielmente las operaciones diarias y resolvieran los principales "dolores de cabeza" de la gestión manual.

A pesar de los resultados positivos en esta fase de formulación, el proyecto presentó limitaciones inherentes a las condiciones del entorno operativo. Por un lado, la infraestructura física del recinto carece de dispositivos IoT (como cámaras o sensores de ocupación), lo que obligó a descartar una automatización total y orientar el diseño hacia flujos de validación por software. Por otro lado, la normativa legal vigente sobre protección de datos personales limitó el modelado de la base de datos, exigiendo un diseño más riguroso en los mecanismos de seguridad y encriptación para el almacenamiento de patentes vehiculares y credenciales.

Como trabajo futuro, el sistema tiene potencial de crecer, incorporando nuevas funcionalidades como integración con sistemas de control de acceso físicos, módulos avanzados de analítica de ocupación y métricas, reportes inteligentes, fortalecer los mecanismos de seguridad y respaldo de la información incluso la experiencia del usuario a través de una retroalimentación de estos mismos.

Mi formación en ingeniería de ejecución en software fue crucial para enfrentar este proyecto tanto en los ámbitos técnicos y organizacionales que surgieron. La experiencia adquirida en análisis de requerimientos, modelado de datos, diseño orientado a objetos y patrones de arquitectura permitió que el sistema fuera robusto, escalable y fácil de mantener.

En resumen, este proyecto me permitió aplicar de manera integral los conocimientos adquiridos durante mi formación profesional, además también fue una oportunidad para adquirir habilidades nuevas, fortalecer la gestión de proyectos y profundizar en el diseño de soluciones centradas en el usuario. El sistema marcara un avance significativo hacia la modernización de los procesos internos del CAR.

5 **BIBLIOGRAFÍA**

- [1] G. Brands, «GYN Brands,» Abril 2024. [En línea]. Available: <https://gnbrands.com/>.
- [2] ParkFacil, «ParkFacil - Gestion de estacionamientos en la nube,» Febrero 2025. [En línea]. Available: <https://www.parkfacil.cl/>.
- [3] L. Randula, Parking Management System, 2023.
- [4] G. I. Tamayo, «Gestion de proyectos de Sw,» Mayo 2025. [En línea].
- [5] B. M. A. Henriquez, «Diagramas UML,» Abril 2024. [En línea].
- [6] INACAP, «Diagramas UML,» Noviembre 2018. [En línea].
- [7] P. G. Barrera, «Diagrama de casos de uso y Actores,» Octubre 2021. [En línea].
- [8] P. G. Barrera, «Modelo Conceptual,» Mayo 2024. [En línea].
- [9] P. G. Barrera, «Casos de uso Narrativos,» Noviembre 2024. [En línea].
- [10] P. G. Barrera, «Diagrama de Secuencia,» Abril 2024. [En línea].
- [11] B. M. A. Henriquez, «Diseño Arquitectonico,» 2024. [En línea].
- [12] B. M. A. Henriquez, «Principios de Diseño,» 2024. [En línea].
- [13] B. M. A. Henriquez, «Modelo de Diseño,» 2024. [En línea].
- [14] U. F. S. M. USM, «Bases de datos Relacionales,» Julio 2024. [En línea].