



DOCUMENTO DE ANÁLISIS DEL SOFTWARE

GESCOMPH

Jesús David Fierro Rivera
Brayan Santiago Guerrero Mendez

Contenido

1. Introducción	3
1.1. Propósito	3
1.2. Alcance	3
1.3. Definiciones, acrónimos y abreviaturas	4
1.4. Referencias	4
1.5. Visión general del software	6
1.6. Objetivos	6
1.6.1. Objetivo General	7
1.6.2. Objetivos Específicos	7
2. Descripción General del Software	7
2.1. Perspectiva del producto	8
2.2. Funcionalidades generales	8
2.3. Tipos de usuarios y roles	8
2.4. Restricciones	9
2.5. Suposiciones y Dependencias	12
2.6. Historias de usuario	13
3. Arquitectura del sistema	23
3.1. Estilo arquitectónico general	23
3.2. Diagrama general	23
3.3. Componentes principales	24
3.4. Tecnologías utilizadas	24
3.5. Seguridad	25
4. Análisis de viabilidad	27
4.1. Viabilidad técnica	27
4.2 Viabilidad económica	27
4.3 Viabilidad operativa	27
4.4 Viabilidad Legal	27
5. Interfaz de usuario y prototipos	28

5.1 Lineamientos generales de diseño	28
5.2 Componentes comunes	28
5.3 Prototipos de pantalla	29
4. Plan de Mantenimiento	31
5. Plan de Trabajo	34
6. Diagramas	36
6.1. Diagrama de Flujo	36
6.2. Diagrama de Clases	36
6.3. Diagrama de Secuencia	36
6.4. Diagrama General de Procesos	36
6.5. Diagrama de Casos de Uso	36

1. Introducción

El presente documento constituye el análisis formal del sistema de información **GESCOMPH (Gestión Comercial Palermo Huila)**, una plataforma destinada a apoyar la administración de los arriendos de establecimientos comerciales ubicados en las plazas públicas del municipio de Palermo, Huila. Su propósito es describir y entender de manera estructurada los procesos, entidades, reglas de negocio, interacciones y escenarios que conforman el funcionamiento del sistema, con el fin de establecer una base sólida para su diseño y desarrollo.

1.1. Propósito

El propósito de este documento es analizar de forma detallada el funcionamiento, los procesos, las entidades del dominio y las reglas de negocio que conforman el sistema de información **GESCOMPH (Gestión Comercial Palermo Huila)**. Este análisis proporciona una comprensión estructurada del problema que se desea resolver y establece las bases conceptuales necesarias para el diseño, modelado e implementación del software.

El documento tiene como finalidad identificar los actores involucrados, los flujos operativos que intervienen en la gestión de arriendos, pagos, obligaciones mensuales, cálculos de mora, manejo de citas y disponibilidad de establecimientos, así como las relaciones entre estos componentes. A través de esta perspectiva analítica, se busca asegurar que la solución tecnológica a desarrollar responda de manera precisa a las necesidades administrativas, operativas y funcionales de la Alcaldía de Palermo.

Este propósito incluye clarificar el alcance conceptual del sistema, definir las reglas de negocio que lo gobiernan y establecer los lineamientos necesarios para que el desarrollo posterior del software sea coherente, consistente y alineado con los objetivos institucionales.

1.2. Alcance

El alcance de este documento comprende el análisis integral del dominio funcional de **GESCOMPH (Gestión Comercial Palermo Huila)**, incluyendo los procesos, actores, entidades y reglas de negocio involucradas en la administración de arriendos de establecimientos comerciales ubicados en las plazas públicas del municipio de Palermo.

Este análisis abarca los siguientes aspectos:

- **Gestión administrativa del arriendo**, incluyendo plazas, establecimientos y asignación de contratos.
- **Modelado de contratos de arrendamiento**, sus atributos, ciclo de vida y sus relaciones con arrendatarios y establecimientos.
- **Generación y control de obligaciones mensuales**, junto con su cálculo automático de mora basado en parámetros institucionales.

- **Procesos de pago electrónico**, integración con la pasarela Mercado Pago y validación mediante webhooks.
- **Flujos de consulta para arrendatarios**, incluyendo visualización de contratos, obligaciones, historial y estado financiero.
- **Procesos asociados a ciudadanos no autenticados**, tales como consulta de locales disponibles y solicitudes de citas.
- **Gestión de citas**, desde su solicitud hasta la validación administrativa.
- **Gestión de imágenes asociadas a establecimientos**, almacenamiento y reglas de manejo.
- **Control de acceso**, basado en usuarios, roles, permisos y formularios.
- **Reglas de negocio transversales**, tales como auditoría, trazabilidad, validación de datos y restricciones operativas.

Quedan fuera del alcance de este análisis:

- Integraciones con sistemas contables, presupuestales, jurídicos o tributarios externos.
- Procesos de firma digital avanzada o certificaciones electrónicas.
- Modelos financieros o contables externos al ámbito del arriendo municipal.
- Procesos de interoperabilidad con plataformas gubernamentales adicionales que no formen parte del flujo actual del sistema.

1.3. Definiciones, acrónimos y abreviaturas

<i>Nombre</i>	<i>Descripción</i>
RF	Requerimiento Funcional.
RNF	Requerimiento No Funcional.
GESCOMP	Gestión Comercial Palermo Huila

1.4. Referencias

- **Requerimientos funcionales**
 - RF-01 Gestión de Contratos de Arrendamiento: Permite crear, administrar y consultar contratos, generando documentos PDF y enviando notificaciones automáticas al arrendatario.

- RF-02 Gestión de Obligaciones Mensuales: Genera obligaciones basadas en UVT e IVA, calcula mora, registra pagos y crea reportes financieros.
- RF-03 Gestión de Establecimientos: Permite registrar, actualizar y eliminar establecimientos con información detallada e imágenes.
- RF-04 Gestión de Plazas Comerciales: Permite administrar plazas y consultar los establecimientos asociados.
- RF-05 Sistema de Citas: Facilita la solicitud, registro y asignación de citas para visitar establecimientos.
- RF-06 Gestión de Usuarios: Permite registrar usuarios, gestionar su información y controlar su acceso.
- RF-07 Gestión de Roles: Permite crear y administrar roles del sistema, evitando duplicados y roles en uso.
- RF-08 Gestión de Permisos: Asigna permisos granulares por formulario a cada rol del sistema.
- RF-09 Gestión de Personas: Administra la información personal de ciudadanos y arrendatarios, evitando duplicidades de documento.
- RF-10 Gestión Geográfica: Gestiona departamentos y ciudades utilizados en direcciones y contratos.
- RF-11 Sistema de Notificaciones: Envía notificaciones internas y por correo ante eventos como pagos, vencimientos o cambios de estado.
- RF-12 Gestión de Parámetros del Sistema: Permite configurar valores clave como UVT, IVA y mora con vigencia temporal.
- RF-13 Gestión de Imágenes: Permite subir y administrar imágenes de establecimientos almacenadas en la nube.
- RF-14 Generación de Documentos PDF: Crea automáticamente documentos PDF de contratos basados en plantillas HTML.
- RF-15 Gestión de Cláusulas Contractuales: Administra cláusulas que pueden asociarse a contratos.

- RF-16 Pago de Obligaciones vía Mercado Pago: Genera checkout, procesa pagos, recibe webhooks y actualiza el estado de las obligaciones.
- **Requerimientos no funcionales**
 - RNF-01 Seguridad de Autenticación y Autorización: Uso de JWT, cookies seguras, roles, permisos y autenticación en dos pasos.
 - RNF-02 Seguridad de Datos: Protección mediante hashing, manejo seguro de contraseñas y datos sensibles.
 - RNF-03 Comunicación en Tiempo Real: Notificaciones instantáneas mediante SignalR.
 - RNF-04 Envío de Correos: Sistema de correo SMTP para notificaciones, 2FA y envío de PDFs.

1.5. Visión general del software

GESCOMPH es una plataforma digital integral diseñada para apoyar la gestión administrativa del arriendo de establecimientos comerciales ubicados en las plazas públicas del municipio de Palermo, Huila. La solución proporciona una visión centralizada del ciclo completo de arrendamiento, abarcando desde el registro de plazas y establecimientos hasta la administración de contratos, obligaciones mensuales, pagos electrónicos y control de ocupación.

El sistema se concibe como una herramienta institucional que moderniza y sistematiza los procesos relacionados con la gestión del espacio público comercial. GESCOMPH ofrece módulos especializados para funcionarios administrativos, arrendatarios y ciudadanos, permitiendo la consulta, seguimiento y administración de información de manera confiable, estructurada y accesible desde cualquier dispositivo con conexión a internet.

La plataforma incorpora funcionalidades esenciales como cálculo automático de obligaciones y mora, integración con pasarela de pagos para recaudo electrónico, generación de documentos PDF, manejo de imágenes y mecanismos de notificación interna y por correo. Su diseño modular facilita la escalabilidad.

1.6. Objetivos

1.6.1. Objetivo General

El objetivo general del presente análisis es comprender de manera estructurada el dominio funcional del sistema GESCOMP, identificando los procesos, actores, entidades, reglas de negocio y relaciones que intervienen en la gestión de arriendos de establecimientos comerciales en las plazas públicas del municipio de Palermo. Este análisis busca establecer una base conceptual sólida que permita orientar el diseño, modelado e implementación del software de forma coherente con las necesidades institucionales.

1.6.2. Objetivos Específicos

- **Identificar los procesos clave del negocio**, incluyendo la administración de plazas, establecimientos, contratos, obligaciones mensuales, pagos electrónicos, citas y consultas ciudadanas.
- **Analizar y definir los actores del sistema**, sus roles, permisos y las interacciones que mantienen con los distintos módulos del software.
- **Documentar las reglas de negocio** que gobiernan el ciclo de vida de contratos, obligaciones, pagos, mora, disponibilidad de locales y solicitudes ciudadanas.
- **Describir los flujos operativos y escenarios de uso**, representando las actividades más relevantes mediante diagramas de procesos, casos de uso y secuencias.
- **Identificar restricciones y condiciones del entorno**, tales como consideraciones tecnológicas, operativas, de seguridad y de integridad de datos.

2. Descripción General del Software

GESCOMP es un sistema de información institucional orientado a la administración integral del arriendo de establecimientos comerciales ubicados en las plazas públicas del municipio de Palermo, Huila. El software centraliza los procesos relacionados con la gestión de plazas, establecimientos, contratos, obligaciones mensuales, pagos electrónicos, control de ocupación y atención ciudadana, proporcionando una plataforma moderna, segura y accesible desde cualquier navegador o dispositivo móvil.

El sistema está concebido como una solución modular que integra los procesos administrativos más relevantes del dominio del arriendo municipal, permitiendo a la Alcaldía disponer de información confiable, actualizada y estructurada para la toma de decisiones. GESCOMP incorpora mecanismos de cálculo automático de obligaciones y mora, validación de pagos a través de la pasarela Mercado Pago, generación de documentos, gestión de imágenes y control de acceso basado en roles y permisos.

A nivel operativo, el software interactúa con tres tipos principales de usuarios:

- **Personal administrativo**, encargado de gestionar plazas, establecimientos, contratos, obligaciones, pagos, citas, reportes y parámetros del sistema.
- **Arrendatarios**, quienes consultan sus contratos, visualizan obligaciones mensuales, realizan pagos en línea y reciben notificaciones relevantes.
- **Ciudadanos no autenticados**, quienes pueden consultar locales disponibles y registrar solicitudes de visita.

2.1. Perspectiva del producto

GESCOMPH (Gestión Comercial Palermo Huila) se concibe como un sistema de información institucional que centraliza los procesos asociados al arriendo de establecimientos comerciales en las plazas públicas del municipio. Desde la perspectiva del dominio, el producto representa una solución integrada que articula el registro, control y seguimiento de los elementos clave del ciclo de arriendo: plazas, establecimientos, arrendatarios, contratos, obligaciones mensuales, pagos electrónicos y solicitudes ciudadanas.

El sistema está estructurado de forma modular para reflejar los principales procesos del negocio, tales como la administración de contratos, la gestión de pagos y mora, la disponibilidad de locales, la atención de arrendatarios y la programación de citas. Esta modularidad permite comprender el funcionamiento del sistema desde una visión conceptual, antes de su diseño técnico o implementación.

GESCOMPH se relaciona con tres grupos de actores principales:

- **Personal administrativo**, responsable de ejecutar y supervisar los procesos de gestión dentro del dominio institucional.
- **Arrendatarios**, quienes interactúan con el sistema para consultar su información contractual, obligaciones, pagos electrónicos y notificaciones asociadas.
- **Ciudadanos no autenticados**, quienes pueden acceder a información pública sobre locales disponibles y registrar solicitudes de visita.

2.2. Funcionalidades generales

GESCOMPH integra un conjunto de funcionalidades orientadas a la administración del ciclo completo de arriendo de los establecimientos comerciales en las plazas públicas de Palermo. De forma general, el sistema abarca las siguientes áreas funcionales:

- **Gestión de plazas y establecimientos**, permitiendo administrar la información estructural del espacio público comercial y su disponibilidad.
- **Administración de contratos de arrendamiento**, incluyendo el registro, seguimiento, vigencia, estados y asociación con arrendatarios y establecimientos.

- **Generación y control de obligaciones mensuales**, incorporando el cálculo automático de mora y la actualización de los valores según parámetros institucionales.
- **Procesamiento y validación de pagos electrónicos**, a través de la integración con la pasarela de pagos Mercado Pago y la gestión de confirmaciones mediante webhooks.
- **Consultas especializadas para arrendatarios**, permitiendo visualizar contratos, obligaciones mensuales, historial de pagos y notificaciones relevantes.
- **Consulta pública de locales disponibles**, con acceso para ciudadanos no autenticados y filtros de búsqueda según características del establecimiento.
- **Gestión de solicitudes de citas**, permitiendo a ciudadanos registrar visitas y a los administradores validar, aprobar o rechazar dichas solicitudes.
- **Gestión de usuarios, roles y permisos**, asegurando el control de acceso y la visibilidad adecuada de funcionalidades para cada perfil.
- **Administración de imágenes asociadas a los establecimientos**, mediante un servicio de almacenamiento externo.
- **Generación de reportes y visualización de estadísticas**, orientadas al análisis de ocupación, pagos, mora y gestión administrativa.

Estas funcionalidades conforman la estructura general del sistema y representan los principales procesos de negocio analizados dentro del dominio del arriendo municipal.

2.3. Tipos de usuarios y roles

Dentro del dominio de GESCOMPH se identifican varios tipos de usuarios y actores que interactúan con el sistema, cada uno con responsabilidades específicas y niveles diferenciados de acceso. Estos perfiles determinan cómo se relacionan con los procesos del negocio y qué información pueden consultar o gestionar.

1. Administrador

Corresponde al personal de la Alcaldía responsable de la gestión operativa del sistema. Sus responsabilidades incluyen:

- Administrar plazas y establecimientos.
- Registrar y actualizar contratos de arrendamiento.
- Gestionar obligaciones mensuales y validar pagos electrónicos.
- Supervisar la ocupación de locales y el estado financiero de los arrendatarios.
- Revisar y gestionar las solicitudes de citas registradas por los ciudadanos.
- Configurar roles, permisos y usuarios del sistema.
- Administrar parámetros institucionales (IVA, UVT, mora).
- Gestionar imágenes y documentos asociados a los establecimientos.
- Supervisar reportes y estadísticas generadas por el sistema.

2. Arrendatario

Es la persona que mantiene un contrato vigente con la Alcaldía. Interactúa con el sistema para:

- Consultar la información de sus contratos.
- Visualizar obligaciones mensuales y valores actualizados en caso de mora.
- Realizar pagos electrónicos a través de la pasarela integrada.
- Historial de pagos y notificaciones internas.
- Revisar la disponibilidad de locales.
-

3. Ciudadano no autenticado

Corresponde a cualquier persona interesada en conocer la disponibilidad de locales. Puede:

- Consultar la lista de locales disponibles para arriendo.
- Visualizar características básicas e imágenes de los establecimientos.
- Registrar solicitudes de citas para visitar los locales.

Su interacción es informativa y no involucra acceso a procesos administrativos o datos privados.

4. Actores del sistema no humanos

Sistema de pagos (Mercado Pago)

Interactúa con GESCOMPH para:

- Crear preferencias de pago.
- Procesar transacciones.
- Enviar notificaciones (webhooks) confirmando pagos, pendientes o rechazados.

Servicio de almacenamiento de imágenes

Permite almacenar y recuperar imágenes de los establecimientos.

Servicio de correo

Envía notificaciones asociadas a pagos, citas o eventos relevantes del sistema.

2.4. Restricciones

El sistema GESCOMPH está sujeto a un conjunto de restricciones derivadas del dominio administrativo, de la infraestructura tecnológica disponible y de las políticas institucionales de la Alcaldía de Palermo. Las principales restricciones son las siguientes:

1. Restricciones del dominio

- El sistema debe ajustarse a los procesos administrativos establecidos por la Alcaldía para la gestión de arriendos en plazas públicas, sin modificar la normativa municipal vigente.
- Los cálculos de obligaciones y mora deben basarse en los parámetros institucionales definidos (IVA, UVT y tasas de interés).
- Toda solicitud de cita debe ser validada por un funcionario antes de ser confirmada.
- La información de arrendatarios y contratos debe considerarse confidencial conforme a la normatividad de protección de datos.

2. Restricciones de seguridad y acceso

- El sistema debe aplicar control de acceso basado en roles y permisos; ninguna funcionalidad crítica podrá ser accedida sin autenticación y autorización.
- Los arrendatarios no podrán modificar contratos, obligaciones ni parámetros del sistema.
- Los ciudadanos no autenticados solo podrán consultar información pública de locales disponibles.

3. Restricciones técnicas

- El sistema debe operar en una arquitectura cliente-servidor basada en API REST utilizando .NET 8.
- El almacenamiento de imágenes debe realizarse mediante un servicio externo compatible (Cloudinary).
- La validación y confirmación de pagos dependerá exclusivamente de los webhooks enviados por la pasarela Mercado Pago.

4. Restricciones de integración

- La plataforma no se integrará con sistemas contables, presupuestales, jurídicos o tributarios externos en su versión actual.
- No se incluirán mecanismos de firma digital avanzada, interoperabilidad con plataformas de gobierno.
- El sistema depende de los servicios externos de Mercado Pago y del proveedor de almacenamiento de imágenes.

5. Restricciones operativas

- Los administradores solo podrán modificar parámetros institucionales mediante autenticación reforzada.
- Ningún usuario podrá eliminar registros esenciales (plazas, contratos, obligaciones), solo inhabilitarlos o marcarlos como eliminados.

2.5. Suposiciones y Dependencias

Suposiciones

- **Disponibilidad del entorno tecnológico institucional:**
Se asume que la Alcaldía de Palermo dispone de infraestructura para alojar la API, base de datos, almacenamiento de imágenes y servicios complementarios.
- **Conectividad adecuada:**
Se asume que los usuarios (administradores, arrendatarios y ciudadanos) cuentan con conexión a internet estable y acceso desde navegadores modernos.
- **Disponibilidad del personal para validaciones:**
Se asume que los funcionarios participarán activamente en levantamiento de requerimientos, pruebas funcionales, validación de módulos y retroalimentación.
- **Actualización de parámetros institucionales:**
Se asume que la Alcaldía proporcionará oportunamente valores de IVA, UVT y tasas de mora para el cálculo de obligaciones.
- **Capacitación básica:**
Se asume que los usuarios administrativos recibirán capacitación inicial para operar el sistema.

Dependencias

1. Dependencias de software

- **Frontend:** Angular 20, Angular Material.
- **Backend:** .NET 8, Entity Framework Core, Mapster, FluentValidation, JWT.
- **DevOps:** GitHub, Docker, Azure App Service (o infraestructura equivalente).
- **Gestión:** Trello, Visual Studio / VS Code.

2. Dependencias de servicios externos

- **Pasarela Mercado Pago:** creación de preferencias, validación de pagos, webhooks.
- **Almacenamiento de imágenes:** Cloudinary o almacenamiento equivalente.
- **Servicio de correo:** SMTP, u otro proveedor configurado.

3. Dependencias de infraestructura

- Base de datos relacional disponible (SQL Server).
- Conectividad de red estable entre cliente, API y BD.
- Certificados SSL/TLS vigentes.

2.6. Historias de usuario

- Administrador
 - HU-001 – Generar contrato

Como administrador,

quiero crear contratos de arrendamiento,

para formalizar la asignación de locales a los arrendatarios de manera estandarizada y trazable.
 - Criterios de aceptación
 - El sistema debe validar que el local esté disponible.
 - Debe generarse un documento PDF del contrato.
 - HU-002 – Ver estadísticas del sistema

Como administrador,

quiero visualizar estadísticas sobre pagos realizados,

para tomar decisiones informadas sobre la gestión financiera.
 - Criterios de aceptación
 - Deben mostrarse métricas de pagos diarios, mensuales y de los últimos seis meses.
 - HU-003 – Gestionar plazas y establecimientos

Como administrador,

quiero registrar, editar y desactivar plazas y establecimientos,

para mantener actualizada la información del espacio público comercial.
 - Criterios de aceptación
 - No puede eliminarse una plaza con locales activos.
 - No puede desactivarse un establecimiento con contrato vigente.

- HU-004 – Gestionar usuarios del sistema

Como administrador,
quiero crear, editar y desactivar cuentas de usuario,
para administrar el acceso al sistema y los permisos asignados.
- Criterios de aceptación
 - No debe permitirse el registro de correos duplicados.
 - La desactivación no elimina información histórica.
 - Las contraseñas deben almacenarse cifradas.
- HU-005 – Gestionar solicitudes de cita

Como administrador,
quiero visualizar y actualizar solicitudes de cita,
para organizar eficientemente la atención a ciudadanos interesados.
- Criterios de aceptación
 - Mostrar citas.
 - Debe poder actualizar el estado de las citas de manera correcta.
- HU-006 – Validar disponibilidad del local

Como administrador,
quiero que el sistema impida asignar contratos a locales ya ocupados,
para evitar conflictos o duplicidad de asignaciones.
- Criterios de aceptación*
 - El sistema debe validar el estado del local antes de permitir la asignación.
 - Si el local está ocupado, debe mostrar un mensaje de restricción.

- HU-007 – Evitar eliminación de plazas con contratos activos
Como administrador,
quiero que el sistema bloquee la eliminación de plazas con contratos vigentes,
para preservar la integridad de la información del dominio.
- Criterios de aceptación*
 - Debe mostrarse un mensaje indicando el motivo del bloqueo.
- HU-008 – Gestionar catálogo de locales
Como administrador,
quiero registrar y actualizar la información técnica de un local,
para mantener un inventario completo y confiable de los espacios disponibles.
- Criterios de aceptación*
 - Debe permitir registrar área, ubicación, características y estado.
 - Debe permitir asociar imágenes del establecimiento.
- HU-009 – Configurar parámetros del sistema
Como administrador,
quiero modificar el IVA, la UVT y la tasa de mora,
para garantizar que los cálculos del sistema se ajusten a las normativas vigentes.
- Criterios de aceptación
 - Los parámetros deben aplicarse a nuevas obligaciones generadas.
 - Solo el administrador puede modificar estos valores.

- HU-010 – Validar pagos electrónicos

Como administrador,

quiero recibir la confirmación automática de los pagos electrónicos realizados por los arrendatarios,

para mantener actualizado el estado financiero de los contratos sin intervención manual.
- Criterios de aceptación
 - El sistema debe procesar correctamente los webhooks de Mercado Pago.
 - El estado de la obligación debe actualizarse automáticamente.
- HU-011 – Administrar imágenes de establecimientos

Como administrador,

quiero cargar, visualizar y eliminar imágenes asociadas a los locales,

para proporcionar información visual clara a los ciudadanos interesados.
- Criterios de aceptación
 - Debe validar extensión y tamaño de las imágenes.
 - Las imágenes deben almacenarse en un servicio externo (p. ej. Cloudinary).
 - Debe permitir eliminar imágenes sin afectar otras asociadas.
- HU-012 – Recibir notificaciones del sistema

Como administrador,

quiero recibir notificaciones internas y por correo sobre eventos relevantes,

para estar informado sobre pagos confirmados, citas solicitadas y cambios en contratos.
- Criterios de aceptación
 - Deben enviarse correos en eventos configurados (pagos, citas, etc.).

- Arrendatario
 - HU-A01 – Iniciar sesión

Como arrendatario,
quiero iniciar sesión con mis credenciales,
para acceder de manera segura a mi información contractual y financiera.
 - Criterios de aceptación*
 - El sistema debe validar correo y contraseña.
 - Las contraseñas deben compararse mediante hash.
 - Debe retornarse un JWT y refresh token válidos.
 - El usuario no podrá ingresar si está inactivo.
 - HU-A02 – Cerrar sesión

Como arrendatario,
quiero cerrar sesión,
para evitar accesos no autorizados a mi cuenta.
 - Criterio de aceptación*
 - Debe invalidarse el token de sesión.
 - Debe redirigir a la pantalla de inicio.
 - HU-A03 – Recuperar contraseña

Como arrendatario,
quiero recuperar el acceso mediante correo electrónico,
para poder ingresar en caso de olvidar mi contraseña.
 - Criterios de aceptación*
 - Debe enviarse un correo con un enlace temporal.
 - El enlace debe expirar según los parámetros del sistema.
 - Debe permitir establecer una nueva contraseña cumpliendo políticas de seguridad.

- HU-A04 – Cambiar mi contraseña

Como arrendatario,
quiero cambiar mi contraseña desde el sistema,
para mantener la seguridad de mi cuenta.
- Criterios de aceptación
 - Debo confirmar mi contraseña actual.
 - La nueva contraseña debe cumplir los requisitos mínimos.
 - El cambio debe registrarse en auditoría.
- HU-A05 – Consultar mis contratos

Como arrendatario,
quiero visualizar mis contratos activos y anteriores,
para conocer el estado de mi relación contractual con la Alcaldía.
- Criterios de aceptación
 - Debe mostrar estado, fecha de inicio, vigencia y valor del contrato.
 - Debe permitir descargar el contrato en PDF.
 - Solo deben aparecer contratos asociados a mi usuario.
- HU-A06 – Ver mis obligaciones mensuales

Como arrendatario,
quiero visualizar mis obligaciones mensuales, fechas de vencimiento y valores,
para evitar retrasos y cargos por mora.
- Criterios de aceptación
 - Debe mostrar obligaciones pagadas, pendientes y vencidas.
 - Debe mostrar el cálculo actualizado de mora cuando aplique.
 - Debe mostrar link para pagar electrónicamente.

- HU-A07 – Consultar historial de pagos

Como arrendatario,

quiero ver mis obligaciones mensuales con estado (pagadas, pendientes, vencidas),

para tener claridad sobre mi estado financiero.
- Criterios de aceptación
 - Cada obligación debe incluir: fecha, monto y estado.
 - Debe mostrar obligaciones confirmadas por la pasarela.
- HU-A08 – Realizar pagos electrónicos

Como arrendatario,

quiero pagar mis obligaciones directamente desde el sistema,

para evitar desplazamientos y asegurar mis pagos a tiempo.
- Criterios de aceptación
 - Debe crearse una preferencia de pago en Mercado Pago.
 - El sistema debe redirigirme al checkout oficial.
 - Al concluir el pago, el sistema debe actualizar el estado mediante webhook.
 - Debe generarse comprobante y notificación interna.
- HU-A09 – Recibir notificaciones del sistema

Como arrendatario,

quiero recibir notificaciones internas y por correo sobre obligaciones, pagos y estados,

para mantenerme informado de vencimientos y movimientos financieros.
- Criterios de aceptación*
 - Debe notificar nuevas obligaciones mensuales.
 - Debe notificar pagos confirmados, rechazados o pendientes.
 - Debe notificar cuando un valor entra en mora.
 - Debe existir un historial de notificaciones.

- HU-A10 – Consultar fechas de pagos

Como arrendatario,

quiero ver mis obligaciones mensuales con fechas de vencimiento,

para evitar retrasos.
- Criterios de aceptación*
 - Debe mostrar vencimientos mensuales.
- HU-A11 – Ver locales disponibles *(corregida)*

Como arrendatario,

quiero consultar locales disponibles en caso de querer arrendar otro contrato,

para evaluar opciones.
- Usuario público (ciudadano interesado)
 - HU-C01 – Consultar locales disponibles

Como ciudadano interesado,

quiero ver los locales disponibles en las plazas públicas,

para identificar opciones que se ajusten a mis necesidades comerciales.
 - Criterios de aceptación
 - Deben mostrarse únicamente locales marcados como "disponibles".
 - Debe permitir filtrar por plaza, área, tipo o características.
 - No debe mostrar información privada de arrendatarios o contratos vigentes.

- HU-C02 – Ver detalles e imágenes del local

Como ciudadano interesado,

quiero ver la información detallada de un local, incluyendo características e imágenes,

para evaluar si el espacio es adecuado para mi actividad antes de solicitar una cita.

- Criterios de aceptación

- Debe mostrar área, tipo, ubicación y descripción.
- Debe mostrar las imágenes cargadas por el administrador.
- No debe mostrar datos sensibles ni información interna del sistema.

- HU-C03 – Solicitar una cita para conocer un local

Como ciudadano interesado,

quiero enviar una solicitud de cita para visitar un local disponible,

para obtener información presencial antes de iniciar un posible proceso de arriendo.

- Criterios de aceptación

- Debo proporcionar datos mínimos: nombre, correo, teléfono y fecha sugerida.
- El sistema debe registrar la solicitud con estado "Pendiente".
- No debe permitirse solicitar cita para un local ocupado.
- El administrador debe recibir notificación interna.

- HU-C04 – Recibir confirmación o rechazo de la cita por correo

Como ciudadano interesado,

quiero recibir notificaciones internas sobre el estado de mi solicitud de cita,

para conocer el estado sin necesidad de ingresar al sistema.

- Criterios de aceptación

- Las notificaciones deben registrarse con estado (leída / no leída).

- HU-C06 – Ver disponibilidad actualizada del local

Como ciudadano interesado,

quiero que el sistema solo muestre como disponibles los locales realmente libres,

para evitar solicitar citas de espacios ya asignados.

- Criterios de aceptación*

- El sistema debe sincronizar estado con los contratos activos.

- No debe permitir enviar solicitudes sobre locales ocupados.

- Si el local cambia de estado mientras veo la pantalla, debe actualizarse al refrescar.

- HU-C07 – Filtrar locales según criterios específicos

Como ciudadano interesado,

quiero aplicar filtros de búsqueda por plaza,

para encontrar fácilmente locales que cumplan con mis requisitos.

- Criterios de aceptación

- Filtros disponibles: plaza.

- Al aplicar filtros, la lista debe actualizarse dinámicamente.

- Debe permitir limpiar filtros.

3. Arquitectura del sistema

3.1. Estilo arquitectónico general

GESCOMPH adopta una arquitectura multicapa basada en principios basada en **separación de responsabilidades**, estructurada en módulos independientes que permiten escalabilidad, mantenibilidad y claridad en el dominio del negocio. El sistema se divide en tres capas principales que se comunican mediante contratos bien definidos:

1. Capa de Presentación (Frontend)

Implementada en **Angular 20**, esta capa gestiona la interacción con el usuario final. Se encarga de:

- Renderizar la interfaz web responsiva.
- Consumir los servicios expuestos por la API mediante HTTP/JSON.
- Gestionar autenticación basada en JWT y Refresh Tokens.
- Administrar estados locales mediante servicios y almacenamiento seguro.
- Manejar validaciones de formularios, visualización de tablas, calendarios, reportes e interfaces dinámicas.

2. Capa de Aplicación y Negocio (Backend)

Desarrollada en **.NET 8**, implementa la lógica central del sistema mediante:

- Servicios de dominio y lógica de negocio.
- Validaciones mediante FluentValidation.
- Mapeo de entidades y DTOs mediante Mapster.
- Controladores API (REST) que exponen los recursos al frontend.
- Gestión de seguridad: JWT, Refresh Tokens, roles, permisos y auditoría.
- Integraciones externas:
 - **Mercado Pago** (checkout + webhooks para validación de pagos)
 - **Cloudinary / almacenamiento de imágenes**
 - **Servicio de correo SMTP**

3. Capa de Persistencia (Base de Datos)

Soportada en **SQL Server** mediante:

- Entity Framework Core para ORM
- Repositorios genéricos y específicos
- Trazabilidad mediante columnas de auditoría
- Integridad referencial entre entidades

Principios arquitectónicos clave

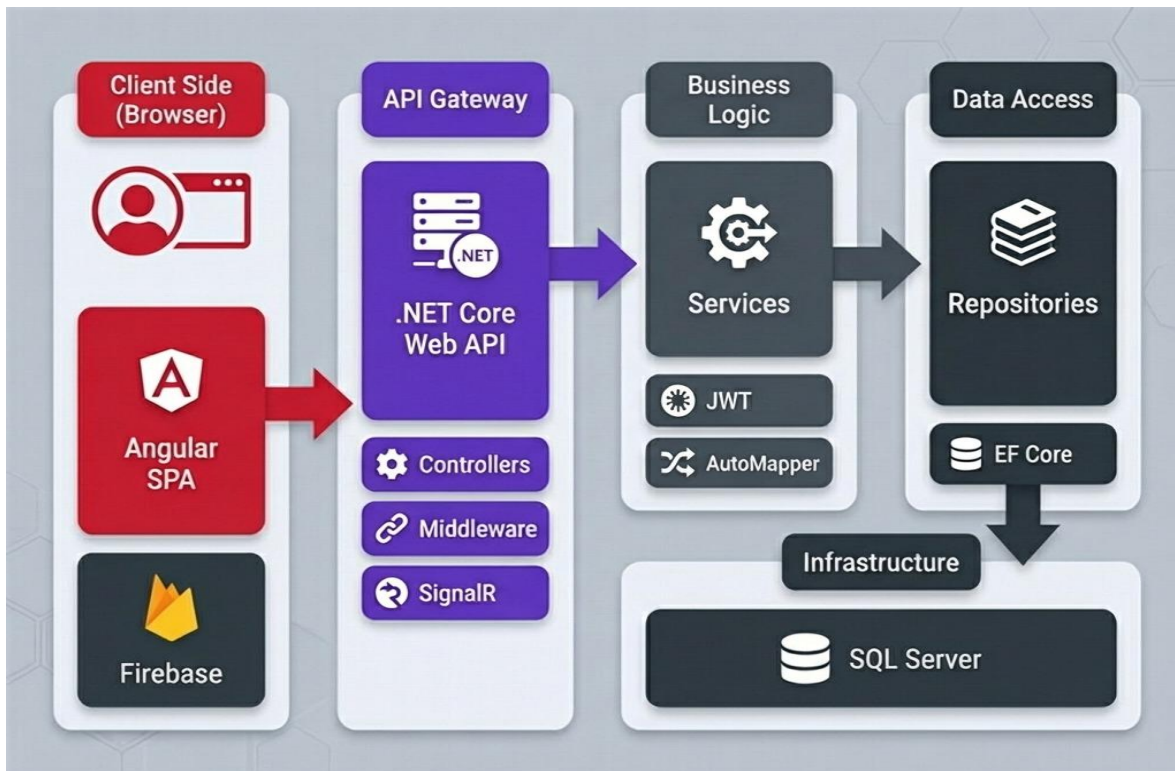
El diseño se fundamenta en:

- Separación estricta de responsabilidades
- Inversión de dependencias
- Modularidad y escalabilidad
- Uso de DTOs para transporte seguro de datos
- Servicios especializados por módulo
- Patrón Repository + Business Services
- Trazabilidad y auditoría en operaciones críticas
- Integraciones desacopladas mediante adaptadores

Modelo de comunicación

- Comunicación entre frontend y backend mediante **REST API JSON**.
- Comunicación backend → servicios externos mediante **HTTP/REST**.
- Procesamiento asíncrono de pagos mediante **webhooks de Mercado Pago**.
- Almacenamiento externo de archivos mediante SDK o API de proveedor elegido.

3.2. Diagrama general



3.3. Componentes principales

El sistema GESCO PH está compuesto por los siguientes componentes estructurales:

1. Interfaz de usuario (Frontend) – Angular 20

- Implementa la capa de presentación.
- Proporciona las interfaces para administradores, arrendatarios y ciudadanos.
- Gestiona formularios, validaciones, tablas dinámicas, listados y flujos de navegación.
- Consume los servicios REST del backend mediante HTTP.
- Maneja autenticación basada en JWT y almacenamiento seguro de tokens.

2. API REST (Backend) – .NET 8

- Expone los servicios del sistema mediante controladores.
- Implementa la lógica de negocio en servicios dedicados (Business layer).
- Administra reglas de negocio: contratos, pagos, mora, obligaciones mensuales, citas, roles, permisos.
- Integra servicios externos:
 - Pasarela Mercado Pago
 - Servicio de imágenes (Cloudinary)
- Implementa validaciones con FluentValidation y mapeos con Mapster.

3. Capa de persistencia – Entity Framework Core

- Gestiona la comunicación entre la API y la base de datos relacional.
- Define entidades del dominio, relaciones, migraciones, restricciones y claves foráneas.
- Implementa repositorios genéricos y específicos.
- Administra trazabilidad mediante campos de auditoría.

4. Base de datos relacional (SQL Server)

- Almacena información estructurada sobre usuarios, contratos, pagos, obligaciones, plazas, establecimientos, citas, auditorías, parámetros del sistema y roles.
- Implementa índices, llaves primarias, llaves externas e integridad referencial.
- Aplica migraciones generadas desde EF Core.

5. Servicios externos complementarios

- **Mercado Pago:** pagos electrónicos + validación automática mediante webhooks.
- **Servicio de imágenes:** almacenamiento y recuperación de imágenes de establecimientos.
- **Correo SMTP :** notificaciones por correo a arrendatarios y ciudadanos.

3.4. Tecnologías utilizadas

Frontend

- **Angular 20** – Framework para aplicaciones web SPA.
- **Angular Material** – Componentes UI estandarizados.
- **CSS** – Estilos, temas y diseño responsivo.
- **RxJS** – Manejo de programación reactiva.

Backend

- **C# .NET 8** – Lenguaje y framework principal del lado servidor.
- **ASP.NET Core Web API** – Exposición de servicios REST.
- **Entity Framework Core 8** – ORM para acceso a datos.
- **Mapster** – Mapeo entre entidades y DTOs.
- **FluentValidation** – Validación de reglas de entrada.

Base de datos

- **SQL Server 2019**

Servicios externos

- **Mercado Pago SDK** – Integración de pagos.
- **Cloudinary** – Almacenamiento de imágenes.
- **SMTP / SendGrid** – Envío de correos.

3.5. Seguridad

El sistema implementa un conjunto de mecanismos de seguridad orientados a proteger la integridad, confidencialidad y disponibilidad de la información:

1. Autenticación y Autorización

- **JWT (JSON Web Tokens)** para autenticación.
- **Refresh Tokens** para reutilización segura de sesiones.
- **Control de acceso basado en roles y permisos**, configurados desde el panel administrativo.
- **Restricción de endpoints** según rol del usuario.

2. Seguridad de datos

- **Cifrado de contraseñas con hashing** (HMAC-SHA256, bcrypt u otro).
- **Validación estricta de datos** en Backend con FluentValidation.

3. Seguridad en API y comunicaciones

- Uso obligatorio de **HTTPS**.
- **CORS restringido** a dominios autorizados.
- **Protección contra ataques comunes:**
 - SQL Injection (mediante ORM)
 - CSRF (JWT y cookies HttpOnly)

4. Análisis de viabilidad

4.1. Viabilidad técnica

El desarrollo de GESCOMPH es técnicamente viable gracias al uso de tecnologías modernas, estables y ampliamente soportadas. El sistema se construirá bajo una arquitectura multicapa con una API REST en **.NET 8**, un frontend en **Angular 20**, y una base de datos relacional en **SQL Server** o **PostgreSQL**, lo que garantiza escalabilidad, mantenibilidad y compatibilidad a largo plazo.

La integración con servicios externos como **Mercado Pago** (pagos electrónicos y webhooks), **Cloudinary** (almacenamiento de imágenes) y servicios de correo SMTP es técnicamente posible mediante SDKs oficiales y protocolos estándar.

4.2 Viabilidad económica

Desde la perspectiva académica, el proyecto es totalmente viable, ya que **no implica costos económicos directos**.

Todo el desarrollo se ha realizado utilizando herramientas gratuitas:

- **Visual Studio Community**
- **Angular CLI**
- **.NET SDK**
- **SQL Server Express / PostgreSQL**
- **Repositorios GitHub**
- **Trello u otras herramientas de gestión gratuitas**

Las integraciones implementadas (por ejemplo, Mercado Pago y almacenamiento de imágenes) se utilizan **en modo de pruebas (sandbox)**, sin generar gastos de operación.

4.3 Viabilidad operativa

GESCOMPH modela fielmente los flujos administrativos asociados al arriendo de locales en plazas públicas. Sus funcionalidades se alinean con procesos reales: gestión de plazas, contratos, pagos, obligaciones mensuales, cálculo de mora y atención de solicitudes ciudadanas.

Durante el proyecto académico, el sistema es operativamente viable porque:

- La interfaz es intuitiva.
- Los módulos representan procesos reales simplificados.
- Los usuarios pueden aprender rápidamente las funciones básicas.
- La arquitectura es clara y fácil de mantener.

4.4 Viabilidad Legal

Aunque el sistema se desarrolla con fines académicos, se consideran lineamientos básicos relacionados con la **Ley 1581 de 2012** sobre protección de datos personales, aplicando principios fundamentales:

- No almacenar información sensible innecesaria.
- Cifrar contraseñas.
- Implementar control de acceso por roles.
- Garantizar la privacidad de los datos simulados usados en las pruebas.

La integración con pagos electrónicos se realiza únicamente en modo **sandbox**, sin manejar recursos económicos reales ni información financiera de usuarios, eliminando riesgos legales.

5. Interfaz de usuario y prototipos

5.1 Lineamientos generales de diseño

Identidad visual

- Uso de la **paleta institucional de la Alcaldía de Palermo** (colores sobrios, tonos verdes/azules según manual de identidad).
- Tipografías consistentes en toda la plataforma (Inter, Roboto o equivalente).

Diseño responsivo

- La interfaz debe adaptarse a dispositivos **PC, tablets y móviles** mediante grid flexible.
- Componentes como tablas, formularios y tarjetas deben reestructurarse dinámicamente.

Navegación clara

- Menú lateral permanente para módulos principales.
- Encabezados con migas de pan (breadcrumbs) para navegación interna.
- Acciones primarias destacadas con botones de color institucional.

Accesibilidad

- Cumplimiento básico de **WCAG 2.1 AA**, incluyendo:
 - Contraste de color adecuado.
 - Navegación por teclado en formularios.
 - Etiquetas descriptivas para lectores de pantalla.

Consistencia

- Uso uniforme de componentes Angular Material/Tailwind.
- Botones, tarjetas, iconografía y espaciados estandarizados.

5.2 Componentes comunes

Estos componentes representan los “bloques base” del diseño y aparecen en múltiples pantallas:

1. Formularios

- Validación visual.
- Indicadores de error y mensajes descriptivos.
- Inputs con formatos según tipo: texto, fecha, correo, teléfono, numérico.

2. Tablas dinámicas

- Filtrado por columnas.
- Ordenamiento asc/desc.
- Paginación integrada.
- Estado de carga (loading skeletons).

3. Tarjetas informativas

- Para resúmenes de contrato, citas, pagos o estados del local.

4. Diálogos modales

- Confirmación de operaciones sensibles (eliminar, aprobar, rechazar).

5. Componentes de navegación

- Sidebar (menú lateral)

6. Feedback visual

- Spinners
- Notificaciones internas (toast/snackbar)
- Banners de alerta

5.3 Prototipos de pantalla

Pantallas principales del sistema

Administrador

1. Panel principal del administrador

- Muestra indicadores clave: número de locales arrendados, disponibles, cantidad de contratos, acciones rápidas.
- Acceso rápido a los módulos de contratos, plazas, establecimientos, usuarios, módulo de seguridad y configuraciones.

2. Gestión de plazas

- Permite registrar, editar y eliminar plazas públicas.
- Incluye listado con filtros por nombre o ubicación.
- 3. Gestión de establecimientos
 - CRUD de locales comerciales asociados a una plaza.
 - Se muestra el estado de ocupación y enlace al contrato activo.
- 4. Gestión de módulo de seguridad
 - CRUD sobre roles, permisos, módulos, formularios.
- 5. Gestión de configuraciones
 - Editar la información del usuario.
 - Cambiar la contraseña
 - CRUD sobre departamentos y ciudades
 - CRUD sobre parámetros del sistema
- 6. Gestión de usuarios
 - Crear y administrar cuentas de arrendatarios.
 - Cambiar roles o recuperar credenciales.
- 7. Gestión de citas
 - Ver solicitudes de citas de ciudadanos para visitar locales.
 - Aprobar, rechazar o reprogramar visitas.

Arrendatario

- 1. Inicio / resumen
 - 1.1. Panel con resumen del contrato, estado de pago y próximas fechas.
- 2. Mis contratos
 - 2.1. Consulta de contratos vigentes y vencidos, con detalles del establecimiento y cláusulas.
- 3. Configuraciones
 - 3.1. Editar sus datos personales y cambiar contraseña.

Ciudadano (no autenticado)

1. Exploración de locales disponibles
 - Listado de locales con filtros por plaza, tamaño o tipo.
 - Vista detallada con imágenes, ubicación y disponibilidad.
2. Formulario de solicitud de cita
 - Formulario para agendar visita presencial a un local disponible.
 - Ingreso de datos de contacto y fecha tentativa.

4. Plan de Mantenimiento

El plan de mantenimiento define las actividades necesarias para asegurar el funcionamiento continuo, estable y actualizado del sistema GESCOMP. Aunque este proyecto se desarrolla en un entorno académico, se establecen lineamientos que serían aplicables en un entorno real de operación.

4.1. Tipos de mantenimiento

GESCOMP podrá requerir los siguientes tipos de mantenimiento:

1. Mantenimiento correctivo

Acciones destinadas a corregir errores detectados durante el uso del sistema, tales como fallas de lógica, errores en validaciones, comportamientos inesperados o problemas de visualización.

2. Mantenimiento adaptativo

Ajustes necesarios para mantener la compatibilidad del sistema ante cambios del entorno tecnológico, como:

- Actualizaciones de Angular o .NET
- Cambios en APIs externas (Mercado Pago)
- Cambios en motores de base de datos
- Ajustes en infraestructura o despliegues

3. Mantenimiento perfectivo

Mejoras orientadas a optimizar rendimiento, usabilidad o estructura del código, incluyendo:

- Refactorización de servicios
- Mejoras de experiencia de usuario

- Optimización de consultas a la base de datos
- Reorganización de componentes Angular

4. Mantenimiento preventivo

Acciones destinadas a prevenir fallas futuras:

- Revisión periódica de logs
- Monitoreo de endpoints
- Limpieza de datos temporales
- Actualización periódica de dependencias

4.2. Procedimiento de mantenimiento

El mantenimiento del sistema seguirá las siguientes etapas:

1. **Identificación del problema o mejora**
Detectado por usuarios o desarrolladores.
2. **Registro del incidente o requerimiento**
Documentado en una herramienta de gestión (Trello, GitHub Issues).
3. **Análisis del impacto**
Se evalúa si la modificación afecta módulos críticos, reglas de negocio o integraciones externas.
4. **Implementación de la solución**
Incluye ajustes en frontend, backend o base de datos.
5. **Pruebas post-mantenimiento**
 - Pruebas unitarias
 - Pruebas funcionales
 - Validación de casos afectados
 - Pruebas de regresión si es necesario
6. **Implementación en el entorno final**
En el proyecto académico, esto corresponde al despliegue en local o servidor temporal.
7. **Documentación de cambios**
Registro de versión, descripción y módulos afectados.

4.3. Gestión de versiones

El sistema utilizará control de versiones mediante **GitHub**, siguiendo buenas prácticas de versionado semántico:

- **MAJOR**: cambios que afectan compatibilidad significativa
- **MINOR**: nuevas funcionalidades sin romper versiones anteriores
- **PATCH**: correcciones o ajustes menores

Ramas recomendadas:

- **main** — versión estable
- **develop** — nuevas características
- **feature/*** — módulos individuales
- **fix/*** — correcciones específicas

4.4. Herramientas de mantenimiento

- **GitHub** – control de versiones y gestión de issues
- **Trello** – seguimiento de tareas y priorización
- **Visual Studio / VS Code** – entorno de desarrollo
- **Postman** – pruebas de API
- **Angular CLI** – mantenimiento del frontend
- **EF Core Migrations** – actualización de la base de datos

4.5. Frecuencia y responsabilidades

Dado que el proyecto es académico:

- El equipo desarrollador será responsable del mantenimiento durante la fase de desarrollo y revisión.
- Las tareas de mantenimiento se ejecutarán según necesidades identificadas durante pruebas funcionales o retroalimentación del docente.

5. Plan de Trabajo

El plan de trabajo para el desarrollo del sistema GESCOMPH contempla cuatro fases secuenciales, con entregables y actividades bien definidas para asegurar una construcción estructurada y académicamente rigurosa del software.

Fase 1 – Análisis y Planificación

- **Objetivo:** Comprender el problema, definir claramente los requerimientos y organizar el desarrollo del proyecto.
- **Actividades principales:**
 - Recolección y validación de requisitos funcionales y no funcionales
 - Definición de actores y casos de uso del sistema
 - Identificación de restricciones, suposiciones y dependencias
 - Elaboración del Documento de Requisitos (SRS)
 - Planificación del cronograma y herramientas a usar
- **Entregables:**
 - Documento de requisitos del sistema (SRS)
 - Matriz de actores y casos de uso
 - Cronograma del proyecto
- **Herramientas:** Google Docs, Trello, Draw.io, Canva, Word, Excel

Fase 2 – Diseño del Sistema

- **Objetivo:** Definir la arquitectura y componentes del sistema, las interfaces de usuario y el modelo de datos.
- **Actividades principales:**
 - Diseño de la arquitectura en capas (presentación, lógica, datos)

- Modelado de base de datos relacional
- Creación de diagramas UML: casos de uso, clases, secuencia, procesos y flujo
- Diseño de interfaces gráficas y prototipos por rol
- Redacción de la especificación de interfaces
- Entregables:
 - Documento de análisis y diseño técnico
 - Prototipos de pantalla (wireframes)
 - Diagramas UML completos
- Herramientas: Figma, Lucidchart, Mermaid.js, Visual Paradigm, Canva

Fase 3 – Desarrollo

- Objetivo: construcción funcional del sistema, aplicando estándares de codificación y buenas prácticas.
- Actividades principales:
 - Configuración de ambiente de desarrollo local
 - Estructura del backend (.NET) y frontend (Angular)
 - Implementación parcial o por módulos para validación estructural
 - Versionamiento y control de cambios con Git
- Entregables:
 - Código fuente funcional de los principales módulos (parcial)
 - Documentación de la arquitectura técnica
 - Capturas de simulaciones funcionales
- Herramientas: Visual Studio, Angular CLI, GitHub, SQL Server Management Studio

Fase 4 – Pruebas

- Objetivo: Validar el comportamiento esperado del sistema a partir de los requisitos definidos.
- Actividades principales:
 - Elaboración del plan de pruebas (unitarias, funcionales y de interfaz)
 - Ejecución de pruebas sobre módulos desarrollados
 - Registro y análisis de resultados
 - Validación de criterios de aceptación
- Entregables:
 - Plan de pruebas
 - Evidencias de ejecución (capturas, casos, logs)
 - Informe de resultados de pruebas y mejoras sugeridas
- Herramientas: Postman (para API), Jasmine/Karma (Angular), Word, Excel

6. Diagramas

6.1. Diagrama de Flujo

[6.1.1. Diagrama de Flujo General del Administrador](#)

[6.1.2. Diagrama de Flujo del Usuario](#)

[6.1.3. Diagrama de Flujo General del Arrendatario](#)

6.2. Diagrama de Clases

6.3. Diagrama de Secuencia

[6.3.1. Diagrama General de Secuencia del Administrador](#)

[6.3.2. Diagrama General de Secuencias del Usuario](#)

[6.3.3. Diagrama General de Secuencia del Arrendatario](#)

6.4. Diagrama General de Procesos

6.5. Diagrama de Casos de Uso