



DOCUMENTACIÓN PROPUESTA TÉCNICA

GESCOMPH

Jesús David Fierro Rivera
Brayan Santiago Guerrero Mendez

Contenido

1.	Resumen Ejecutivo	3
1.1.	Situación Actual.....	3
1.2.	Problemática identificada.....	4
1.3.	Beneficios Esperados	4
2.	Introducción	5
2.1.	Objetivo del documento	5
2.2.	Alcance técnico	5
2.3.	Audiencia objetivo	6
2.4.	Referencias técnicas	6
2.5.	Descripción general del sistema	6
2.6.	Viabilidad técnica.....	7
2.7.	Viabilidad económica.....	7
2.8.	Viabilidad operativa	7
2.9.	Viabilidad legal (protección de datos, normativa pública).....	¡Error! Marcador no definido.
3.	Descripción General	8
3.1.	Propósito del sistema	8
3.2.	Visión técnica del producto	9
3.3.	Tipos de usuarios y dispositivos de acceso	9
3.4.	Reglas generales del negocio (a nivel técnico).....	10
4.	Arquitectura del sistema.....	10
4.1.	Diagrama general	11
4.3.	Tecnologías seleccionadas.....	12
4.4.	Consideraciones de seguridad	12
5.	Interfaz de usuario y prototipos.....	12
5.1.	Principios de diseño de la UI	12
5.2.	Descripción general de pantallas	13
5.3.	Prototipos visuales	13
6.	Especificación de interfaces.....	13
6.1.	Interfaces internas (API RESTful).....	13
6.2.	Interfaces externas	13
6.3.	Validaciones y estructuras de datos esperadas	14
7.	Plan de pruebas	14
7.1.	Objetivo del plan.....	14
7.2.	Tipos de pruebas consideradas.....	14

7.3.	Criterios de aceptación simulados	15
7.4.	Revisión de posibles errores comunes	15
8.	Planificación del Proyecto	15
8.1.	Metodologías de Desarrollo	15
8.2.	Cronograma general y por fases	16
8.2.1.	Fase 1.....	16
8.2.2.	Fase 2: Diseño del sistema.....	17
8.2.3.	Fase 3: Desarrollo.....	17
8.2.4.	Fase 4: Pruebas y cierre	17
9.	Diagramas	18
9.1.	Diagrama de Flujo	18
9.2.	Diagrama de Clases	18
9.3.	Diagrama de secuencia	18
9.5.	Diagrama de Casos de Uso.....	18
10.	Consideraciones finales	 Error! Marcador no definido.

1. Resumen Ejecutivo

El proyecto GESCOMPH (Gestión Comercial Palermo Huila) tiene como objetivo diseñar una solución web especializada para la administración de arriendos de locales comerciales bajo la jurisdicción de la Alcaldía Municipal de Palermo, Huila. El sistema centraliza procesos relacionados con la **gestión contractual, control de ocupación, seguimiento de obligaciones y vencimientos y asignación de espacios arrendables** en plazas públicas.

La visión funcional del sistema consiste en establecer una plataforma modular, segura y extensible que facilite el acceso estructurado a la información relevante para los distintos perfiles de usuario (administradores, arrendatarios y ciudadanos), reduciendo las fricciones operativas actuales y fortaleciendo el control institucional sobre los activos arrendados.

1.1. Situación Actual

La Alcaldía de Palermo – Huila presenta deficiencias estructurales en la gestión administrativa de los arriendos de bienes inmuebles municipales destinados a actividades comerciales, especialmente en plazas de mercado y otros espacios públicos arrendables. El sistema actualmente en uso es una solución tecnológica obsoleta, con funcionalidades limitadas, una arquitectura monolítica poco flexible y sin capacidad de adaptación a los procesos actuales ni a las necesidades de seguimiento y trazabilidad de la información.

Esta desactualización tecnológica ha derivado en las siguientes problemáticas concretas:

- Falta de trazabilidad y control efectivo sobre contratos vigentes, vencidos o inactivos, dificultando la supervisión de los compromisos contractuales por parte de los arrendatarios.
- Ausencia de alertas o mecanismos automatizados de seguimiento, lo que incrementa el riesgo de vencimientos no gestionados y acumulación de incumplimientos sin intervención oportuna.
- Procesos manuales y duplicación de tareas, lo cual genera ineficiencias administrativas, posibles errores humanos y tiempos prolongados en la gestión de trámites.
- Imposibilidad de acceder a estadísticas consolidadas o en tiempo real, limitando la toma de decisiones estratégicas sobre ocupación de espacios, regularización del cumplimiento contractual y asignación de recursos.

- Falta de visibilidad para la ciudadanía respecto a locales disponibles, obstaculizando la transparencia del proceso de arriendo y disminuyendo la participación ciudadana.

En términos de valor esperado, GESCOMPH proporcionará a la administración municipal una herramienta adaptable a sus procesos internos, que posibilitará una gestión más eficiente, estructurada y basada en datos.

1.2. Problemática identificada

Esta situación ha derivado en diversas limitaciones operativas y de gestión que afectan el desempeño institucional. Entre los principales impactos se encuentran la dificultad para acceder a información consolidada y actualizada, un control insuficiente sobre procesos críticos de seguimiento contractual y operativo, la persistencia de tareas manuales propensas a errores y la ausencia de mecanismos eficientes para el seguimiento de obligaciones contractuales. Asimismo, la falta de canales digitales accesibles dificulta la interacción efectiva con la ciudadanía y limita la transparencia de los procesos administrativos.

1.3. Beneficios Esperados

La puesta en marcha del sistema GESCOMPH permitirá a la Alcaldía de Palermo – Huila contar con una solución informática para la gestión integral de bienes inmuebles en arriendo bajo dominio municipal. Esta herramienta, estructurada bajo una arquitectura N Capas y orientada a la trazabilidad documental y operativa, generará beneficios tangibles en tres niveles: operativo, administrativo y ciudadano.

Entre los principales beneficios se destacan:

- Automatización de procesos críticos: como el seguimiento a vencimientos y obligaciones contractuales, generación de alertas internas y comunicación con arrendatarios a través de recordatorios.
- Reducción del riesgo de errores humanos en el registro y seguimiento de información contractual o documental, gracias a la estructuración de formularios y validaciones.

- Facilitación de la interacción ciudadana mediante funcionalidades como la visualización de locales disponibles, solicitud de citas para inspección y consulta de contratos vigentes por parte de los arrendatarios.
- Fortalecimiento de la transparencia institucional, al garantizar que la información sobre espacios arrendables, contratos activos y ocupación esté documentada, actualizada y disponible para consulta interna.

2. Introducción

2.1. Objetivo del documento

El presente documento técnico tiene como propósito describir en detalle la propuesta de solución tecnológica para el desarrollo del sistema GESCOMPH (Gestión Comercial Palermo Huila). Se especifican los componentes arquitectónicos, tecnologías seleccionadas, patrones de diseño, modelos de datos, seguridad y aspectos técnicos relevantes que permitirán llevar a cabo la implementación de una solución robusta, escalable y mantenable. Este documento servirá como guía técnica para el equipo de desarrollo y como fuente de validación para los docentes, asesores e interesados institucionales del proyecto.

2.2. Alcance técnico

La solución propuesta contempla el desarrollo de un sistema de información accesible desde navegadores web y dispositivos móviles, que centraliza la gestión de plazas, establecimientos, contratos y citas para visitas a locales. La arquitectura del sistema se basa en un enfoque N-capas, organizada en los siguientes niveles:

- Capa Entity: Define las entidades del dominio y estructuras de datos del sistema.
- Capa Data: Contiene los mecanismos de acceso a datos, implementación de repositorios y configuración de la base de datos relacional.
- Capa Business: Implementa la lógica de negocio, validaciones y reglas operativas.
- Capa Web: Exposición de servicios a través de API REST y consumo desde el frontend Angular.

GESCOMPH será un sistema independiente, sin funcionalidades transaccionales ni integración con plataformas contables externas. No se contempla el uso de firma electrónica ni validaciones jurídicas de los contratos en esta fase.

2.3. Audiencia objetivo

Este documento está dirigido a los siguientes perfiles:

- Equipo de desarrollo: Como guía de implementación técnica, definición de arquitectura y componentes.
- Instructores y asesores académicos: Para revisión estructural, validación de decisiones técnicas y seguimiento del avance del proyecto.
- Cliente institucional (Alcaldía de Palermo): Para comprensión del enfoque técnico adoptado y evaluación de la viabilidad de la solución.
- Equipo de pruebas y mantenimiento: Para asegurar la consistencia técnica en futuras fases de soporte o evolución del sistema.

2.4. Referencias técnicas

Las siguientes fuentes técnicas y normativas fueron consultadas para definir la propuesta tecnológica:

- Angular 19 – Documentación oficial: <https://angular.io>
- ASP.NET Core (.NET 8) – <https://learn.microsoft.com/en-us/dotnet/>
- Entity Framework Core – <https://learn.microsoft.com/en-us/ef/core/>
- SQL Server – <https://learn.microsoft.com/en-us/sql/>

2.5. Descripción general del sistema

GESCOMPH (Gestión Comercial Palermo Huila) es un sistema de información orientado a digitalizar y centralizar la gestión administrativa relacionada con el arriendo de establecimientos comerciales ubicados en plazas públicas del municipio de Palermo, Huila. El sistema permite controlar y organizar la información correspondiente a plazas, establecimientos, contratos de arriendo y disponibilidad de locales.

GESCOMPH contempla tres perfiles de usuario: administradores, arrendatarios y ciudadanos no autenticados.

- Administradores: Gestión integral de plazas y establecimientos, creación y control de contratos, registro del estado contractual mensual, consulta de estadísticas de ocupación y generación de reportes administrativos.
- Arrendatarios: Consulta de contratos vigentes, acceso a un calendario de vencimientos/hitos contractuales, visualización del estado de cumplimiento y configuración de recordatorios internos.
- Ciudadanos no autenticados: Consulta de locales disponibles aplicando filtros y solicitud de citas para conocerlos, sujetas a confirmación por parte del administrador.

El sistema no contempla funcionalidades de procesamiento transaccional ni integración con plataformas de recaudo. La información relacionada con el cumplimiento de obligaciones es de carácter informativo para los arrendatarios.

GESCOMPH será accesible desde navegadores web en dispositivos de escritorio y móviles, mediante una interfaz responsive que garantice su correcta visualización y operatividad en diferentes formatos de pantalla.

2.6. Viabilidad técnica

GESCOMPH será desarrollado utilizando tecnologías modernas, robustas y compatibles con los requerimientos funcionales del sistema. El equipo cuenta con experiencia en herramientas como Visual Studio, .NET 8, Angular 20 y SQL Server, lo que asegura un desarrollo controlado y técnicamente viable. La arquitectura en N capas permite escalabilidad, facilidad de mantenimiento y reutilización de componentes.

2.7. Viabilidad económica

Dado que se utilizarán herramientas de código abierto y entornos de desarrollo gratuitos (Visual Studio Community, Angular CLI, SQL Server Express), los costos asociados al software son nulos o mínimos.

2.8. Viabilidad operativa

GESCOMPH refleja flujos de trabajo reales de la Alcaldía de Palermo. Los usuarios (administradores, arrendatarios y ciudadanos) pueden adaptarse fácilmente a sus funcionalidades debido a su diseño intuitivo y enfoque en tareas comunes. Además, el sistema prioriza la centralización de la información, la visibilidad de estados y vencimientos contractuales y la reducción de trámites presenciales. Se considera operativamente viable.

3. Descripción General

GESCOMPH (Gestión Comercial Palermo Huila) es un sistema de información orientado a digitalizar y centralizar la gestión administrativa relacionada con el arriendo de establecimientos comerciales ubicados en plazas públicas del municipio de Palermo, Huila. El sistema permite controlar y organizar la información correspondiente a plazas, establecimientos, contratos de arriendo y disponibilidad de locales.

GESCOMPH contempla tres perfiles de usuario: administradores, arrendatarios y ciudadanos no autenticados.

- Administradores: Gestión integral de plazas y establecimientos, creación y control de contratos, registro del estado contractual mensual, consulta de estadísticas de ocupación y generación de reportes administrativos.
- Arrendatarios: Consulta de contratos vigentes, acceso a un calendario de vencimientos/hitos, visualización del estado de cumplimiento y configuración de recordatorios internos.
- Ciudadanos no autenticados: Consulta de locales disponibles aplicando filtros y solicitud de citas para conocerlos, sujetas a confirmación por parte del administrador.

El sistema no contempla funcionalidades de procesamiento transaccional ni integración con plataformas de recaudo; la información de cumplimiento de obligaciones se presenta con carácter informativo para los arrendatarios.

GESCOMPH será accesible desde navegadores web en dispositivos de escritorio y móviles, mediante una interfaz responsive que garantice su correcta visualización y operatividad en diferentes formatos de pantalla.

3.1. Propósito del sistema

El propósito de GESCOMPH es proporcionar una solución tecnológica que permita a la Alcaldía de Palermo, Huila, gestionar de forma centralizada y controlada el arriendo de establecimientos comerciales ubicados en plazas públicas del municipio.

El sistema facilitará el registro, seguimiento y actualización de la información relacionada con plazas, locales, contratos de arrendamiento y estados contractuales, brindando a los funcionarios herramientas estructuradas para la toma de decisiones y la trazabilidad operativa.

Adicionalmente, permitirá a los arrendatarios acceder a su información contractual, visualizar vencimientos/hitos y configurar recordatorios internos, reduciendo la necesidad de trámites presenciales.

Por su parte, los ciudadanos interesados podrán consultar los locales disponibles y agendar citas para conocerlos, contribuyendo a una gestión más transparente, accesible y organizada del espacio público comercial.

3.2. Visión técnica del producto

- GESCOMPH es una aplicación desarrollada bajo una arquitectura en N capas, estructurada en:
- Entity: Define las clases del modelo de dominio (Local, Plaza, Contrato, Usuario, etc.).
- Data: Administra el acceso a datos mediante Entity Framework Core, incluyendo contextos, repositorios y configuraciones de mapeo.
- Business: Contiene la lógica de negocio del sistema (validaciones, reglas contractuales, vencimientos y alertas, control de ocupación, etc.).
- Web: Expone los servicios a través de una API REST en .NET y contiene el frontend desarrollado con Angular 19 para acceso vía navegador.

3.3. Tipos de usuarios y dispositivos de acceso

- Administrador (Gestor o secretario de Hacienda): Tendrá acceso completo a los módulos de infraestructura, contratos, usuarios y reportes.
- Arrendatario: Podrá consultar sus contratos activos, visualizar un calendario de vencimientos, configurar recordatorios y conocer su estado de cumplimiento.
- Ciudadano: Podrá consultar locales disponibles y agendar citas sin necesidad de autenticación.

3.4. Reglas generales del negocio (a nivel técnico)

- El sistema no incorpora funcionalidades transaccionales ni de cobranza.
- Un local no puede ser arrendado simultáneamente por más de un arrendatario; solo podrá estar en estado “Disponible” u “Ocupado”.
- Cada contrato está vinculado a un único arrendatario y establecimiento.
- El sistema deberá impedir la creación de contratos en conflicto (fechas superpuestas, locales ya ocupados).
- Las citas solicitadas por ciudadanos serán aprobadas o rechazadas por el administrador.
- Los arrendatarios no pueden ver la información de otros usuarios ni de locales ya ocupados.
- Los datos críticos (modificación de contratos, estados contractuales, usuarios) deberán dejar trazabilidad en logs para auditoría.
- El sistema no debe depender de sistemas externos para su operación; será autónomo.

4. Arquitectura del sistema

Enfoque arquitectónico (N capas: Entity, Data, Business, Web)

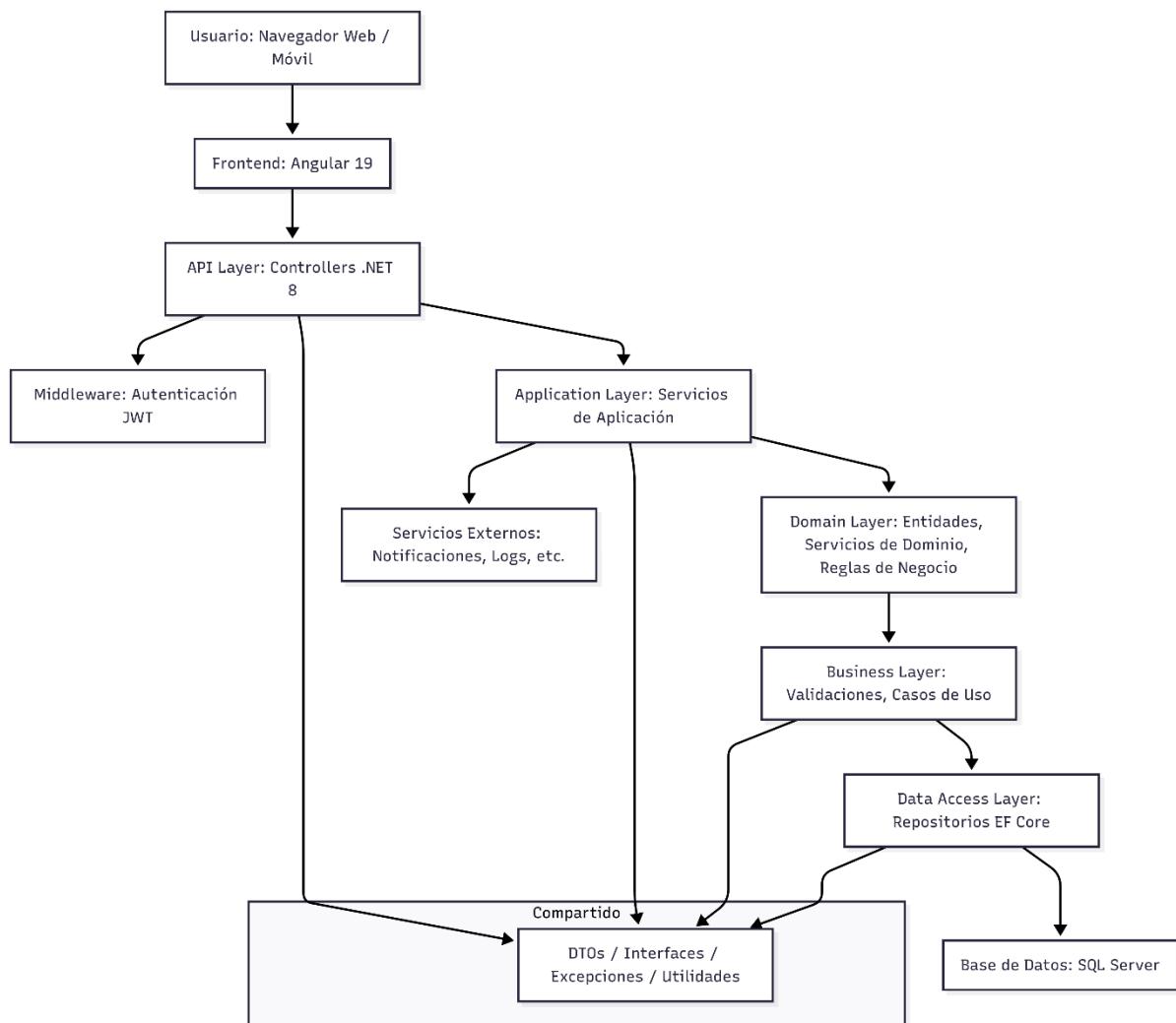
- El sistema GESCOMPH se construirá bajo una arquitectura en N capas para reforzar la separación de responsabilidades, la mantenibilidad y la escalabilidad.

Las capas definidas son:

- Entity Layer (Dominio): contiene las clases del modelo de dominio y reglas invariantes (por ejemplo, Establecimiento, Plaza, Contrato, Usuario). El dominio debe permanecer libre de dependencias de infraestructura.
- Data Layer (Persistencia): encapsula el acceso a datos y la configuración de la base relacional con Entity Framework Core (DbContext, mapeos/fluentes, migraciones y consultas optimizadas/proyectadas). Se recomienda evitar repositorios genéricos innecesarios porque DbContext ya implementa patrones UoW/Repository; usar repositorios específicos solo cuando aporten valor (consultas complejas o agregados del dominio).

- Business Layer (Aplicación): implementa casos de uso, orquestación y validaciones de negocio (p. ej., creación/actualización de contratos, control de ocupación, reglas de vencimientos). Expone interfaces consumidas por la capa Web y depende de abstracciones de Data (inyección de dependencias) para invertir dependencias.
- Web Layer (API/Presentación): expone servicios REST en ASP.NET Core y sirve al frontend Angular. No contiene lógica de negocio; realiza autorización/autenticación, DTO mapping, versionado, validación de entrada, paginación y manejo de errores estandarizado. El frontend Angular consume la API y sigue el style guide oficial (componentes presentacionales, servicios para acceso a datos/estado).

4.1. Diagrama general



4.2. Componentes y responsabilidades

Capa	Componentes principales	Responsabilidad Clave
Entity	Clases: Usuario, contrato, establecimiento. DbContext, Migraciones	Modelo de dominio
Data	Repositorios	Persistencia de datos
Business	Servicios, validaciones, lógica de negocios	Procesos internos
Web	Servicios HTTP, Rutas	Comunicación con el cliente

4.3. Tecnologías seleccionadas

Frontend: Angular 20, Angular Material, TypeScript

Backend: .NET 8, C#, Entity Framework Core

Base de Datos: SQL Server

Control de versiones: Git (GitHub)

Gestión del proyecto: Trello

4.4. Consideraciones de seguridad

Autenticación basada en JWT.

Control de acceso por roles: Administrador, Arrendatario, Ciudadano.

Validaciones en backend para entrada de datos.

Separación del frontend y backend en servicios desacoplados.

5. Interfaz de usuario y prototipos

5.1. Principios de diseño de la UI

El diseño de la interfaz de usuario seguirá los siguientes principios:

Consistencia visual: uso de paleta institucional, componentes reutilizables, y diseño modular basado en Material Design o Bootstrap.

Diseño responsive: adaptación automática a pantallas de escritorio, tablets y móviles.

Accesibilidad: cumplimiento básico de estándares WCAG 2.1 para acceso universal.

Navegación intuitiva: menús estructurados, flujos predecibles y mensajes de ayuda en tareas poco frecuentes.

Retroalimentación clara: alertas, spinners, tooltips y mensajes de validación comprensibles.

5.2. Descripción general de pantallas

Pantalla de Inicio: Información general del sistema, accesos directos para consulta de locales disponibles y agendamiento de citas.

Pantalla de Login: Acceso mediante credenciales para administradores y arrendatarios.

Dashboard del Administrador: Resumen de estadísticas, gestión de usuarios, plazas, contratos y citas.

Gestión de Plazas y Establecimientos: CRUD de plazas y locales comerciales disponibles o arrendados.

Gestión de Contratos: Creación, consulta y edición de contratos de arriendo.

Registro de Pagos: Opción para registrar manualmente el estado de pago mensual.

Calendario del Arrendatario: Vista de fechas de pago, deuda acumulada y recordatorios configurables.

Pantalla de Citas: Consulta pública de locales disponibles y solicitud de cita presencial.

Soporte: Formulario de contacto para incidencias técnicas o consultas.

5.3. Prototipos visuales

Para esta versión académica, se consideran prototipos simulados elaborados con herramientas como Figma o Balsamiq. No forman parte del entregable técnico, pero sirven como referencia visual para validar el flujo esperado:

Prototipo 1: Inicio y login

Prototipo 2: Dashboard administrador

Prototipo 3: Calendario del arrendatario

Prototipo 4: Pantalla de cita pública

6. Especificación de interfaces

6.1. Interfaces internas (API RESTful)

Las capas del sistema se comunican mediante interfaces bien definidas:

La Web Layer se comunica con la Business Layer a través de una API REST desarrollada con .NET 8.

Las rutas siguen convenciones RESTful (/api/usuarios, /api/contratos, etc.).

La API retorna datos en formato JSON.

Se emplean controladores para manejar peticiones HTTP y servicios inyectados (DI) para separar lógica.

6.2. Interfaces externas

Los usuarios acceden al sistema desde navegadores modernos (Chrome, Firefox, Edge).

Se garantiza compatibilidad cross-browser y diseño responsivo.
El sistema está optimizado para dispositivos móviles, tablets y equipos de escritorio.

6.3. Validaciones y estructuras de datos esperadas

Se aplican validaciones en frontend (Angular) y backend (C#).

Validaciones comunes incluyen: campos requeridos, formatos de correo, número de identificación, fechas válidas, y montos numéricos.

Las estructuras de datos se modelan en clases DTO (Data Transfer Object), para cada entidad:

ContratoDTO: incluye ID, nombre del arrendatario, fechas, valor, estado.

EstablecimientoDTO: ID, nombre, ubicación, estado (disponible/arrendado).

PagoDTO: contratoID, mes, valorPagado, estado, fechaRegistro.

Los errores se devuelven en objetos JSON con códigos HTTP estándar y mensajes claros para el usuario final.

7. Plan de pruebas

7.1. Objetivo del plan

El objetivo de este plan es establecer los criterios de verificación funcional y técnica del sistema GESCOMPH bajo un entorno simulado, asegurando que los módulos definidos cumplan con los requerimientos estipulados y que el comportamiento general del sistema sea coherente, estable y predecible, a pesar de que no se realizará un despliegue real.

7.2. Tipos de pruebas consideradas

Tipo de prueba	Descripción
Pruebas unitarias	Validación de métodos individuales de lógica de negocio (por ejemplo: cálculo de mora, validación de contrato, etc.).
Pruebas de integración	Verificación de que los módulos interactúan correctamente (por ejemplo: creación de contrato → asignación de establecimiento).
Pruebas de validación	Simulación de entrada de datos en formularios para asegurar el cumplimiento de reglas definidas.
Pruebas de interfaz (UI)	Revisión de la disposición y comportamiento visual esperado en cada pantalla.

Pruebas de usabilidad	Evaluación del flujo de navegación e interacción esperada por parte de usuarios tipo: administrador, arrendatario y ciudadano.
Pruebas de seguridad básica	Simulación de intentos de acceso no autorizado o ejecución de acciones fuera del rol permitido.

7.3. Criterios de aceptación simulados

Un módulo o funcionalidad se considerará aceptado si cumple con los siguientes criterios:

Cumple con todos los requisitos funcionales documentados.

Las validaciones de datos funcionan correctamente en frontend y backend.

El flujo de navegación permite completar las tareas sin errores.

El sistema responde con mensajes claros ante errores o acciones inválidas.

Las restricciones por rol están correctamente implementadas y no hay fugas de acceso.

El diseño UI es visualmente consistente, responsive y usable.

7.4. Revisión de posibles errores comunes

Durante la validación conceptual del sistema, se identificaron posibles errores comunes que se deben considerar en una futura implementación:

Errores de validación mal manejados: como permitir guardar un contrato con campos vacíos o sin validaciones cruzadas.

Falta de retroalimentación visual: ausencia de confirmaciones o errores que confundan al usuario.

Desalineación entre estados: ejemplo, un pago registrado no refleja correctamente el estado del contrato.

Problemas de navegación: botones sin funcionalidad o rutas incorrectas entre vistas.

Errores de permisos: usuarios accediendo a módulos restringidos por error en la lógica de roles.

Estos puntos se documentan con el fin de preverlos durante el desarrollo, incluso si el sistema no será desplegado en esta etapa académica.

8. Planificación del Proyecto

8.1. Metodologías de Desarrollo

El desarrollo del sistema GESCOMPH se realizará utilizando la metodología **Scrum**, un marco ágil de trabajo orientado a la entrega incremental de valor

funcional, mediante iteraciones cortas, enfocadas en la colaboración con el cliente y en la mejora continua del producto.

Scrum permite adaptar el proceso de desarrollo a los cambios razonables que surjan durante el ciclo del proyecto, manteniendo visibilidad del progreso y alineando las prioridades del equipo técnico con las necesidades reales de la Alcaldía de Palermo – Huila.

Principales características de la metodología aplicada:

- **Iteraciones cortas (sprints):** el proyecto se dividirá en ciclos de trabajo regulares (normalmente entre 2 y 4 semanas), al final de los cuales se entregarán incrementos funcionales del sistema.
- **Backlog priorizado:** los requerimientos funcionales y no funcionales se organizan en un backlog gestionado por el Product Owner, con revisiones frecuentes de prioridades.
- **Revisión y demostración continua:** al final de cada sprint se presentarán avances funcionales para revisión por parte de los usuarios clave, facilitando ajustes tempranos.
- **Adaptación continua:** tras cada entrega se realiza una retrospectiva del sprint para identificar oportunidades de mejora en el proceso de desarrollo.
- **Colaboración activa con el cliente:** los representantes institucionales participarán como *stakeholders activos*, revisando entregables, validando funcionalidades y aportando retroalimentación de valor.

El desarrollo teórico del sistema GESCOMPH se organiza en **cuatro fases principales**, siguiendo una metodología estructurada que permite simular el ciclo completo de vida del software. A continuación, se describe cada fase, sus actividades y entregables claves

8.2. Cronograma general y por fases

8.2.1. Fase 1

Objetivo: Definir claramente los requisitos funcionales, no funcionales y las condiciones técnicas y operativas del sistema.

Actividades clave:

- Levantamiento de requerimientos con el cliente.
- Definición de alcance funcional y técnico.
- Redacción del Documento de Requisitos del Software (SRS).
- Análisis de viabilidad técnica, económica y operativa.
- Revisión de tecnologías y arquitectura seleccionada.

Entregables:

- Documento SRS completo.
- Análisis de viabilidad.

- Cronograma base del proyecto.
- Estructura general del sistema y actores.

8.2.2. Fase 2: Diseño del sistema

Objetivo: Establecer la arquitectura técnica, los flujos lógicos y las interfaces de usuario del sistema.

Actividades clave:

- Diseño de la arquitectura en N capas (Entity, Data, Business, Web).
- Modelado de base de datos y relaciones entre entidades.
- Elaboración de diagramas (casos de uso, flujo, clases, secuencia).
- Creación de prototipos visuales y pantallas clave.
- Definición de interfaces internas (API REST) y validaciones.

Entregables:

- Documento de arquitectura técnica.
- Diagramas estructurales y de comportamiento.
- Prototipos visuales de la UI.
- Especificación de interfaces.

8.2.3. Fase 3: Desarrollo

Objetivo: Simular el desarrollo funcional del sistema según la arquitectura definida, sin llegar a desplegar el producto.

Actividades clave:

- Construcción de estructura base en Visual Studio y Angular.
- Implementación de modelos, controladores y servicios (estructura).
- Simulación de interacciones API (con ejemplos de endpoints).
- Documentación de flujos de negocio (crear contrato, registrar pagos, etc.).
- Validación interna del modelo técnico.

Entregables:

- Estructura funcional de carpetas y módulos (no ejecutable).
- Código estructurado de componentes clave (simulado).
- Registro de funcionalidades diseñadas.

8.2.4. Fase 4: Pruebas y cierre

Objetivo: Simular la validación del sistema a través de casos de prueba conceptuales y documentar los resultados esperados.

Actividades clave:

- Diseño de plan de pruebas teóricas por módulo.
- Revisión de criterios de aceptación y control de calidad.
- Simulación de errores comunes y respuestas esperadas.
- Documentación final técnica.

Entregables:

- Plan de pruebas documentado.
- Casos de prueba representativos por rol.
- Documento final de análisis técnico.
- Presentación de resultados (académica).

9. Diagramas

9.1. Diagrama de Flujo

9.1.1. *Diagrama de Flujo General Administrador*

9.1.2. *Diagrama de Flujo del Usuario*

9.1.3. *Diagrama de Flujo General del Arrendatario*

9.2. *Diagrama de Clases*

9.3. Diagrama de secuencia

9.3.1. *Diagrama General de Secuencia del Administrador*

9.3.2. *Diagrama General de Secuencias del Usuario*

9.3.3. *Diagrama General de Secuencia del Arrendatario*

9.4. *Diagrama General de Procesos*

9.5. *Diagrama de Casos de Uso*