| DUOC UC - Escuela de informática y telecomunicaciones |
| --- |
| Propuesta de Proyecto y Especificación de Requisitos de Software |
| *Proyecto: MacotaConectada* |
|  |
| **Revisión*: [01]*** |
| **[Seleccionar fecha]** |

| Planificación y Especificación de Requisitos según estándares; IEEE 830, ISO9000 y PMI. |
| --- |

**Contenido**

[*DUOC UC - Escuela de informática y telecomunicaciones 1*](#_heading=h.j7zovmy1hxml)

[**Ficha del documento 4**](#_heading=h.als98cln958f)

[**1. Introducción 5**](#_heading=h.udhkkwtnhrop)

[1.1. Propósito 5](#_heading=h.cy8tu7en1908)

[1.2. Ámbito del Sistema 5](#_heading=h.swt4f7yyqcb8)

[1.3. Definiciones, Acrónimos y Abreviaturas 6](#_heading=h.yw26bmyw2djm)

[1.4. Referencias 6](#_heading=h.em4x3d8v1x4e)

[1.5. Visión General del Documento 6](#_heading=h.vdjcrbqvy1bl)

[**2. Descripción General 6**](#_heading=h.m32dav3m6srw)

[2.1. Perspectiva del Producto 6](#_heading=h.37hwhpg0op3t)

[2.2. Funciones del Producto 7](#_heading=h.7yy9uj26sqsf)

[2.3. Características de los Usuarios 7](#_heading=h.7hcltbv4mvdc)

[2.4. Restricciones 7](#_heading=h.evbsnpeyrkdp)

[2.5. Suposiciones y Dependencias 7](#_heading=h.tsioyf2ok49k)

[2.6. Requisitos Futuros 8](#_heading=h.sqaww35fx3bk)

[**3. Requisitos Específicos 8**](#_heading=h.35up5r6z3fo)

[3.1 Requisitos comunes de las interfaces 8](#_heading=h.jr2pglblsftw)

[*3.1.1 Interfaces de usuario 8*](#_heading=h.xhjzvx6z705g)

[*3.1.2 Interfaces de hardware 8*](#_heading=h.8e9o19i86c4x)

[*3.1.3 Interfaces de software 9*](#_heading=h.w35yz9jz6hy0)

[3.2 Requisitos funcionales 9](#_heading=h.7uozh5olylgw)

[3.3 Requisitos no funcionales (Organización) y de calidad (Producto) 10](#_heading=h.hllddqlikpi)

[*3.3.1 Requisitos de Rendimiento o Eficiencia 10*](#_heading=h.71n2wcgu19xd)

[*3.3.2 Requisitos de Seguridad 10*](#_heading=h.skdadho5en06)

[*3.3.3 Requisitos de Usabilidad 11*](#_heading=h.s6d2c9y7zgj7)

[*3.3.4 Requisitos de Disponibilidad 11*](#_heading=h.cy8nq25g2qek)

[*3.3.5 Requisitos de Portabilidad 11*](#_heading=h.3zeeq0ro2408)

[*3.3.6 Requisitos de Mantenibilidad 11*](#_heading=h.nm8ibdj80wae)

[*3.3.7 Requisitos de Funcionalidad 11*](#_heading=h.ffaz3epdkip6)

[3.4 Requisitos No funcionales Organizacionales 11](#_heading=h.3ns0nmdolgcx)

[**4. Propuesta de Planificación 12**](#_heading=h.7ff23eqf9ep)

[4.1 Descripción general acerca de la Planificación 12](#_heading=h.w9e2eupzspla)

[*4.1.2 Definición del Equipo de Trabajo 12*](#_heading=h.yjzk0a37ef2t)

[*4.1.3 Definición de Actividades principales del Proyecto 12*](#_heading=h.mohi7fh5fyc8)

[*4.1.4 Resumen Costos del Desarrollo del Proyecto 12*](#_heading=h.sbw5r47nez5f)

[*4.1.6 Carta Gantt 12*](#_heading=h.mmp11gozlor0)

[5. Anexos 13](#_heading=h.eesoi7w4l5f3)

[*5.1 Acta de Proyecto 13*](#_heading=h.yzvyclby6ebc)

[*5.2 Matriz Especificación de Requerimientos 13*](#_heading=h.50zejxwt6gq2)

[*5.3 Prototipado de Software 13*](#_heading=h.6czcy0xqvbhq)

[*5.4 Matriz EDT. Planilla Detallada Cálculo de Esfuerzo 13*](#_heading=h.khhtw07448lx)

[*5.5 Planilla Carta Gantt 13*](#_heading=h.hbtpsfboa5d)

# Ficha del documento

| **Fecha** | **Revisión** | **Autor** | **Modificación** |
| --- | --- | --- | --- |
| *21/08/2025* | *1.0* | *Jaime Canales* | *Desarrollo hasta punto 2.6* |
| *31/08/2025* | *1.1* | *Jaime Canales* | *Desarrollo total* |
| *01/09/2025* | *1.2* | *Felipe Muñoz* | *Corrección de documento* |

Documento validado por las partes en fecha:

**Integrantes:**

| **Nombre Integrante del Equipo** | **Rol Definido** |
| --- | --- |
| *Jaime Canales* | *Jefe de Proyecto / Diseñador* |
| *Giovanni Garrido* | *Analista Programador* |
| *Felipe Muñoz* | *Administrador BD / Calidad Y Testing* |

# 1. Introducción

Este documento detalla la Especificación de Requisitos de Software(ERS) del sistema MascotaConectada, una aplicación web orientada a la gestión de la salud de mascotas.

La plataforma permitirá a los usuarios con rol de dueños registrar y consultar información médica de sus mascotas, mientras que los usuarios con rol de veterinarios podrán actualizar historiales clínicos y fichas de atención. Además, el sistema integrará notificaciones automáticas para recordar citas y vacunas, junto con la funcionalidad de geolocalización para identificar veterinarias cercanas.

Este enfoque permite a MascotaConectada centralizar la información relevante en un entorno digital accesible, facilitando la organización y cuidado de las mascotas tanto para sus dueños como para los profesionales.

## 1.1. Propósito

El propósito de este documento es establecer de manera formal y detallada los requisitos del sistema MascotaConectada. Este documento servirá como base para el diseño, desarrollo, validación y futura evolución de la aplicación web, asegurando que todas las partes interesadas tengan una visión clara y compartida de los objetivos y funcionalidades del sistema.

Asimismo, busca alinear las expectativas entre los usuarios finales (dueños de mascotas y veterinarias), el equipo de desarrollo, QA, administradores de bases de datos y el jefe de proyecto, estableciendo un marco común de trabajo que permita garantizar calidad, usabilidad y cumplimiento de los plazos definidos.

## 1.2. Ámbito del Sistema

MascotaConectada será una aplicación web que permitirá:

* Registro y autenticación de usuarios.
* Creación y gestión de fichas médicas de mascotas.
* Actualización de historiales clínicos por parte de veterinarios.
* Agenda médica con recordatorios automáticos (citas, vacunas, controles).
* Localización de veterinarias cercanas mediante geolocalización.
* Panel de administración para la gestión de usuarios y mascotas.

MascotaConectada no permitirá:

* Compras de alimentos y/o accesorios para mascotas.
* Agendamiento de consultas médicas.

## 1.3. Definiciones, Acrónimos y Abreviaturas

* **ERS:** Especificación de Requisitos de Software.
* **BD:** Base de datos.
* **QA:** Quality Assurance (Aseguramiento de calidad).
* **API:** Interfaz de Programación de Aplicaciones.
* **Geolocalización:** Determinación de la ubicación física de un usuario.

## 1.4. Referencias

[IEEE 830 – Recommended Practice for Software Requirements Specifications.](https://www.fdi.ucm.es/profesor/gmendez/docs/is0809/ieee830.pdf)

[Documentación oficial Django.](https://docs.djangoproject.com/es/5.2/)

[Google Maps API – Servicios de geolocalización.](https://developers.google.com/maps/documentation/geolocation/overview?hl=es-419)

[Acta de Constitución del Proyecto MascotaConectada.](https://docs.google.com/document/d/1y4Frbchdp3OMDvAhan7azTJScSy9CpbC/edit?usp=drive_link&ouid=103522799761880098077&rtpof=true&sd=true)

[Carta Gantt Proyecto MascotaConectada.](https://docs.google.com/spreadsheets/d/1Tpg-he_VeeiUawrz187olZsl3Zo9N8RR/edit?usp=drive_link&ouid=103522799761880098077&rtpof=true&sd=true)

## 1.5. Visión General del Documento

Este documento se estructura en las siguientes secciones:

**Descripción general:** Contexto, funciones principales, usuarios, restricciones y dependencias.

**Requisitos específicos:** Lista de requisitos funcionales y no funcionales.

**Planificación y anexos:** Información complementaria como EDT, prototipos, matriz de requerimientos y cronograma.

# 2. Descripción General

## 2.1. Perspectiva del Producto

MascotaConectada es una aplicación web de nueva creación que funcionará como un sistema independiente, pero con integración a servicios externos como Google Maps API para la geolocalización de clínicas u hospitales veterinarios. El producto está orientado a la centralización de información médica de mascotas, ofreciendo acceso a dueños y veterinarios en un entorno seguro y fácil de usar.

La aplicación será desarrollada con Python (Django) para la capa de backend, Angular en frontend, y MySQL como sistema de gestión de base de datos. Su arquitectura permitirá escalabilidad futura hacia aplicaciones móviles y de escritorio.

## 2.2. Funciones del Producto

El sistema ofrecerá las siguientes funcionalidades principales:

* **Gestión de usuarios:** registro, autenticación y administración de cuentas.
* **Gestión de mascotas:** creación y mantenimiento de fichas médicas.
* **Historial clínico:** registro y actualización de controles, vacunas y tratamientos, con acceso para veterinarios.
* **Agenda médica:** programación de citas, envío de recordatorios y alertas automáticas.
* **Geolocalización:** localización de veterinarias cercanas mediante integración con Google Maps API.
* **Panel administrativo:** gestión de usuarios, control de datos y reportes básicos.

## 2.3. Características de los Usuarios

**Dueños de mascotas:** usuarios con conocimientos básicos de informática, requieren una interfaz intuitiva y fácil navegación.

**Veterinarios:** profesionales que accederán a los historiales médicos de las mascotas, con privilegios para editar información clínica.

**Administradores del sistema:** usuarios con perfil técnico responsables de la gestión del sitema, seguridad, base de datos y soporte.

## 2.4. Restricciones

* El desarrollo debe realizarse utilizando Python (Django), Angular y MySQL.
* El sistema debe integrar Google Maps API para la geolocalización de veterinarias.
* El acceso al sistema estará restringido a navegadores modernos (Google Chrome, Mozilla Firefox, Microsoft Edge).
* Debe garantizarse la seguridad de datos sensibles (cumplimiento de la ley n°19.628, protección de datos personales en Chile).
* El sistema debe operar bajo arquitectura web inicialmente.

## 2.5. Suposiciones y Dependencias

* Los usuarios contarán con conexión a internet para acceder al sistema.
* La API de Google Maps estará disponible y operativa.
* Los veterinarios y usuarios finales se registrarán voluntariamente en la plataforma.
* El proyecto depende de la disponibilidad del equipo de desarrollo y recursos de infraestructura para el despliegue.

## 2.6. Requisitos Futuros

* Desarrollo de una aplicación móvil multiplataforma.
* Implementación de un módulo de comunidad para interacción entre dueños de mascotas (foros, consejos y experiencias).
* Integración con sistemas de gestión de clínicas veterinarias.
* Incorporación de recordatorios multicanal (Whatsapp y correo electrónico).

# 3. Requisitos Específicos

## 3.1 Requisitos comunes de las interfaces

Descripción detallada de todas las entradas y salidas del sistema de software.

### 3.1.1 Interfaces de usuario

El sistema contará con una interfaz gráfica orientada a la facilidad de uso, considerando usuarios con conocimientos técnicos básicos. Las características principales serán:

* Las pantallas serán presentadas en un color verde esmeralda, transmitiendo frescura, confianza y relación con el cuidado animal.
* El tamaño de la letra por defecto será 10 pt, garantizando legibilidad.
* Los botones de acción estarán diseñados en negrita, para resaltar su funcionalidad y facilitar la interacción.
* Navegación intuitiva con menús claros y accesibles.
* Diseño responsive que permita acceso desde dispositivos de escritorio, tablets y móviles.

### 3.1.2 Interfaces de hardware

El sistema no dependerá de hardware especializado, pero debe cumplir con las siguientes características:

* Servidor de aplicación: mínimo 4 CPU virtuales, 8 GB RAM, 50 GB de almacenamiento.
* Servidor de base de datos: mínimo 2 CPU virtuales, 4 GB RAM, 30 GB de almacenamiento.
* Dispositivo cliente: computadores de escritorio o portátiles con conexión a internet, procesador de al menos 1.6 GHz, 4 GB RAM y navegador actualizado (Google Chrome, Mozilla Firefox, Microsoft Edge).
* Configuración adicional: el sistema debe soportar acceso concurrente mínimo a 100 en su versión inicial, con posibilidad de escalar en infraestructura cloud.

### 3.1.3 Interfaces de software

El sistema MascotaConectada se integrará con los siguientes productos de software:

**Google Maps API.**

* Propósito: proporcionar la funcionalidad de geolocalización para identificar veterinarias cercanas al usuario.
* Definición del interfaz: intercambio de datos mediante solicitudes HTTP/HTTPS con respuestas en formato JSON.

**Servidor de correo electrónico.**

* Propósito: enviar notificaciones automáticas y recordatorios de citas y vacunas.
* Definición del interfaz: conexión mediante protocolo SMTP, mensajes en formato texto/HTML.

**Sistema de base de datos MySQL.**

* Propósito: almacenar la información de usuarios, mascotas, historiales clínicos y citas.
* Definición del interfaz: consultas SQL estructuradas y seguras (scripts de creación, actualización lectura y borrado).

**Framework Django y Angular.**

* Propósito: proveer la lógica de negocio (backend) y la capa de presentación (frontend).
* Definición del interfaz: comunicación a través de API RESTful con intercambio de datos en formato JSON.

## 3.2 Requisitos funcionales

| **[Nombre del Requerimiento]** | **Actores Usuarios Relacionados** | **[Descripción corta del requerimiento]** |
| --- | --- | --- |
| Registro de usuarios | Dueño de mascota | El sistema debe permitir crear cuentas de usuarios. |
| Recuperación de contraseña | Dueño de mascota | El sistema debe permitir recuperar la contraseña por correo. |
| Perfil de usuario | Dueño de mascota | Cada usuario debe tener un perfil editable con sus datos. |
| Registro de mascotas | Dueño de mascota | Cada mascota debe tener un perfil con información básica. |
| Historial médico | Dueño de mascota / Veterinaria | Se debe registrar vacunas, consultas y tratamientos. |
| Recordatorios automáticos | Dueño de mascota | El sistema debe generar alertas de vacunas o controles. |
| Agenda de citas | Dueño de mascota | El sistema debe agendar citas veterinarias. |
| Mapa de veterinarias | Dueño de mascota / Veterinaria | Mostrar veterinarias cercanas mediante geolocalización. |
| Búsqueda de veterinarias | Dueño de mascota | Filtrar veterinarias por ubicación, servicios o precios. |
| Perfil de mascota | Dueño de mascota | El sistema debe permitir registrar mascotas asociadas a un usuario. |
| Autorización de roles | Dueño de mascota / Veterinaria / Admin | Acceso diferenciado según el tipo de usuario. |
| Panel administrador | Admin | Permitir gestión de usuarios y mascotas desde un panel. |
| Registro de veterinarias | Veterinaria / Admin | Las clínicas podrán registrarse y ofrecer servicios. |
| Gestión de servicios veterinarios | Veterinaria | Cada clínica podrá publicar servicios disponibles. |
| Historial de visitas | Dueño de mascota / Veterinaria | Registrar visitas y resultados de consultas médicas. |
| Exportar historial | Dueño de mascota / Veterinaria | Permitir exportar historial médico a PDF. |
| Notificar por correo | Dueño de mascota | El sistema enviará correos recordatorios. |
| Soporte multicuenta | Dueño de mascota | Permitir que un usuario tenga varias mascotas registradas. |
| Registro de alergias | Dueño de mascota / Veterinaria | Permitir guardar condiciones especiales de salud. |
| Registro de vacunas | Dueño de mascota / Veterinaria | Control digital de vacunas con fecha y lote. |
| Registro de tratamientos | Dueño de mascota / Veterinaria | Permitir agregar desparacitaciones y medicamentos. |
| Subida de documentos | Dueño de mascota / Veterinaria | Adjuntar archivos como recetas o exámenes. |
| Reportes de salud | Dueño de mascota | Generar reportes básicos de la salud de la mascota. |
| Módulo de adopción | Dueño de mascota | Permitir publicar adopciones. |

## 3.3 Requisitos no funcionales (Organización) y de calidad (Producto)

### 3.3.1 Requisitos de Rendimiento o Eficiencia

| **[Nombre del Requerimiento]** | **Actores Usuarios Relacionados** | **[Descripción corta del requerimiento]** |
| --- | --- | --- |
| **Base de datos escalable** | **Sistema** | **Se debe usar MySQL optimizado para crecimiento futuro.** |
| **API REST** | **Sistema** | **Backend expuesto mediante API REST para escalabilidad.** |
| **Rendimiento aceptable** | **Todos** | **El sistema debe cargar en < 3 segundos por vista.** |

### 3.3.2 Requisitos de Seguridad

| **[Nombre del Requerimiento]** | **Actores Usuarios Relacionados** | **[Descripción corta del requerimiento]** |
| --- | --- | --- |
| **Autenticación segura** | **Dueño de mascota** | **El sistema debe contar con login seguro (contraseña encriptada).** |
| **Seguridad de datos** | **Todos** | **La información debe almacenarse cifrada en la base de datos.** |
| **Backup automático** | **Sistema** | **Copias de seguridad en la base de datos semanales.** |

### 3.3.3 Requisitos de Usabilidad

| **[Nombre del Requerimiento]** | **Actores Usuarios Relacionados** | **[Descripción corta del requerimiento]** |
| --- | --- | --- |
| **Interfaz responsiva** | **Dueño de mascota** | **La web debe adaptarse a móviles, tablets y pc.** |
| **Usabilidad básica** | **Todos** | **La interfaz debe ser clara y simple para usuarios no técnicos.** |
| **Accesibilidad** | **Todos** | **Cumplir buenas prácticas para usuarios con discapacidad.** |

### 3.3.4 Requisitos de Disponibilidad

| **[Nombre del Requerimiento]** | **Actores Usuarios Relacionados** | **[Descripción corta del requerimiento]** |
| --- | --- | --- |
| **Disponibilidad** | **Todos** | **El sistema debe estar disponible el 95% del tiempo.** |

### 3.3.5 Requisitos de Portabilidad

| **[Nombre del Requerimiento]** | **Actores Usuarios Relacionados** | **[Descripción corta del requerimiento]** |
| --- | --- | --- |
| **Compatibilidad cross-browser** | **Todos** | **Funcionar en Chrome, Firefox, Edge y Safari.** |

### 3.3.6 Requisitos de Mantenibilidad

| **[Nombre del Requerimiento]** | **Actores Usuarios Relacionados** | **[Descripción corta del requerimiento]** |
| --- | --- | --- |
| **Manual de usuario** | **Dueño de mascota / Veterinaria** | **Entregar documentación clara del uso de la plataforma.** |

### 3.3.7 Requisitos de Funcionalidad

| **[Nombre del Requerimiento]** | **Actores Usuarios Relacionados** | **[Descripción corta del requerimiento]** |
| --- | --- | --- |
| **Integración con Google Maps** | **Sistema** | **Usar Google Maps API para geolocalización.** |
| **Validación de datos** | **Sistema** | **Todos los formularios deben validar datos antes de guardarlos.** |

## 3.4 Requisitos No funcionales Organizacionales

1. El sistema debe cumplir con la identidad visual institucional del patrocinador, utilizando su logotipo .

# 4. Propuesta de Planificación

## 4.1 Descripción general acerca de la Planificación

EL proyecto MascotaConectada se abordará mediante un ciclo de desarrollo incremental, en el que se contemplan fases de planificación, análisis, diseño, desarrollo, pruebas, implementación, etc.

El trabajo se extenderá desde el 12 de Agosto hasta el 02 de Diciembre del 2025, con un equipo multidisciplinario compuesto por un jefe de proyecto, dos analistas programadores, un diseñador, un administrador de base de datos y un QA.

Para su correcto desarrollo se aplicarán buenas prácticas de ingeniería de software:

* Uso de control de versiones (GIT/GitHub).
* Revisiones de código y pruebas periódicas para asegurar la calidad.
* Aplicación de estándares de seguridad y cumplimiento con la ley 19.628 sobre la protección de datos personales en Chile.
* Comunicación constante entre el equipo (reuniones de seguimiento semanales).

### 4.1.2 Definición del Equipo de Trabajo

| **Rol** | **Responsabilidades principales** |
| --- | --- |
| Jefe de Proyecto | Liderar el proyecto, gestionar plazos, costos, riesgos y comunicación con patrocinador. |
| Analistas Programadores | Desarrollar los módulos de la plataforma web, implementar backend y frontend. |
| Diseñador | Diseñar mockups, interfaz gráfica, experiencia de usuario (UX/UI). |
| Administrador de Base de Datos | Diseñar, implementar y optimizar la base de datos; asegurar integridad y respaldos. |
| QA | Planificar y ejecutar pruebas de software, verificar cumplimiento de requisitos funcionales y no funcionales. |

### 4.1.3 Definición de Actividades principales del Proyecto

1. **Planificación** (Organización del equipo, Acta de Constitución, Aprobación).
2. **Análisis y Diseño** (Casos de uso, Documento de arquitectura, Mockups, ERS).
3. **Desarrollo** (Ambiente de desarrollo, scripts BD, módulos de usuarios, mascotas, historial, agenda, geolocalización).
4. **Pruebas y QA** (Ambiente de pruebas, pruebas funcionales, pruebas con usuarios).
5. **Implementación y Cierre** (Migración a producción, capacitaciones, manuales, acta de cierre).
6. **Entrega final** (presentación, documentación y entrega del proyecto completo).

### 4.1.4 Resumen Costos del Desarrollo del Proyecto

* Costos por FASE

| **Fase** | **Costo estimado** |
| --- | --- |
| Planificación | $2.500.000 |
| Análisis y Diseño | $4.000.000 |
| Desarrollo | $15.000.000 |
| Pruebas y QA | $4.500.000 |
| Implementación y Cierre | $2.500.000 |
| Entrega Final | $1.500.000 |

* Costos por Actor o Rol

| **Rol** | **Costo estimado** |
| --- | --- |
| Jefe de Proyecto | $3.500.000 |
| Analistas Programadores | $14.000.000 |
| Diseñador | $2.500.000 |
| Administrador de BD | $4.500.000 |
| QA | $3.500.000 |
| Infraestructura y Servicios (Hosting, dominio, SSL, nube) | $2.000.000 |

### 4.1.6 Carta Gantt

Inicio de proyecto: 12 de Agosto de 2025.

* **Inicio del Proyecto:** 12 de agosto de 2025.
* **Fase de Planificación:** 12 de agosto – 16 de agosto de 2025.
* **Análisis y Diseño:** 17 de agosto – 2 de septiembre de 2025.
* **Desarrollo (Parte 1):** 3 de septiembre – 14 de octubre de 2025.
* **Desarrollo (Parte 2):** 15 de octubre – 24 de octubre de 2025.
* **Pruebas y QA:** 25 de octubre – 8 de noviembre de 2025.
* **Implementación y Cierre:** 9 de noviembre – 18 de noviembre de 2025.
* **Entrega Final:** 19 de noviembre – 2 de diciembre de 2025.
* **Término del Proyecto:** 2 de diciembre de 2025.

## 5. Anexos

### 5.1 Acta de Proyecto

[Acta de Constitución del Proyecto.](https://docs.google.com/document/d/1y4Frbchdp3OMDvAhan7azTJScSy9CpbC/edit?usp=drive_link&ouid=103522799761880098077&rtpof=true&sd=true)

### 5.2 Matriz Especificación de Requerimientos

[Matriz de Requerimientos.](https://docs.google.com/spreadsheets/d/14trAQPH5ZIg9kZyz7lHF3uNwGCWO_B0L/edit?usp=drive_link&ouid=103522799761880098077&rtpof=true&sd=true)

### 5.3 Prototipado de Software

[Mockups de las interfaces de usuario del Sistema.](https://docs.google.com/document/d/1fPtjswX8VupLggZDsoAbBgNHtYJSmvJo/edit?usp=drive_link&ouid=103522799761880098077&rtpof=true&sd=true)

### 5.4 Matriz EDT. Planilla Detallada Cálculo de Esfuerzo

[Matriz EDT.](https://docs.google.com/spreadsheets/d/1HLPqn0HE8Ma9YXK6nBAdqCijMP_MKwp7KgcRvcYzXFc/edit?usp=sharing)

### 5.5 Planilla Carta Gantt

[Carta Gantt.](https://docs.google.com/spreadsheets/d/1Tpg-he_VeeiUawrz187olZsl3Zo9N8RR/edit?usp=drive_link&ouid=103522799761880098077&rtpof=true&sd=true)