|  |
| --- |
| http://www.duoc.cl/sites/default/files/logo_summit_0.png |
| Especificación de Requerimientos del Software |
| *Proyecto: Junta360 Digital* |
|  |
| **Revisión*: [1.0]*** |
| **[20.08.2025]** |

|  |
| --- |
| **ISO/EIC/IEEE 29148 de Ingeniería de Requisitos** |

Tabla de Contenidos

Contenido

[Ficha del documento 4](#_Toc209525181)

[1. Introducción 5](#_Toc209525182)

[1.1. Propósito del Documento 5](#_Toc209525183)

[1.2. Ámbito del Sistema 5](#_Toc209525184)

[1.3. Definiciones, Acronimos y Abreviaturas 5](#_Toc209525185)

[1.3.1. Perspectiva del producto 6](#_Toc209525186)

[1.3.2. Funciones del producto 6](#_Toc209525187)

[1.3.3. Características del usuario 6](#_Toc209525188)

[1.3.4. Limitaciones 6](#_Toc209525189)

[1.4. Referencias 7](#_Toc209525190)

[1.5. Visión General del Documento 7](#_Toc209525191)

[2. Descripción General 7](#_Toc209525192)

[2.1. Perspectiva del Producto 7](#_Toc209525193)

[2.2. Funciones del Producto 7](#_Toc209525194)

[2.3. Características de los usuarios 8](#_Toc209525195)

[2.4. Restricciones 8](#_Toc209525196)

[2.5. Suposiciones y Dependencias 8](#_Toc209525197)

[2.6. Requisitos Futuros 8](#_Toc209525198)

[3. Requisitos Específicos 9](#_Toc209525199)

[3.1. Requisitos de las Interfaces 9](#_Toc209525200)

[3.1.1. Interfaces de Usuario 9](#_Toc209525201)

[3.1.2. Interfaces de Hardware 9](#_Toc209525202)

[3.1.3. Interfaces de Software 10](#_Toc209525203)

[3.1.4. Interfaces de Comunicación 10](#_Toc209525204)

[3.2. Requisitos Funcionales 11](#_Toc209525205)

[3.3. Requisitos no Funcionales 13](#_Toc209525206)

[3.3.1. Requisitos de Rendimiento 13](#_Toc209525207)

[3.3.2. Seguridad 14](#_Toc209525208)

[3.3.3. Fiabilidad 14](#_Toc209525209)

[3.3.4. Disponibilidad 14](#_Toc209525210)

[3.3.5. Mantenibilidad 14](#_Toc209525211)

[3.4. Otros requisitos 15](#_Toc209525212)

[3.5. Requisitos de Base de Datos Lógica. 15](#_Toc209525213)

[4. Propuesta de Planificación 17](#_Toc209525214)

[4.1. Descripción general acerca de la planificación 17](#_Toc209525215)

[4.1.1. Definición del Equipo de Trabajo 17](#_Toc209525216)

[4.1.2. Definición de Actividades principales del Proyecto 18](#_Toc209525217)

[4.1.3. Diagrama EDT 19](#_Toc209525218)

[4.1.4. Carta Gantt 20](#_Toc209525219)

[4.1.5. Resumen de Costos del Desarrollo del Proyecto 21](#_Toc209525220)

[4.2. Plan de control de Cambio 22](#_Toc209525221)

[5. Anexos 22](#_Toc209525222)

# Ficha del documento

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Modificación** |
| |  | | --- | | *22-09-2025* |  |  | | --- | |  | | *1.0* | *Alexander Chamorro* | *Creación de documento* |
|  |  |  |  |

# Introducción

## Propósito del Documento

El presente documento tiene como objetivo principal **definir, de forma clara y detallada, los requisitos funcionales y no funcionales** para el desarrollo del software "Junta360 Digital". Su finalidad es servir como una guía de referencia fundamental para todas las partes interesadas en el proyecto: el equipo de desarrollo, el equipo de control de calidad (QA), el cliente (junta de vecinos) y los futuros administradores del sistema. Este ERS asegura una comprensión unificada del producto a construir y establece una base para las actividades de diseño, implementación y pruebas.

## Ámbito del Sistema

El sistema "Junta360 Digital" se concibe como una solución tecnológica integral para la **gestión interna y la comunicación externa de las juntas de vecinos en Chile**. El alcance del proyecto incluye:

* **Gestión de usuarios:** Registro, autenticación y manejo de perfiles para los vecinos y el directorio.
* **Gestión de trámites:** Automatización y seguimiento de solicitudes de certificados de residencia y postulaciones a proyectos.
* **Comunicación:** Sistema de notificaciones (email, WhatsApp) y publicación de noticias y avisos.
* **Organización:** Herramienta para la gestión de actividades y la reserva de espacios comunitarios.

El sistema **no cubrirá** la integración con sistemas de pago en línea ni la integración profunda con redes sociales, más allá de la compartición de enlaces. Se espera que el sistema mejore la **eficiencia operativa**, fomente la **participación ciudadana** y centralice la información de la comunidad.

## Definiciones, Acronimos y Abreviaturas

* **ERS:** Especificación de Requerimientos del Software.
* **QA:** Quality Assurance (Aseguramiento de Calidad).
* **UI:** User Interface (Interfaz de Usuario).
* **BD:** Base de Datos.
* **API:** Application Programming Interface (Interfaz de Programación de Aplicaciones).
* **TI:** Tecnologías de la Información.
* **Junta de Vecinos:** Organización comunitaria de carácter territorial.
* **Directorio:** Grupo de personas responsables de la gestión de la junta de vecinos.

### Perspectiva del producto

El software **"Junta360 Digital"** será una **aplicación web** accesible a través de navegadores modernos en dispositivos de escritorio y móviles (diseño **responsivo**). Se implementará como un sistema independiente, conectado a su propia **base de datos**, y no dependerá de un sistema anfitrión o de otro software para su operación, aunque sí interactuará con servicios de terceros para funcionalidades específicas como el envío de correos.

### Funciones del producto

El producto proporcionará las siguientes funcionalidades clave para sus usuarios:

* **Funciones de Gestión:** Inscripción de vecinos, solicitud de certificados, y postulación a proyectos vecinales.
* **Funciones de Comunicación:** Envío de notificaciones y publicación de noticias.
* **Funciones de Organización:** Gestión de actividades, eventos y reserva de recursos comunitarios.
* **Funciones Administrativas:** Revisión y aprobación de solicitudes, gestión de usuarios y generación de reportes.

### Características del usuario

El sistema está diseñado para dos perfiles de usuario principales:

* **Vecinos (Usuarios finales):** Requieren una **interfaz intuitiva y de fácil navegación**, con conocimientos informáticos básicos. Su objetivo es realizar trámites y estar informados sin complicaciones técnicas.
* **Directorio (Usuarios administrativos):** Necesitan un **panel de administración simple y robusto**, con una curva de aprendizaje mínima, que les permita gestionar la comunidad de manera eficiente sin necesidad de ser expertos en TI. La usabilidad es un requisito crítico para este grupo.

### Limitaciones

* **Tecnológicas:** El sistema debe ser accesible a través de navegadores web y dispositivos móviles estándar.
* **Operativas:** La funcionalidad de trámites como solicitudes de certificados y postulación a proyectos debe incluir un proceso de revisión y aprobación.
* **Seguridad:** El sistema debe garantizar la privacidad de los datos personales de los vecinos.
* **Económicas:** Se debe considerar un costo de desarrollo y mantenimiento que sea accesible para una organización comunitaria.

# Referencias

En esta subsección se mostrará una lista completa de todos los documentos referenciados en la ERS.

# Visión General del Documento

Este documento de ERS está organizado en cinco secciones principales. La **Introducción** presenta el contexto y propósito del proyecto. La **Descripción General** (sección 2) detalla el ámbito y las características de los usuarios. La sección 3, **Requisitos Específicos**, enumera los requerimientos funcionales y no funcionales del software. La sección 4, **Propuesta de Planificación**, describe el enfoque metodológico y los recursos. Finalmente, la sección 5, **Anexos**, incluirá documentos complementarios como la matriz de trazabilidad de requisitos y otros.

# Descripción General

Esta sección describe los factores principales que contextualizan el sistema y sus requisitos. Ofrece una visión de alto nivel del producto y de los elementos que influyen en su desarrollo, facilitando una comprensión más profunda de los requisitos específicos que se detallarán en las secciones posteriores.

## Perspectiva del Producto

El sistema Junta360 Digital será una aplicación independiente. Se conectará a una base de datos para almacenar toda la información. No se integrará con sistemas externos complejos de pago o redes sociales, aunque sí permitirá compartir contenido a través de enlaces básicos a plataformas como WhatsApp.

## Funciones del Producto

Las principales funciones del sistema son:

* Gestión de usuarios: Registro, perfiles y roles (directorio y vecinos).
* Gestión de trámites: Manejo de solicitudes de certificados de residencia y postulación a proyectos vecinales.
* Comunicación: Envío de notificaciones (por correo, WhatsApp y afiches digitales) y publicación de noticia
* Organización comunitaria: Gestión de eventos, actividades y reserva de espacios (canchas, salas, plazas).
* Administración interna: Herramientas para que el directorio pueda revisar, aprobar o rechazar trámites, gestionar inscripciones y visualizar reportes.

## Características de los usuarios

Se definen dos tipos principales de usuarios:

* **Usuarios finales (vecinos):** Personas mayores de 14 años que residen en la unidad territorial. Tienen conocimientos básicos de navegación web o uso de aplicaciones móviles. Buscan un sistema **intuitivo y fácil de usar** para realizar trámites, informarse y participar en la comunidad.
* **Usuarios administrativos (directorio):** Miembros de la junta de vecinos encargados de la gestión. Deben poder utilizar el sistema sin necesidad de tener **conocimientos informáticos avanzados**, por lo que la interfaz de administración debe ser clara y sencilla (un requisito de usabilidad clave).

## Restricciones

Se identifican las siguientes limitaciones que influirán en el desarrollo del sistema:

* **Hardware:** El sistema debe funcionar en un entorno de hardware genérico, accesible a través de navegadores web en dispositivos modernos.
* **Usabilidad:** Se requiere una interfaz de usuario sencilla y accesible, especialmente para el personal administrativo, que puede no tener experiencia técnica.
* **Seguridad:** Es fundamental proteger la información personal de los vecinos y garantizar la integridad de los datos almacenados.
* **Comunicación:** La capacidad de enviar notificaciones por WhatsApp dependerá de la correcta integración con la API oficial de la plataforma, lo cual puede implicar costos adicionales.

## Suposiciones y Dependencias

**Suposiciones**: Se asume que los usuarios finales tendrán acceso a una conexión a Internet estable y a un dispositivo compatible. También se presupone que la junta de vecinos proporcionará la información necesaria para la configuración inicial del sistema y para el envío de notificaciones.

**Dependencias**: El funcionamiento del sistema depende de la disponibilidad de un servidor web y de una base de datos. La funcionalidad de envío de notificaciones por email depende de un servicio de correo electrónico externo.

## Requisitos Futuros

En una fase posterior al desarrollo inicial, se podrían considerar las siguientes mejoras:

* **Integración de Pagos:** Habilitar el procesamiento de cuotas vecinales o pagos por el uso de espacios a través de plataformas en línea.
* **Mayor Integración:** Conexión con sistemas de redes sociales para una difusión más amplia de eventos y noticias.
* **Análisis Avanzado:** Implementación de reportes y estadísticas más detallados para el directorio, permitiendo un análisis más profundo de la participación y las actividades.
* **Asistente por Whatsapp:** mensajes automáticos, por usuarios que necesiten información de forma rápida mediante **Whatsapp,** proporcionando información consultada por los vecinos sobre los eventos de la comunidad.

# Requisitos Específicos

En esta sección tienes que definir los requisitos con un nivel de detalle suficiente para que el equipo de desarrollo y el equipo de pruebas puedan trabajar con ellos. Deben ser claros, unívocos y, lo más importante, verificables.

## Requisitos de las Interfaces

Aquí describirás cómo interactúa el sistema con el usuario y con otros componentes de hardware y software.

## Interfaces de Usuario

* + Diseño: Deberá ser limpio, moderno y responsivo, adaptándose a pantallas de distintos tamaños. Se debe utilizar una paleta de colores coherente y una tipografía legible.
  + Navegación: La navegación debe ser intuitiva, con un menú principal claro que permita a los usuarios (vecinos y directorio) acceder rápidamente a las secciones clave (Inicio, Noticias, Actividades, etc.).
  + Accesibilidad: Debe considerar a usuarios con diferentes capacidades (por ejemplo, con un buen contraste de colores y tamaños de texto ajustables).

## Interfaces de Hardware

Se especifica los componentes físicos necesarios para usar el sistema.

* Dispositivos de entrada: El sistema debe ser operable con un teclado y un ratón, así como con pantallas táctiles para dispositivos móviles.
* Dispositivos de salida: La visualización se hará en monitores, pantallas de laptops, tablets y smartphones. Se requiere una resolución mínima para una experiencia de usuario óptima.
* Servidor: El servidor debe tener la capacidad de almacenamiento y procesamiento suficientes para soportar la base de datos y el tráfico de usuarios esperado.

## Interfaces de Software

Se describe cómo el sistema interactúa con otros programas.

* **Servidor de Aplicaciones:** El sistema se ejecutará sobre un servidor web (por ejemplo, Apache, Nginx o un servicio en la nube como AWS o Azure).
* **Base de Datos:** Se conectará a una base de datos relacional (como MySQL o PostgreSQL) para almacenar la información de usuarios, trámites, noticias, etc.
* **API:** La comunicación entre el frontend (la interfaz de usuario) y el backend (la lógica del servidor) se hará a través de una **API REST**.
* **Servicio de Correo:** El sistema debe integrarse con un servicio de email (n8n para el envío automático de notificaciones a los usuarios.
* **Servicio de Mensajería:** Se utilizará la **API de WhatsApp Business** para el envío de notificaciones.

### Interfaces de Comunicación

Describe los protocolos de comunicación.

* **Protocolo:** El sistema utilizará **HTTP/HTTPS** para una comunicación segura a través de Internet.
* **Formato de datos:** La información se transmitirá en formato **JSON** a través de la API REST.

## Requisitos Funcionales

Los requisitos funcionales, que definen las acciones que el software debe realizar, se detallan a continuación. Cada requisito está identificado por un código único (R-N°), un nombre descriptivo y una breve descripción de su funcionalidad.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| [R-N°] | [Nombre del Requerimiento] | Tipo Requerimiento [Funcional, No Funcional] | Actores Relacionados | [Descripción corta del requerimiento] |
| R-1 | Registro de usuarios | Funcional | Usuario Navega, Usuario Registrado | El sistema debe permitir a un visitante registrarse como vecino ingresando sus datos. |
| R-2 | Inicio de sesión | Funcional | Usuario Registrado, Vecino Aliado, Directivo | Permitir el acceso con credenciales para funciones avanzadas según el rol. |
| R-3 | Perfil de usuario | Funcional | Usuario Registrado, Vecino Aliado | Cada usuario registrado debe poder visualizar y editar sus datos personales. |
| R-4 | Gestión de roles | Funcional | Directivo | El directivo asigna y gestiona los roles de usuarios (vecino, directivo). |
| R-5 | Solicitud de certificado de residencia | Funcional | Usuario Registrado | El sistema debe permitir solicitar certificados de residencia en línea. |
| R-6 | Emisión de certificado | Funcional | Directivo | El directivo valida y emite certificados, enviándolos al correo del solicitante. |
| R-7 | Postulación a proyectos vecinales | Funcional | Vecino Aliado | Los vecinos aliados pueden postular proyectos vecinales con documentación adjunta. |
| R-8 | Revisión de proyectos | Funcional | Directivo | El directivo aprueba o rechaza proyectos, notificando al postulante vía email. |
| R-9 | Solicitud de uso de espacios públicos | Funcional | Usuario Registrado, Vecino Aliado | Permitir solicitar salas, canchas o plazas mediante un calendario de reservas. |
| R-10 | Gestión de reservas | Funcional | Directivo | Aprobar, rechazar y organizar las reservas de espacios comunitarios. |
| R-11 | Inscripción a actividades vecinales | Funcional | Usuario Registrado, Vecino Aliado | Los usuarios pueden inscribirse en actividades mientras haya cupos disponibles. |
| R-12 | Control de cupos e inscritos | Funcional | Sistema, Directivo | El sistema debe gestionar automáticamente los cupos y la lista de inscritos. |
| R-13 | Publicación de noticias | Funcional | Directivo | Permitir al directivo publicar noticias en el portal. |
| R-14 | Visualización de noticias | Funcional | Usuario Navega, Registrado, Vecino Aliado | Todos los usuarios pueden ver noticias publicadas. |
| R-15 | Envío de notificaciones | Funcional | Directivo | Enviar avisos por email, WhatsApp o afiche digital a los habitantes. |
| R-16 | Recepción de notificaciones | Funcional | Usuario Registrado, Vecino Aliado | Los usuarios deben recibir las notificaciones enviadas por la junta. |
| R-17 | Panel de gestión | Funcional | Directivo | Panel administrativo para gestionar usuarios, solicitudes y proyectos. |
| R-18 | Reportes de gestión | Funcional | Directivo | Generar reportes sobre vecinos inscritos, proyectos aprobados y reservas. |
| R-19 | Recuperación de contraseña | Funcional | Usuario Registrado, Vecino Aliado, Directivo | Permitir recuperar la contraseña por email. |
| R-20 | Contacto y soporte | Funcional | Usuario Navega, Registrado | El sistema debe permitir enviar mensajes de contacto o soporte. |

## Requisitos no Funcionales

Estos requisitos definen los criterios de calidad y las restricciones del sistema.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| [R-N°] | [Nombre del Requerimiento] | Tipo Requerimiento [Funcional, No Funcional] | Actores Relacionados | [Descripción corta del requerimiento] |
| R-21 | Seguridad de datos | No funcional | Sistema | El sistema debe almacenar contraseñas encriptadas y proteger datos sensibles. |
| R-22 | Disponibilidad | No funcional | Sistema | El sistema debe estar disponible el 99% del tiempo, salvo mantenimientos programados. |
| R-23 | Rendimiento | No funcional | Sistema | El sistema debe responder a consultas en menos de 2 segundos en condiciones normales. |
| R-24 | Escalabilidad | No funcional | Sistema | El sistema debe soportar el crecimiento de usuarios y solicitudes en el tiempo. |
| R-25 | Usabilidad | No funcional | Todos los actores | La interfaz debe ser responsiva, intuitiva y fácil de usar en PC y móvil. |
| R-26 | Compatibilidad | No funcional | Sistema | Debe funcionar en navegadores modernos y dispositivos móviles. |
| R-27 | Confidencialidad | No funcional | Sistema, Directivo | La información de los vecinos solo podrá ser accedida por usuarios autorizados. |
| R-28 | Mantenibilidad | No funcional | Equipo de desarrollo | El código debe ser modular y documentado para facilitar mantenimiento. |
| R-29 | Interoperabilidad | No funcional | Sistema | El sistema debe permitir integración futura con otros sistemas (ej: municipalidad). |

### Requisitos de Rendimiento

* **Velocidad:** Las páginas deben cargar en menos de 3 segundos con una conexión a Internet promedio.
* **Concurrencia:** El sistema debe ser capaz de soportar hasta 50 usuarios simultáneos sin degradación significativa del rendimiento.

### Seguridad

* **Autenticación:** Todos los usuarios deben autenticarse con un nombre de usuario y una contraseña segura.
* **Autorización:** El acceso a las funcionalidades administrativas (panel del directorio) debe estar restringido solo a los usuarios con el rol adecuado.
* **Protección de datos:** Toda la información sensible, como contraseñas, debe ser encriptada. La comunicación entre el cliente y el servidor debe usar HTTPS.

### Fiabilidad

* **Disponibilidad:** El sistema debe estar disponible el 99.5% del tiempo.
* **Recuperación de errores:** En caso de un fallo, el sistema debe recuperarse automáticamente y sin pérdida de datos en un tiempo máximo de 15 minutos.

### Disponibilidad

* El sistema debe garantizar una **disponibilidad del 99%** del tiempo, excluyendo las ventanas de mantenimiento programado. Esto significa que el tiempo de inactividad no planificado anual no debe exceder de aproximadamente 3 días y 15 horas.
* Se implementarán procedimientos de monitoreo para detectar fallos y tiempos de respuesta lentos de manera proactiva.

### Mantenibilidad

El código fuente del sistema debe ser **modular y estar bien documentado**, lo que facilitará al equipo de desarrollo realizar correcciones, actualizaciones y mejoras.

La arquitectura debe permitir la implementación de nuevas funcionalidades sin afectar la lógica del sistema existente.

## Otros requisitos

Esta sección incluye cualquier requisito adicional que no se clasifique en las categorías anteriores.

* **R-30: Cumplimiento Legal**

El desarrollo y la operación del sistema deben adherirse estrictamente a la **legislación chilena** en materia de protección de datos personales, asegurando la privacidad y el consentimiento de los usuarios.

* **R-31: Atributos de Usabilidad Adicionales**

El panel de gestión para el directorio debe tener un diseño tan intuitivo que no se requieran conocimientos informáticos previos para su uso efectivo. La interfaz debe ser **autoexplicativa** a través de iconos, etiquetas claras y un flujo de trabajo lógico.

## Requisitos de Base de Datos Lógica.

El sistema se fundamenta en un modelo de datos relacional diseñado en Django. La estructura lógica de la base de datos se compone de las siguientes entidades principales, que reflejan la arquitectura de tu solución.

**Entidad JuntaVecinos:**

Representa a cada organización comunitaria, siendo la entidad principal que agrupa al resto de la información.

**Atributos clave:** nombre, direccion, comuna, region.

**Relaciones:** Se relaciona con CustomUser, ProyectoVecinal, Noticia, Actividad y Espacio a través de claves foráneas.

**Entidad CustomUser:**

Modelo de usuario personalizado que almacena la información de todos los miembros de la comunidad.

**Atributos clave:** email (único), nombre, apellido, rut (único), telefono, direccion, rol.

**Relaciones:** Es la entidad central de la que dependen muchas otras. SolicitudCertificado, ProyectoVecinal, Noticia, Actividad, InscripcionActividad y SolicitudEspacio tienen claves foráneas que la referencian.

**Entidad SolicitudCertificado:**

Gestiona las peticiones de certificados.

**Atributos clave:** tipo, motivo, fecha\_solicitud, estado.

**Relaciones:** Mantiene una relación con CustomUser para identificar al vecino solicitante y al resuelto\_por (el directivo que gestionó la solicitud).

**Entidad ProyectoVecinal:**

Contiene la información de los proyectos propuestos por los vecinos.

**Atributos clave:** titulo, descripcion, fecha\_creacion, estado.

**Relaciones:** Se relaciona con la JuntaVecinos y con CustomUser para identificar al proponente y a quien lo revisado\_por.

**Entidades Noticia y NoticiaImagen:**

Noticia representa cada publicación o aviso.

**Atributos clave:** titulo, contenido, fecha\_creacion, autor.

**Relaciones:** Se vincula a JuntaVecinos y CustomUser (autor). NoticiaImagen tiene una clave foránea a Noticia para manejar las imágenes relacionadas.

**Entidades Actividad e InscripcionActividad:**

Actividad almacena los eventos comunitarios.

**Atributos clave:** titulo, descripcion, fecha, cupo\_maximo.

**Relaciones:** Se vincula a JuntaVecinos y CustomUser (creada\_por). InscripcionActividad es una tabla intermedia que registra qué vecino está inscrito en qué actividad.

**Entidades Espacio y SolicitudEspacio:**

Espacio representa un recurso físico (cancha, sala).

**Atributos clave:** nombre, tipo, descripcion.

**Relaciones:** Se vincula a JuntaVecinos. SolicitudEspacio registra las reservas de los espacios, vinculando a un solicitante (CustomUser) con un espacio.

# Propuesta de Planificación

## Descripción general acerca de la planificación

El proyecto de desarrollo del sistema "Junta360 Digital" se abordará mediante un enfoque estructurado y en cascada, garantizando un control riguroso de cada fase. El tiempo total estimado para la ejecución del proyecto es de **18 semanas**, seguido por un periodo de soporte inicial de 2 semanas para la corrección de errores. El equipo de trabajo multidisciplinario, con roles claramente definidos, será clave para el éxito.

Para asegurar un buen término, se consideran las siguientes buenas prácticas y condiciones:

* **Comunicación constante** entre el equipo y la junta de vecinos.
* **Validación y aprobación formal** de los entregables en cada fase.
* **Pruebas exhaustivas** en las etapas de integración y aceptación.

### Definición del Equipo de Trabajo

El equipo de trabajo está compuesto por los siguientes roles clave, con sus responsabilidades definidas para cada fase del proyecto.

|  |  |  |
| --- | --- | --- |
| Recurso | Departamento / División | Rol y Funciones |
| Analista de Sistemas | Análisis y Requisitos | Lidera el levantamiento y la validación de los requerimientos del cliente. |
| Diseñador de Sistemas | Diseño | Responsable del diseño funcional del sistema, incluyendo los wireframes y la experiencia de usuario (UX). |
| Arquitecto de Software | Arquitectura | Define la estructura técnica, selecciona las tecnologías y asegura que la solución sea escalable y mantenible. |
| Desarrollador Front-End | Desarrollo | Programa la interfaz de usuario, garantizando la usabilidad y la compatibilidad con dispositivos móviles. |
| Desarrollador Back-End | Desarrollo | Desarrolla la lógica del negocio, las APIs y la integración con la base de datos. |
| QA Tester | Calidad y Pruebas | Diseña y ejecuta los planes de pruebas para identificar y reportar errores. |
| DevOps | Operaciones | Gestiona el despliegue del sistema y configura el entorno de producción. |
| Líder de Proyecto | Gestión | Coordina al equipo, controla los plazos y el presupuesto, y actúa como el principal punto de contacto con el cliente. |
| Equipo de Soporte | Mantenimiento | Resuelve los problemas técnicos y los errores reportados después del lanzamiento. |

### Definición de Actividades principales del Proyecto

Las actividades principales se han agrupado en las siguientes fases:

1. **Fase de Planificación:** Incluye el levantamiento y validación de requerimientos.
2. **Fase de Diseño:** Abarca el diseño funcional y técnico de la solución.
3. **Fase de Implementación:** Se enfoca en la programación del Front-End, Back-End y la integración de módulos.
4. **Fase de Pruebas:** Comprende las pruebas unitarias, de integración y de aceptación.
5. **Fase de Despliegue y Soporte:** Incluye el despliegue del sistema, la capacitación y el periodo inicial de corrección de errores.

### Diagrama EDT

El **Diagrama de Estructura de Desglose del Trabajo (EDT)** descompone el proyecto en componentes jerárquicos. La estructura principal es la siguiente: Diagrama

El contenido generado por IA puede ser incorrecto.

### Carta Gantt

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nº** | **Actividad / Tarea** | **Duración** | **Inicio** | **Fin** | **Responsable** |
| **1** | Levantamiento de Requerimientos | 1 semana | 06-ago-25 | 12-ago-25 | Analista de sistemas |
| **2** | Validación de Requerimientos | 2 días | 13-ago-25 | 14-ago-25 | Analista de sistemas |
| **3** | Diseño Funcional | 1 semana | 15-ago-25 | 21-ago-25 | Diseñador de sistemas |
| **4** | Diseño Técnico | 1 semana | 22-ago-25 | 28-ago-25 | Arquitecto de software |
| **5** | Programación Front-End | 2 semanas | 29-ago-25 | 11-sept-25 | Desarrollador Front-End |
| **6** | Programación Back-End | 3 semanas | 12-sept-25 | 02-oct-25 | Desarrollador Back-End |
| **7** | Integración de Módulos | 1 semana | 03-oct-25 | 09-oct-25 | Equipo de desarrollo |
| **8** | Pruebas Unitarias | 1 semana | 10-oct-25 | 16-oct-25 | QA Tester |
| **9** | Pruebas de Integración y Aceptación | 1 semana | 17-oct-25 | 23-oct-25 | QA |
| **10** | Despliegue del Sistema | 3 días | 24-oct-25 | 28-oct-25 | DevOps |
| **11** | Capacitación a Usuarios | 3 días | 29-oct-25 | 31-oct-25 | Líder de proyecto |
| **12** | Corrección de Errores Iniciales (post-entrega) | 2 semanas | 01-nov-25 | 14-nov-25 | Equipo de soporte |

### Resumen de Costos del Desarrollo del Proyecto

El costo estimado del proyecto se ha calculado considerando tanto los recursos tecnológicos necesarios para la operación anual del sistema como el esfuerzo en horas-hombre del equipo de desarrollo. La siguiente tabla presenta el desglose detallado de los costos proyectados.

|  |  |  |
| --- | --- | --- |
| Costos Generales | Desglose de Gastos | Costo Total |
| Recursos Tecnológicos | Hosting web | $100.000 CLP (anual) |
|  | Dominio | $15.000 CLP (anual) |
|  | Licencias/Servicios (WhatsApp API) | $360.000 CLP (anual, aprox.) |
| Recursos Humanos | Desarrollo (300 horas x $10.000 CLP/hr) | $3.000.000 CLP |
|  | Testing y QA (50 horas x $10.000 CLP/hr) | $500.000 CLP |
|  |  |  |
| TOTAL | Costo Estimado del Proyecto | $3.975.000 CLP |

Nota: Se considera un costo anual para la Hosting y Dominio, y un costo mensual para la API de WhatsApp que asciende a 360.000 CLP al año. El costo total estimado del proyecto es de **$3.975.000 CLP**, lo que incluye la inversión inicial en desarrollo y un año de operación de los servicios tecnológicos.

## Plan de control de Cambio

El plan de control de cambios es una actividad transversal que se aplica durante todo el proyecto. Su propósito es gestionar de manera formal cualquier solicitud de modificación que surja después de la aprobación de los requerimientos iniciales.

**Tipos de Cambio y Alcances**: Se podrán aceptar cambios menores en la funcionalidad (por ejemplo, mejoras de usabilidad) o en aspectos no funcionales (rendimiento) siempre que no afecten la arquitectura central de la solución. Los cambios que impliquen una reestructuración significativa de la base de datos o de los flujos principales del sistema serán rechazados o requerirán una nueva evaluación de alcance.

**Procedimiento**: Todo cambio debe ser propuesto por escrito, validado por el analista y el líder del proyecto, y aprobado por el cliente. Se registrará en una Plantilla de Control de Cambio para su seguimiento.

**Importancia**: El control de cambios es vital para evitar la "deriva del y garantizar que el proyecto se complete a tiempo y dentro del presupuesto.

# Anexos

**Plantilla de Control de Cambio ¿????????**

**Calculo hora hombre ¿????????**