**Documento De Requerimientos De Software**

Sistema Territorial Comunitario (SITECOM)

Fecha: 06-09-2025

Integrantes: Francisco Bravo

Rodrigo Bahamonde

Martin Hernandez

Sección: 707 V

**Tabla de contenido**

Historial de Versiones 3

Información del Proyecto 3

Aprobaciones 3

1. Propósito 4

2. Alcance del producto / Software 4

3. Referencias 4

4. Funcionalidades del producto 5

5. Clases y características de usuarios 5

6. Entorno operativo 5

7. Requerimientos funcionales 6

9.1. (Nombre de la funcionalidad 1) 6

9.2. (Nombre de la funcionalidad 2) 7

9.3. (Nombre de la funcionalidad N) 7

8. Reglas de negocio 8

9. Requerimientos de interfaces externas 9

9.1. Interfaces de usuario 9

9.2. Interfaces de hardware 9

9.3. Interfaces de software 9

9.4. Interfaces de comunicación 9

10. Requerimientos no funcionales 10

11. Otros requerimientos 11

12. Glosario 12

**Historial de Versiones**

| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| --- | --- | --- | --- | --- |
| 06-09-2025 | 1 | Francisco Bravo | DevWeb | Primera versión donde definimos los principales puntos de nuestro proyecto. |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Información del Proyecto**

| Empresa / Organización |  |
| --- | --- |
| Proyecto | Sistema Territorial Comunitario (SITECOM) |
| Fecha de preparación | 06/08/2025 |
| Cliente | Junta de Vecinos |
| Patrocinador principal | Vecinos del barrio |
| Gerente / Líder de Proyecto | Francisco Bravo |
| Gerente / Líder de Análisis de negocio y requerimientos | Rodrigo Bahamonde |

**Aprobaciones**

| **Nombre y Apellido** | **Cargo** | **Departamento u Organización** | **Fecha** | **Firma** |
| --- | --- | --- | --- | --- |
| Fernando Herrera | Profesor | Duoc UC | 06-09-2025 |  |
| Francisco Bravo | Gerente / Líder de Proyecto | Empresa De Desarrollo | 06-09-2025 |  |
| Rodrigo Bahamonde | Gerente / Líder de Análisis de negocio y requerimientos | Empresa De Desarrollo | 06-09-2025 |  |
| Martin Hernandez | QA Manager | Empresa De Desarrollo | 06-09-2025 |  |

1. **Propósito**

El presente documento especifica los requerimientos del sistema denominado **“Sistema Territorial Comunitario (SITECOM)”**, versión **1.0 (Release Inicial)**.

El propósito del software es proveer a las juntas de vecinos de Chile una herramienta tecnológica que permita centralizar y digitalizar sus procesos administrativos, comunicacionales y de gestión comunitaria, mejorando la eficiencia, transparencia y participación ciudadana dentro de la unidad territorial.

Este documento cubre la **totalidad del sistema** en su versión inicial (MVP), incluyendo los siguientes componentes principales:

* **Gestión de vecinos:** inscripción, registro y administración de datos de los socios de la junta.
* **Gestión de certificados de residencia:** solicitud, revisión, aprobación/rechazo y emisión digital.
* **Gestión de proyectos vecinales:** postulación, seguimiento y resolución de proyectos comunitarios.
* **Gestión de notificaciones:** envío multicanal (correo electrónico, WhatsApp y afiches digitales) y recepción por los vecinos.
* **Gestión de noticias:** publicación y visualización de noticias comunitarias.
* **Gestión de espacios comunitarios:** solicitud, aprobación y control de reservas de salas, canchas y plazas mediante calendario.
* **Gestión de actividades vecinales:** creación de actividades, control de cupos e inscripción de vecinos.
* **Gestión de usuarios y roles:** control de accesos según perfiles (administrador, directorio, vecino).

Quedan **fuera del alcance de este documento** los siguientes elementos:

* Desarrollo de aplicaciones móviles nativas (Android/iOS).
* Integraciones externas con sistemas gubernamentales.
* Procesos de pago en línea o recaudación digital.

Este documento abarca los **requerimientos funcionales y no funcionales** necesarios para la construcción del sistema en su primera entrega, sirviendo de base para el diseño, desarrollo, implementación y validación del producto.

1. **Alcance del producto / Software**

El **Sistema Territorial Comunitario (SITECOM)** tiene como objetivo principal proporcionar a las juntas de vecinos una plataforma digital integral que facilite la gestión administrativa, la comunicación interna y externa, y la participación activa de la comunidad.

#### Propósito u objetivo general

Desarrollar un sistema web que centralice y digitalice los procesos más relevantes de la junta de vecinos, permitiendo optimizar tiempos, recursos humanos y materiales, además de fortalecer la comunicación y la transparencia en la gestión comunitaria.

#### Beneficios para la organización

* **Eficiencia administrativa:** reducción significativa de tiempos y costos en la gestión de trámites como inscripciones y certificados.
* **Mejora en la comunicación:** envío y recepción de notificaciones multicanal (email, WhatsApp, afiches digitales) que aseguran mayor alcance entre los vecinos.
* **Participación ciudadana:** incremento en la inscripción de vecinos, postulaciones de proyectos y participación en actividades comunitarias.
* **Transparencia:** trazabilidad en la revisión y aprobación de trámites y proyectos.
* **Accesibilidad:** disponibilidad del sistema desde cualquier dispositivo con conexión a internet.

#### Objetivos y metas del software

* Implementar módulos de gestión para: vecinos, certificados, proyectos, actividades, reservas de espacios, noticias y notificaciones.
* Alcanzar una adopción inicial de al menos el 70% de los vecinos de la unidad territorial durante el primer año.
* Reducir en un 50% el tiempo de tramitación de certificados y solicitudes comunitarias.
* Generar un aumento del 30% en la participación de vecinos en actividades comunitarias.
* Establecer un sistema escalable y adaptable que pueda ser replicado en diferentes juntas de vecinos a nivel nacional.

#### Relación con objetivos organizacionales

El SITECOM contribuye directamente a los objetivos estratégicos de las juntas de vecinos:

* **Promover la participación comunitaria** a través de herramientas digitales accesibles.
* **Optimizar la gestión organizacional** mediante la reducción de tiempos y costos administrativos.
* **Mejorar la comunicación con los vecinos**, fortaleciendo la transparencia y confianza en la organización.
* **Facilitar la planificación y control de recursos comunitarios**, como espacios y actividades.

#### Referencia documental

Este alcance se encuentra alineado con lo definido en el **Acta de Constitución del Proyecto** y se detalla de manera más específica en este Documento de Requerimientos de Software, que servirá de base para el diseño, desarrollo e implementación del SITECOM.

1. **Referencias**

**Acta de Constitución del Proyecto SITECOM**

* Autor: Equipo de Proyecto SITECOM
* Versión: 1.0
* Fecha: Agosto 2025
* Ubicación: Carpeta de documentación del proyecto / Confluence / Repositorio compartido

**Definición de Alcance – SITECOM**

* Autor: Equipo de Análisis de Requerimientos
* Versión: 1.0
* Fecha: Agosto 2025
* Ubicación: Documento de gestión de alcance en Jira / Carpeta de proyecto

**Documento de Visión del Proyecto SITECOM**

* Autor: Dirección del Proyecto – Junta de Vecinos
* Versión: 1.0
* Fecha: Agosto 2025
* Ubicación: Carpeta de documentación del proyecto

**Políticas y Procedimientos de Junta de Vecinos (Chile)**

* Autor: Junta de Vecinos – Reglamentos internos
* Versión: Vigente
* Fecha: Última actualización municipal/local
* Ubicación: Biblioteca municipal / Normativas locales en sitio web municipal

**Flujogramas de Procesos del SITECOM**

* Autor: Equipo de Análisis de Procesos
* Versión: 0.9 (borrador)
* Fecha: Agosto 2025
* Ubicación: Carpeta de diagramas BPMN en repositorio compartido

**Metodología de trabajo – Guía Scrum aplicada en Jira**

* Autor: Project Manager SITECOM
* Versión: 1.0
* Fecha: Agosto 2025
* Ubicación: Espacio de Jira / Manual interno del equipo

1. **Funcionalidades del producto**

**Gestión de vecinos**

* Registro e inscripción de nuevos vecinos en la junta.
* Administración de información personal y actualización de datos.

**Gestión de certificados de residencia**

* Solicitud en línea de certificados.
* Revisión, aprobación o rechazo por parte del directorio.
* Emisión digital del certificado con notificación al solicitante.

**Gestión de proyectos vecinales**

* Postulación de proyectos por parte de los vecinos inscritos.
* Revisión, aprobación o rechazo por parte de la junta/directorio.
* Comunicación de resultados vía email o notificación digital.

**Gestión de notificaciones y avisos**

* Creación y envío de avisos comunitarios.
* Distribución multicanal (email, WhatsApp, afiches digitales).
* Recepción de notificaciones por parte de los vecinos.

**Gestión de noticias comunitarias**

* Publicación de noticias relevantes por la junta.
* Visualización pública de noticias por los habitantes de la unidad territorial.

**Gestión de espacios comunitarios**

* Solicitud de uso de canchas, salas, plazas u otros espacios.
* Administración de calendario de reservas.
* Aprobación y control de disponibilidad de los espacios.

**Gestión de actividades comunitarias**

* Creación de actividades y eventos vecinales.
* Inscripción de vecinos en actividades (con control de cupos).
* Listado de participantes inscritos.

**Gestión de usuarios y roles**

* Creación de cuentas con diferentes perfiles (administrador, directorio, vecino).
* Control de accesos y permisos según rol.

**Sistema de comunicación y transparencia**

1. Registro de resoluciones de solicitudes y proyectos.
2. Envío automático de respuestas a solicitantes.
3. Historial de comunicaciones y documentos disponibles para consulta.
4. **Clases y características de usuarios**

El **Sistema Territorial Comunitario (SITECOM)** está diseñado para ser utilizado por distintos tipos de usuarios, cada uno con características específicas en cuanto a frecuencia de uso, nivel de acceso, y funcionalidades disponibles.

#### 1. Administrador del sistema

* **Descripción:** Usuario con privilegios totales, encargado de la configuración inicial y mantenimiento general del sistema.
* **Características:**
  + Uso ocasional, principalmente en etapas de configuración o soporte.
  + Requiere conocimientos técnicos intermedios-altos.
* **Funcionalidades relevantes:**
  + Gestión de usuarios y roles.
  + Configuración de notificaciones multicanal.
  + Administración general del sistema.

#### 2. Directorio de la junta de vecinos

* **Descripción:** Autoridad de la junta encargada de aprobar/rechazar solicitudes, gestionar proyectos, publicar noticias y garantizar la comunicación con los vecinos.
* **Características:**
  + Uso frecuente (varias veces por semana).
  + Privilegios medios-altos.
  + Experiencia básica en sistemas web.
* **Funcionalidades relevantes:**
  + Revisión y aprobación/rechazo de solicitudes y proyectos.
  + Emisión de certificados de residencia.
  + Creación de actividades y eventos comunitarios.
  + Publicación de noticias y avisos.
  + Administración del calendario de reservas de espacios.

#### 3. Personal autorizado (secretaría, apoyo administrativo)

* **Descripción:** Colaboradores designados por la junta para apoyar en la gestión administrativa.
* **Características:**
  + Uso frecuente.
  + Privilegios medios, limitados a gestión operativa.
* **Funcionalidades relevantes:**
  + Registro de vecinos.
  + Emisión de certificados de residencia.
  + Carga de documentos y noticias.
  + Apoyo en la gestión de inscripciones a actividades.

#### 4. Vecino miembro de la junta (registrado)

* **Descripción:** Habitante de la unidad territorial inscrito formalmente como miembro de la junta de vecinos.
* **Características:**
  + Uso ocasional o frecuente según necesidades.
  + Privilegios medios-bajos.
  + Requiere solo conocimientos básicos de navegación web.
* **Funcionalidades relevantes:**
  + Inscripción en la junta.
  + Solicitud de certificados de residencia.
  + Postulación de proyectos vecinales.
  + Inscripción en actividades comunitarias.
  + Recepción de notificaciones y avisos.
  + Visualización de noticias.

#### 5. Habitante de la unidad territorial (no registrado / público)

* **Descripción:** Personas mayores de 14 años que residen en la unidad territorial, pero que aún no forman parte de la junta.
* **Características:**
  + Uso ocasional.
  + Privilegios mínimos.
* **Funcionalidades relevantes:**
  + Solicitud de inscripción en la junta de vecinos.
  + Consulta de noticias comunitarias.
  + Recepción de avisos públicos.
  + Inscripción en actividades comunitarias abiertas.

1. **Entorno operativo**

En esta sección se describe el entorno operativo en el que se desenvolverá el sistema, software, módulo o grupo de funcionalidades, mencionando aspectos como la plataforma de hardware, versiones de sistema operativo y otros sistemas o componentes con los que debe coexistir.

DEFINIR BIEN LO QUE VAMOS A UTILIZAR PARA CUMPLIR ESTE PUNTO

1. **Requerimientos funcionales**
2. **Gestión de Inscripción de Vecinos:**

**Descripción**: Permite a los habitantes registrarse en la junta de vecinos mediante un formulario en línea.

**Prioridad**: Alta

**Acciones iniciadas y comportamiento esperado:**

* El usuario accede al formulario de inscripción.
* Ingresa sus datos personales (nombre, dirección, RUT, correo, teléfono).
* El sistema valida los datos ingresados (formato de correo, RUT válido, campos obligatorios).
* El sistema registra al usuario en la base de datos.
* El sistema muestra un mensaje de confirmación y envía un correo electrónico de bienvenida.

**Requerimientos funcionales:**

* **REQ-INS-1:** El sistema debe permitir el registro de vecinos mediante un formulario en línea.
* **REQ-INS-2:** El sistema debe validar campos obligatorios antes de guardar la información.
* **REQ-INS-3:** El sistema debe verificar que el RUT tenga un formato válido.
* **REQ-INS-4:** El sistema debe enviar un correo de confirmación al vecino inscrito.
* **REQ-INS-5:** El sistema debe mostrar un mensaje de error en caso de datos inválidos.

1. **Emisión de Certificados de Residencia**

**Descripción:** Permite a los vecinos solicitar y obtener un certificado de residencia firmado digitalmente por la directiva.

**Prioridad:** Alta

**Acciones iniciadas y comportamiento esperado:**

* El vecino solicita el certificado desde su cuenta en el sistema.
* El sistema verifica que el vecino esté inscrito y activo.
* El sistema genera un certificado en formato PDF con los datos del solicitante.
* El sistema notifica a la directiva para validación/aprobación.
* Una vez aprobado, el certificado queda disponible para descarga y se envía por correo electrónico.

**Requerimientos funcionales:**

1. **REQ-CER-1:** El sistema debe permitir a los vecinos solicitar un certificado de residencia.
2. **REQ-CER-2:** El sistema debe generar certificados en formato PDF.
3. **REQ-CER-3:** El sistema debe permitir a la directiva aprobar o rechazar solicitudes de certificados.
4. **REQ-CER-4:** El sistema debe enviar un correo al vecino con el certificado aprobado.
5. **REQ-CER-5:** El sistema debe registrar el historial de certificados emitidos.
6. **Gestión de Proyectos Vecinales**

**Descripción:** Permite a los miembros de la junta postular proyectos vecinales y que la directiva los apruebe o rechace.

**Prioridad:** Alta

**Acciones iniciadas y comportamiento esperado:**

* El vecino registrado presenta un proyecto a través del sistema (formulario con título, descripción, presupuesto estimado).
* El sistema almacena la propuesta en la base de datos.
* La directiva recibe la notificación de un nuevo proyecto.
* La directiva aprueba o rechaza el proyecto.
* El sistema notifica al vecino del resultado de la evaluación.

**Requerimientos funcionales:**

1. **REQ-PRO-1:** El sistema debe permitir a los vecinos postular proyectos mediante un formulario en línea.
2. **REQ-PRO-2:** El sistema debe notificar a la directiva sobre nuevos proyectos postulados.
3. **REQ-PRO-3:** El sistema debe permitir a la directiva aprobar o rechazar proyectos.
4. **REQ-PRO-4:** El sistema debe notificar al vecino sobre la resolución del proyecto.
5. **REQ-PRO-5:** El sistema debe almacenar el historial de proyectos postulados.
6. **Gestión de Noticias y Avisos**

**Descripción:** Permite a la directiva o personal autorizado crear, editar y publicar noticias o avisos importantes para los habitantes de la unidad territorial.

**Prioridad:** Media

**Acciones iniciadas y comportamiento esperado:**

* El usuario autorizado accede al módulo de noticias.
* Crea una nueva noticia ingresando título, descripción, fecha de publicación y (opcional) imagen o documento adjunto.
* El sistema guarda la noticia en la base de datos.
* La noticia se publica en el portal público de la junta de vecinos y queda visible para todos los usuarios.
* El sistema puede enviar una notificación (email o WhatsApp) a los vecinos registrados informando sobre la nueva publicación.
* El usuario autorizado puede editar o eliminar noticias existentes.

**Requerimientos funcionales:**

* **REQ-NOT-1:** El sistema debe permitir a los usuarios autorizados crear noticias y avisos.
* **REQ-NOT-2:** El sistema debe mostrar las noticias en un portal público accesible a todos los vecinos.
* **REQ-NOT-3:** El sistema debe permitir adjuntar imágenes o archivos a las noticias.
* **REQ-NOT-4:** El sistema debe permitir editar y eliminar noticias publicadas.
* **REQ-NOT-5:** El sistema debe enviar notificaciones a los vecinos cuando se publique una nueva noticia (opcional, según configuración).
* **REQ-NOT-6:** El sistema debe mostrar las noticias ordenadas por fecha de publicación, de más reciente a más antigua.

1. **Reglas de negocio**

**Registro de usuarios**

* Solo los habitantes de la unidad territorial pueden inscribirse en el sistema.
* Cada usuario debe estar identificado por un RUT válido y único en la base de datos.

**Roles y permisos**

* Solo los miembros de la directivao personal autorizado pueden aprobar/rechazar solicitudes, emitir certificados y publicar noticias.
* Los vecinos registrados pueden realizar solicitudes, postular proyectos y visualizar noticias.
* El sistema debe diferenciar claramente entre roles de usuario (vecino, directiva, administrador).

**Aprobaciones**

* Toda solicitud de certificado, proyecto vecinal o reserva de espacio debe ser aprobada o rechazada por la directiva antes de estar disponible para el solicitante.
* El sistema debe registrar el nombre del aprobador, la fecha y el resultado de la resolución.

**Notificaciones**

* Toda acción importante (aprobación, rechazo, publicación de noticia, confirmación de inscripción) debe generar una notificación automática al usuario involucrado.
* El sistema debe permitir que las notificaciones se envíen vía correo electrónico y, cuando corresponda, vía WhatsApp.

**Historial y auditoría**

* Todas las acciones relevantes (inscripción, solicitudes, proyectos, certificados, publicaciones) deben quedar registradas en un historial con fecha, hora y usuario responsable.
* Los registros no deben eliminarse, solo inhabilitarse (mantener trazabilidad).

**Disponibilidad del sistema**

* El sistema debe estar disponible para los usuarios 24/7.
* Las tareas administrativas de mantenimiento (copias de seguridad, actualizaciones) deben realizarse sin afectar la disponibilidad principal.

**Integridad de datos**

* Ningún registro puede quedar con campos obligatorios vacíos (ejemplo: RUT, nombre, dirección en inscripción).
* El sistema debe rechazar información inválida o incompleta, mostrando mensajes de error claros al usuario.

**Comunicación oficial**

* Todas las noticias y avisos publicados en el sistema se consideran comunicación oficial de la junta de vecinos.
* Solo los usuarios con rol de directiva pueden crear, modificar o eliminar noticias.

1. **Requerimientos de interfaces externas**
2. **Interfaces de usuario**

La interfaz del usuario será muy intuitiva, se mostrará en pantalla las opciones justas y necesarias debido a que la página la utilizara tanto una persona joven como un adulto mayor, es por esto que será una interfaz muy amigable con el usuario.

En la parte del menú, solo tendrá las opciones dada por los roles definidos, el administrador podrá ver absolutamente todo, en cambio los usuarios tendrán acceso limitado.

En las pantallas de creación de usuarios,tendremos una api para la validación de rut, con la finalidad de asegurarnos que las personas que se quieran unir sea gente que existe, además la disposición de los botones será de un tamaño grande para que sea posible visualizar todas las opciones y que quede claro con tan solo mirar la pantalla.

1. **Interfaces de hardware**

### Dispositivos soportados

* Computadores de escritorio y notebooks**.**
* Tablets y smartphones (Android 10+ / iOS 14+), orientaciones vertical/horizontal.

### Interacciones y restricciones

* **Impresión:** desde el navegador de certificados y reportes en PDF (A4).
* **Cámara/archivos:** carga opcional de imagen para noticias o comprobantes (límite configurable, ej. 5 MB; tipos: JPG/PNG/PDF).

### 

### Protocolos y compatibilidad

* Conectividad HTTPS (TLS 1.2+).
* Navegadores soportados: Chrome, Edge, Firefox, Safari versiones actuales y dos anteriores.

1. **Interfaces de software**

## Sistema operativo, runtime y frameworks

* **Backend:** ASP.NET Core 8+ (o .NET soportado por la institución).
* **Frontend:** HTML5/CSS3/JavaScript (ES6+), Bootstrap 5.x, FullCalendar.

## Base de datos e interacción

* **Motor de BD:** **SQL Server** (alternativa: PostgreSQL/MySQL si lo requiere la infraestructura).
* **Acceso:** Procedimientos almacenados solo para casos críticos.
* **Integridad:** Claves foráneas, índices, constraints y transacciones (ACID).
* **Conectividad:** Cadena de conexión vía secrets/variables de entorno; **pooling** habilitado.

## Librerías y componentes de terceros

* **UI:** Bootstrap 5, Bootstrap Icons.
* **Calendario:** FullCalendar (módulo de reservas con eventos).
* **PDFs:** RDLC
* **Validación:** Librerías para validador de RUT (front y back).
* **Uploads:** Manejo de archivos con validación de tipo/tamaño y antivirus del servidor si aplica.

## Integraciones externas (APIs)

**- Email transaccional:** SMTP institucional o proveedor (SendGrid/SES) vía SMTP o REST.

**- WhatsApp Business Platform:** API REST con plantillas aprobadas; almacenamiento de estado de envío.

**- Captcha/anti bot:** reCAPTCHA (v2/v3) en formularios públicos.

1. **Interfaces de comunicación**

## Protocolos y navegación web

* **HTTP/1.1 y HTTP/2** obligatoriamente sobre **HTTPS (TLS 1.2+)**.
* **Compresión** (Gzip/Brotli) habilitada para respuestas > 1 KB.
* **Headers estándar**:

-Content-Type: application/json; charset=utf-8

-Cache-Control según criticidad (no-cache para datos sensibles).

-Authorization: Bearer <JWT> para APIs seguras.

* **Límites**: tamaño máximo de payload (p. ej., 5–10 MB); timeout por request (30–60 s).

1. **Requerimientos no funcionales**

### Usabilidad

* **RNF-01:** El sistema debe ser intuitivo y fácil de usar, permitiendo que cualquier usuario sin conocimientos técnicos pueda operar las funcionalidades principales.
* **RNF-02:** La interfaz debe seguir principios de accesibilidad (WCAG 2.1 AA), garantizando legibilidad, contraste adecuado y navegación por teclado.

### Rendimiento

* **RNF-03:** El sistema debe ser capaz de atender al menos 100 usuariosconcurrentes sin degradación significativa del rendimiento.
* **RNF-04:** Los tiempos de respuesta a las solicitudes de usuario no deben exceder los 3 segundos en condiciones normales de carga.
* **RNF-05:** La generación de certificados y reportes en PDF debe completarse en menos de 10 segundos.

**Seguridad**

* **RNF-06:** Todas las comunicaciones entre cliente y servidor deben realizarse bajo protocolo HTTPS.
* **RNF-07:** Las contraseñas deben almacenarse de manera cifrada usando algoritmos seguros.
* **RNF-08:** El sistema debe implementar controles de acceso por roles y permisos para diferenciar usuarios (vecinos, directiva, administrador).
* **RNF-09:** El sistema debe contar con validaciones para prevenir ataques de SQL Injection.
* **RNF-10:** El sistema debe registrar en bitácoras (logs) las acciones críticas de los usuarios (altas, bajas, modificaciones, accesos).

### Confiabilidad y disponibilidad

* **RNF-11:** El sistema debe tener una disponibilidad las 24 horas del día.
* **RNF-12:** En caso de caída, el sistema debe poder recuperarse en un tiempo máximo de 2 horas.

### Mantenibilidad y escalabilidad

* **RNF-13:** La base de datos debe poder manejar un crecimiento de **al menos 10.000 registros de vecinos** sin pérdida de rendimiento significativo.
* **RNF-14:** El sistema debe ser fácilmente escalable para soportar nuevas funcionalidades (ejemplo: integración con otros sistemas municipales).

### Portabilidad

* **RNF-15:** El sistema debe ser responsivo, accesible desde navegadores modernos (Chrome, Firefox, Edge, Safari).
* **RNF-16:** Debe funcionar correctamente en computadores, tablets y smartphones con resoluciones mínimas de 360x640 px.
* **RNF-17:** La base de datos debe poder migrarse a otro servidor sin pérdida de datos ni corrupción de información.

1. **Otros requerimientos**

1. **Glosario**

SITECOM: Sistema Territorial Comunitario