
Especificación de requisitos de software

Para

Non-Profit Organization

Versión 3.1 sin aprobar

Preparado por Juan Silgado, Miguel Villa, Dylan Ecker

Universidad Tecnológica de Bolívar

23 de abril de 2025

Tabla de contenido

Historial de revisiones	3
1. Introducción	1
2. Descripción general	2
3. Características del sistema	3
3.1 Gestión de Donantes y Voluntarios	3
3.1.1 Descripción y Prioridad	3
3.1.2 Secuencias de Estímulo/Respuesta	3
3.1.3 Requisitos Funcionales	3
3.2 Gestionar Usuario	3
3.2.2 Secuencias de estímulo/respuesta	3
3.2.3 Requisitos funcionales	4
3.3 Gestionar Contenido	4
3.3.2 Secuencias de estímulo/respuesta	4
3.3.3 Requisitos funcionales	4
3.4 Gestión de Campañas	4
3.4.2 Secuencias de estímulo/respuesta	5
3.4.3 Requisitos funcionales	5
3.5 Gestión de Eventos	5
3.5.2 Secuencias de estímulo/respuesta	5
3.5.3 Requisitos funcionales	5
3.6 Reportes y Analítica	5
3.6.2 Secuencias de estímulo/respuesta	6
3.6.3 Requisitos funcionales	6
3.7 Gestión de Roles y Permisos	6
3.7.2 Requisitos funcionales	6
3.8 Gestión de Programas	6
3.8.2 Requisitos funcionales	6
4. Requisitos de la interfaz	7
4.1 Interfaces de Usuario	7
4.2 Interfaces de hardware	7
4.3 Interfaces de software	7
4.4 Requerimientos de desempeño	8
4.5 Requisitos de seguridad	8
4.6 Atributos de calidad del software	8
5. Otros Requerimientos	9
6. Características del MVP	9
7. Pruebas	9
7.1 Pruebas funcionales de servicios	9
7.2 Pruebas de integración	9

Historial de revisiones

Nombre	Fecha	Motivo de los cambios	Versión
Inicio del proyecto	16/03/25	documentación inicial del proyecto	0.1
MVP	16/03/25	documentación inicial del MVP del proyecto	1.0
Actualización 2.0	22/03/25	Añadir información parcial de arquitectura y demás requerimientos	2.0
Actualización 3.0	26/03/25	Integración completa de la arquitectura, interfaz, MVP y actualización de requerimientos	3.0
Actualización 3.1	04/06/25	SRS ampliado con alineación al MVP	3.1

1. Introducción

Este documento especifica los requisitos para el desarrollo de una plataforma digital integral orientada a la gestión de datos en una organización sin fines de lucro. El objetivo principal es consolidar la información relacionada con donantes, voluntarios, campañas, eventos y programas comunitarios, facilitando su administración desde una única plataforma centralizada.

La solución propuesta estará basada en una arquitectura de microservicios escalable, lo que permitirá flexibilidad en la implementación y mantenimiento. Esta arquitectura incluirá servicios independientes para usuarios, contenido, métricas y programas, los cuales estarán expuestos a través de APIs seguras. Además, se integrará un data warehouse en la nube que permitirá centralizar, consultar y analizar la información de manera eficiente, con soporte para visualización de métricas en tiempo real mediante paneles interactivos.

Este documento detalla tanto los requisitos funcionales como no funcionales del sistema, así como sus interfaces, restricciones de diseño, y consideraciones de calidad, rendimiento y seguridad.

2. Descripción general

El sistema reemplazará la infraestructura fragmentada actual, unificando herramientas y flujos operativos de recaudación de fondos, gestión de voluntarios, campañas y programas. La nueva plataforma se construirá sobre una arquitectura de microservicios basada en APIs RESTful, facilitando el mantenimiento modular, la escalabilidad y la interoperabilidad con soluciones de terceros. La solución estará alojada en la nube para garantizar alta disponibilidad, redundancia, y capacidad de análisis avanzado de datos en tiempo real.

2.1 Características del producto

- Arquitectura de microservicios para gestión modular de donantes, voluntarios, eventos y campañas.
- API Gateway para centralizar las llamadas entre servicios y aplicar reglas de autenticación y control de acceso.
- Integración con un data warehouse en la nube, que centraliza información para fines analíticos y de toma de decisiones.
- Sistema de analítica y reportes interactivos para visualizar indicadores clave (KPI) de forma dinámica.
- Interfaz web responsiva para permitir el acceso desde múltiples dispositivos.

2.2 Clases de usuarios y características

- **Administradores:** Acceso completo al sistema, con permisos para gestionar usuarios, roles, campañas y contenido.
- **Coordinadores de voluntarios:** Permisos para planificar eventos, asignar voluntarios y realizar seguimiento de actividades.
- **Donantes:** Acceso a su historial de donaciones, métricas de impacto social y opciones para participar en campañas.
- **Voluntarios:** Visualización de eventos activos, registro a campañas, y consulta de reconocimientos obtenidos.

2.3 Limitaciones de diseño y aplicación

- El sistema dependerá de conexión estable a Internet para su operación completa.
- Se integrará exclusivamente con servicios externos que cumplan estándares de seguridad y documentación (APIs REST con OAuth2).
- Cumplimiento obligatorio con normas de privacidad (GDPR, Ley Habeas Data, etc.) y principios de seguridad de la información (ISO 27001).

2.4 Documentación del Usuario

- Manual detallado para administradores y coordinadores, disponible en línea y en formato PDF.
- Guía técnica de integración de APIs y servicios para desarrolladores externos.
- Videos tutoriales accesibles desde el panel de ayuda de la plataforma.

3. Características del sistema

3.1 Gestión de Donantes y Voluntarios

3.1.1 Descripción y Prioridad

- Este módulo permite la administración integral de la información relacionada con donantes y voluntarios dentro de una plataforma centralizada. Incluye funcionalidades para el **registro, consulta, edición, eliminación lógica y seguimiento histórico** de actividades asociadas. Permite asociar donaciones a campañas específicas, categorizar voluntarios según habilidades o disponibilidad, y mantener contacto a través de notificaciones automáticas. Prioridad: Alta.

3.1.2 Secuencias de Estímulo/Respuesta

1. Un usuario ingresa y consulta información de donantes o voluntarios.
2. El sistema recupera y muestra los datos desde el data warehouse.

3.1.3 Requisitos Funcionales

- REQ-1: El sistema debe permitir registrar nuevos donantes y voluntarios mediante formularios validados en frontend y backend.
- REQ-2: Debe permitir consultar, filtrar, editar y eliminar registros existentes, manteniendo integridad referencial con campañas, eventos o actividades asociadas.
- REQ-4: Los cambios realizados sobre los registros deben quedar trazados en un historial de auditoría (quién, cuándo y qué fue modificado).

3.2 Gestionar Usuario

3.2.1 Descripción y Prioridad

Esta funcionalidad permite a los administradores gestionar la información de los usuarios registrados en la plataforma. Incluye creación, modificación, eliminación lógica y asignación de roles (como administrador, coordinador, donante, voluntario, etc.). También se implementará el control de acceso basado en roles (RBAC) para definir permisos específicos.

Prioridad: Alta

3.2.2 Secuencias de estímulo/respuesta

1. Un administrador accede al panel de gestión de usuarios.
2. El sistema solicita credenciales y verifica autenticación.
3. Se muestra la lista de usuarios existentes con filtros y opciones de edición.
4. El administrador realiza acciones como agregar, editar o eliminar un usuario.
5. El sistema valida los datos, realiza la operación y actualiza el historial de auditoría.

3.2.3 Requisitos funcionales

- **REQ-5:** El sistema debe permitir registrar nuevos usuarios mediante formularios con validación.
- **REQ-6:** Debe permitir asignar y cambiar roles de los usuarios existentes.
- **REQ-7:** El sistema debe mantener un historial de cambios asociados a cada usuario (quién, cuándo y qué).
- **REQ-8:** Debe integrarse con un sistema de autenticación centralizado (OAuth2, JWT).
- **REQ-9:** El sistema debe permitir la búsqueda y filtrado avanzado de usuarios por rol, estado o nombre.

3.3 Gestionar Contenido

3.3.1 Descripción y Prioridad

Este módulo permite a los administradores publicar, editar y eliminar contenido relacionado con campañas, eventos, noticias y materiales educativos. La publicación de contenido dinámico en la plataforma es clave para mantener informados y motivados a los donantes y voluntarios.

Prioridad: Media

3.3.2 Secuencias de estímulo/respuesta

1. Un administrador ingresa a la sección de contenidos.
2. Elige una categoría de contenido (evento, campaña, noticia, etc.).
3. Crea o edita una entrada con texto, imágenes y fechas.
4. El contenido se publica y es visible para los usuarios según su rol.

3.3.3 Requisitos funcionales

- **REQ-10:** El sistema debe permitir a los administradores gestionar contenido multimedia y textual.
- **REQ-11:** El contenido debe poder clasificarse por tipo (evento, campaña, etc.) y tener una fecha de expiración.
- **REQ-12:** Los usuarios solo deben ver contenido relevante según su rol y preferencias.
- **REQ-13:** Debe existir un historial de ediciones por contenido.

3.4 Gestión de Campañas

3.4.1 Descripción y Prioridad

Este módulo permite la planificación, ejecución y seguimiento de campañas de donación o sensibilización. Las campañas podrán estar asociadas a metas financieras o de participación y estarán vinculadas con donaciones y voluntarios.

Prioridad: Alta

3.4.2 Secuencias de estímulo/respuesta

1. Un administrador crea una nueva campaña.
2. El sistema permite definir metas, duración, responsables y recursos.
3. Se notifica a los usuarios registrados y se visualiza en el dashboard.
4. A medida que se reciben donaciones o participación, la campaña se actualiza.

3.4.3 Requisitos funcionales

- **REQ-14:** Permitir la creación y edición de campañas con campos como nombre, objetivo, fecha, responsable y estado.
- **REQ-15:** Asociar donaciones y voluntarios a campañas específicas.
- **REQ-16:** Generar reportes automáticos con avances por campaña.
- **REQ-17:** Permitir finalizar campañas y archivarlas para consulta histórica.

3.5 Gestión de Eventos

3.5.1 Descripción y Prioridad

Los eventos permiten convocar a voluntarios y donantes para actividades presenciales o virtuales. El módulo incluye la inscripción, gestión logística y seguimiento posterior.

Prioridad: Media

3.5.2 Secuencias de estímulo/respuesta

1. Un coordinador crea un evento.
2. Los voluntarios reciben notificación y pueden inscribirse.
3. El sistema registra asistencias y genera seguimiento posterior.

3.5.3 Requisitos funcionales

- **REQ-18:** Crear eventos con fecha, ubicación (física o virtual), descripción y capacidad máxima.
- **REQ-19:** Permitir inscripción, cancelación y envío de recordatorios automáticos.
- **REQ-20:** Registrar asistencia y participación efectiva de los voluntarios.
- **REQ-21:** Asociar eventos con campañas o programas existentes.

3.6 Reportes y Analítica

3.6.1 Descripción y Prioridad

La plataforma debe ofrecer paneles de control y reportes con indicadores clave: volumen de donaciones, usuarios activos, campañas exitosas, participación de voluntarios, entre otros.

Prioridad: Alta

3.6.2 Secuencias de estímulo/respuesta

1. Un administrador accede al panel de métricas.
2. El sistema genera visualizaciones en tiempo real y permite exportarlas.
3. Se configuran alertas o informes recurrentes.

3.6.3 Requisitos funcionales

- **REQ-22:** Generar reportes visuales filtrables por fecha, campaña o tipo de usuario.
- **REQ-23:** Permitir exportación a formatos como PDF o Excel.
- **REQ-24:** Integrar herramientas de BI como Power BI o Tableau.
- **REQ-25:** Incluir KPIs configurables y alertas automáticas.

3.7 Gestión de Roles y Permisos

3.7.1 Descripción y Prioridad

Permite definir roles específicos (como administrador, coordinador, donante, voluntario) con permisos diferenciados de acceso, edición y visualización.

Prioridad: Alta

3.7.2 Requisitos funcionales

- **REQ-26:** Definir roles personalizados con permisos específicos.
- **REQ-27:** Asignar y modificar roles desde el panel de usuarios.
- **REQ-28:** Garantizar separación de privilegios críticos para la seguridad.
- **REQ-29:** Integración con el sistema de autenticación y control de sesión.

3.8 Gestión de Programas

3.8.1 Descripción y Prioridad

Este módulo permite agrupar campañas, actividades y eventos bajo un objetivo común (programa), como “Educación para todos” o “Salud comunitaria”.

Prioridad: Media

3.8.2 Requisitos funcionales

- **REQ-30:** Crear programas que agrupen múltiples campañas o eventos.
- **REQ-31:** Asociar indicadores y metas generales a cada programa.
- **REQ-32:** Visualizar progreso consolidado del programa.

4. Requisitos de la interfaz

4.1 Interfaces de Usuario

- Aplicación web con panel de administración:

Diseñada para ser intuitiva y de fácil navegación, esta interfaz permitirá a los administradores gestionar usuarios, roles, permisos, contenidos y generar reportes personalizados. Se incluirán funcionalidades como filtrado por fechas, exportación de datos en distintos formatos y control de acceso según rol.

- Dashboard interactivo con gráficos y métricas de impacto:

Permitirá visualizar en tiempo real indicadores clave, como número de usuarios activos, cantidad de donaciones recibidas, actividades voluntarias realizadas, y participación en eventos.

4.2 Interfaces de hardware

- Servidores en la nube:

Se utilizarán servicios como AWS, Azure o Google Cloud para garantizar escalabilidad automática, balanceo de carga, recuperación ante desastres, copias de seguridad automatizadas, y procesamiento eficiente de grandes volúmenes de datos.

- Dispositivos móviles y computadoras:

El sistema será accesible desde navegadores modernos, con diseño responsivo (mobile-first), garantizando compatibilidad con Android, iOS, Windows y macOS, asegurando una experiencia de usuario óptima en distintos tamaños de pantalla.

4.3 Interfaces de software

- Integración con plataformas de pago y CRM:

Se integrará con pasarelas de pago para gestionar donaciones y pagos. También se conectará con CRMs para seguimiento de relaciones con donantes y voluntarios.

- API Gateway:

Servirá como punto de entrada único para acceder a servicios internos relacionados con la gestión de usuarios, actividades, reportes y datos de donantes. Permitirá autenticación, limitación de velocidad, y control de acceso granular a servicios.

Otros requisitos No funcionales

4.4 Requerimientos de desempeño

- Baja latencia en APIs (<200ms):

Las consultas estándar (como obtener usuarios o actividades recientes) deberán responder en menos de 200 milisegundos, gracias a técnicas como caché en memoria (Redis), índices bien optimizados en bases de datos y balanceo de carga.

- Procesamiento masivo de datos (10,000 registros/minuto):

El sistema deberá soportar operaciones de carga, transformación y análisis de grandes volúmenes de datos en tiempo casi real, aprovechando herramientas como ETL automatizados, particionado de tablas y procesamiento paralelo

4.5 Requisitos de seguridad

- Cifrado de datos:

Toda la información sensible será cifrada en tránsito usando HTTPS con TLS 1.2/1.3 y en reposo mediante técnicas como AES-256. Se incluirán auditorías periódicas de seguridad y cumplimiento de normativas como GDPR o ISO/IEC 27001.

4.6 Atributos de calidad del software

- Escalabilidad:

Arquitectura basada en microservicios, uso de contenedores (Docker) y orquestadores como Kubernetes para permitir el escalado horizontal y vertical sin afectar la disponibilidad del sistema.

- Alta disponibilidad:

Despliegue en múltiples zonas de disponibilidad (AZ) dentro de la nube, balanceo de carga redundante y sistemas de failover automático aseguran un tiempo de actividad de al menos 99.9%.

- Mantenibilidad:

Código limpio, estructurado y documentado, siguiendo principios SOLID, con pruebas automatizadas y CI/CD para facilitar despliegues y mantenimiento continuo sin afectar a los usuarios finales.

5. Otros Requerimientos

- Compatibilidad con navegadores modernos y dispositivos móviles.
- Cumplimiento con normativas de privacidad como GDPR.

6. Características del MVP

- **Arquitectura basada en microservicios**
 - Para gestionar donantes, voluntarios y programas de forma modular y escalable.
- **API Gateway**
 - Control de acceso centralizado y seguridad en el intercambio de datos.
- **Data Warehouse en la nube**
 - Almacenamiento de datos estructurado y preparado para análisis.
- **Dashboard interactivo**
 - Visualización de métricas y reportes importantes para la toma de decisiones.

7. Pruebas

7.1 Pruebas funcionales de servicios

Los servicios de `donor`, `volunteer` y `analytics` están cubiertos con pruebas unitarias que aseguran el cumplimiento de requisitos funcionales.

- Cada método de servicio cuenta con pruebas específicas.
- Se utilizan `pytest` y `mock` para pruebas desacopladas del backend.

7.2 Pruebas de integración

- Valida la interacción entre los servicios (`donor ↔ analytics`, `volunteer ↔ donor`).
- Pruebas alojadas en `tests/integration`.
- Datos de prueba gestionados con `mock_data.py`.