



RedVital

Conecta tu Sangre, Salva una Vida

*Pilar Bourg, Olivia Gallego, Elena Morales,
Alejandra O'Shea, Marta Sánchez y Josefina Silva*

Sitemap

El sitemap de RedVital representa la estructura completa de navegación del sistema y sirve como base para organizar las funcionalidades según cada rol. Su objetivo es garantizar que donantes, hospitales, administradores y receptores accedan de forma clara y jerárquica a las funciones que les corresponden, alineadas con los requisitos funcionales del proyecto.

La página principal (Inicio/Home) actúa como punto de acceso común. Desde ella se despliegan secciones informativas (“Sobre nosotros”, “Misión y objetivos”, “Equipo”) y los accesos a los módulos de autenticación (“Registrarse” e “Iniciar sesión”).

Una vez iniciado sesión, el usuario accede a un área específica dependiendo de su rol:

Donante

Incluye las funcionalidades destinadas al usuario que ofrece donaciones de sangre:

- **Perfil del donante:** datos personales, grupo sanguíneo e información médica relevante.
- **Disponibilidad:** gestión de fechas, horarios y centros donde puede donar.
- **Alertas y solicitudes:** listado de notificaciones recibidas por solicitudes compatibles.
- **Historial de donaciones:** registro histórico de donaciones previas y convocatorias.

Este módulo da soporte directo a RF1–RF5, permitiendo gestionar disponibilidad y responder solicitudes.

Hospital

Es la sección operativa del personal médico o de enfermería:

- **Crear solicitud de sangre:** registro de tipo sanguíneo, cantidad y urgencia.
- **Buscar donantes compatibles:** listado filtrado por proximidad, tipo sanguíneo y disponibilidad.
- **Enviar notificaciones:** envío de convocatorias a donantes elegibles.
- **Actualizar estado de solicitud:** gestión del ciclo de vida (pendiente, cubierta, cancelada).
- **Registrar resultados de donaciones:** validación final de cada aporte recibido.
- **Panel de estadísticas hospitalarias:** indicadores de actividad interna y tiempos de respuesta.

Este módulo abarca RF6–RF10 y RF20.

Administrador

Administra globalmente la plataforma y asegura su correcto funcionamiento:

- **Gestión de usuarios y roles:** creación, edición, validación y eliminación de cuentas.
- **Gestión de centros y bancos de sangre:** registro de hospitales, datos operativos y ubicación.
- **Mandar notificaciones globales:** avisos dirigidos a todos los usuarios del sistema.
- **Bloquear / suspender cuentas:** control del cumplimiento de políticas de uso.
- **Panel de estadísticas globales:** datos agregados por región, periodo y grupo sanguíneo.

Corresponde a RF12–RF15 y RF21.

Receptor

Pensada para pacientes o sus representantes:

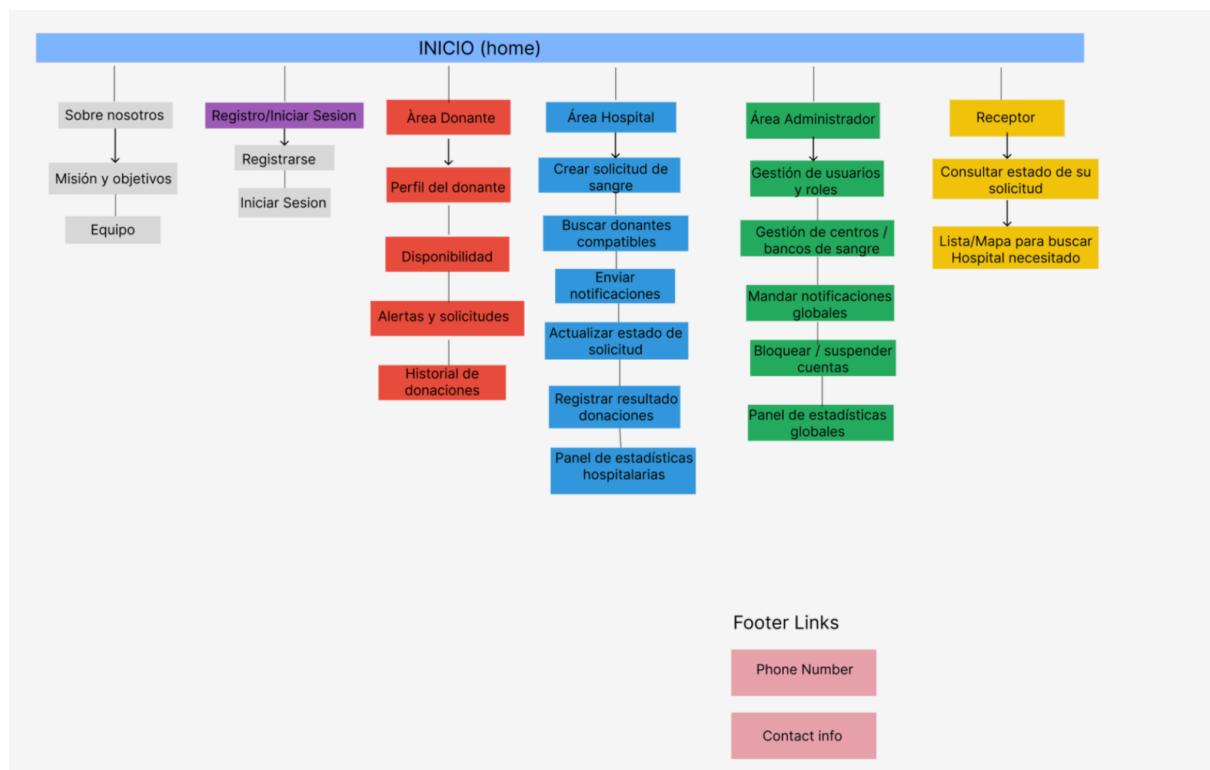
- **Consultar estado de solicitud:** información sobre si su transfusión está cubierta o en proceso.
- **Lista/mapa de hospitales:** visualización de centros con disponibilidad compatible.

Esta parte cubre RF11.

Footer

Incluye elementos de soporte como:

- Información de contacto
- Teléfono y medios de asistencia



Wireframe

Página principal

La página de inicio presenta:

- Un banner principal con el mensaje central de la plataforma.
- Secciones informativas sobre la importancia de donar sangre.
- Tarjetas de acceso rápido a funciones destacadas.
- Footer con información de contacto y ubicación.

Esta pantalla introduce al usuario en la misión del proyecto y facilita navegaciones rápidas.

Pantallas de Registro y Login

Ambas interfaces siguen un diseño limpio y simplificado. Incluyen:

- Formulario básico de autenticación.
- Identidad visual de RedVital (logo y encabezados).
- Botones de navegación entre "Iniciar sesión" y "Registrarse".

Estas pantallas forman el punto de entrada para los distintos roles del sistema.

Hospital

Los wireframes del área hospitalaria reflejan las operaciones médicas más relevantes:

- **Crear solicitud:** formulario estructurado para tipo de sangre, urgencia y cantidad.
- **Buscar donantes compatibles:** tabla con filtros y acciones de notificación.
- **Actualizar estado de solicitud:** listado para marcar avance del proceso.
- **Registrar resultados:** formulario para registrar donaciones efectivas.
- **Panel de estadísticas:** gráficos de barras y tablas con indicadores clave.

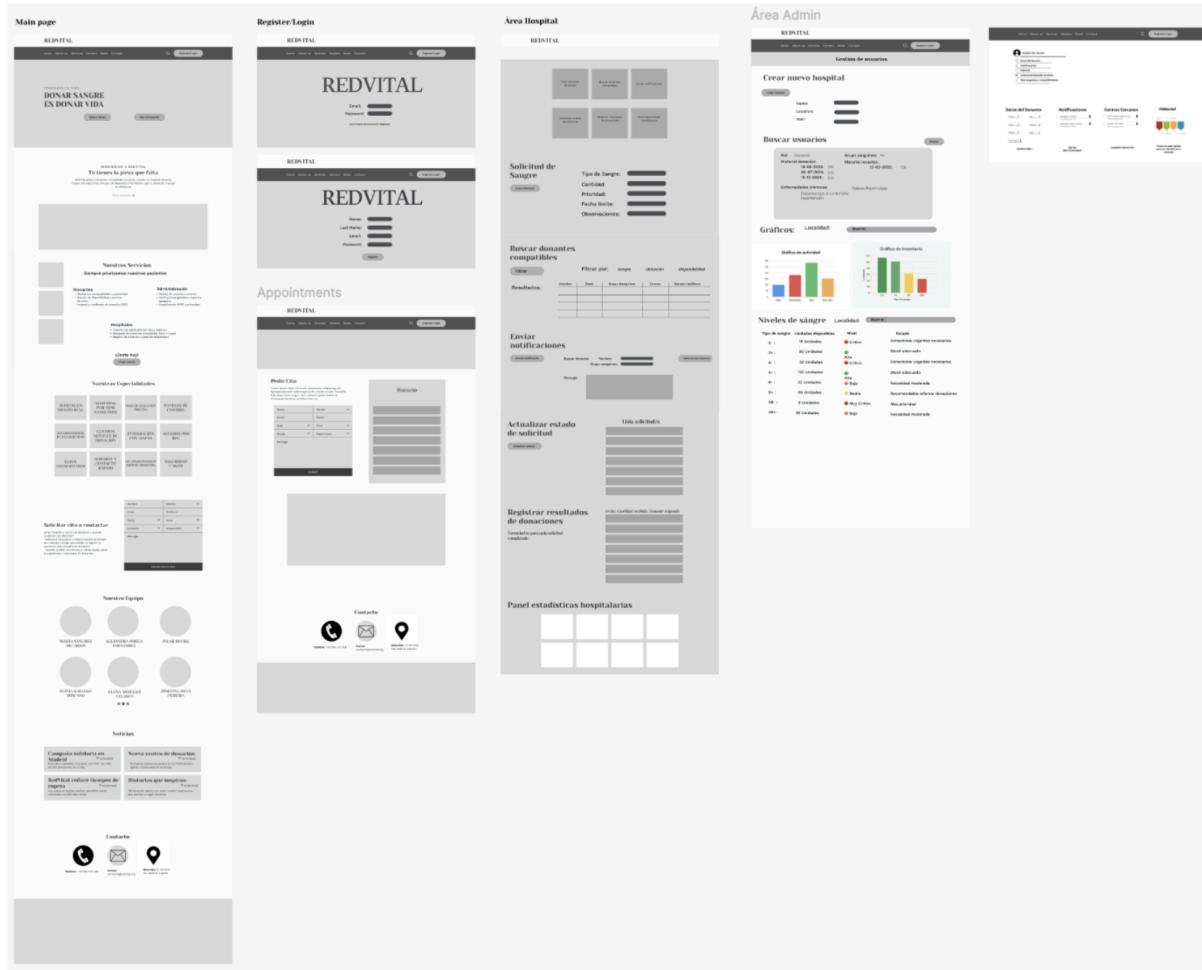
Visualmente, estas pantallas resaltan la funcionalidad por encima del diseño estético, garantizando claridad operativa.

Administrador

Incluye interfaces más orientadas a gestión:

- **Registrar hospital:** formulario con datos de centro sanitario y ubicación.
- **Buscar usuarios:** tabla con filtros para administración de cuentas.
- **Gráficos globales:** visualización de niveles de sangre, actividad por tipo sanguíneo y estadísticas agregadas.

Los wireframes priorizan la usabilidad, usando bloques bien definidos y secciones claramente diferenciadas para evitar sobrecarga visual.



Arquitectura del Sistema

La arquitectura de RedVital se diseñó siguiendo un enfoque modular y orientado a componentes, con el objetivo de garantizar escalabilidad, mantenibilidad y una separación clara entre la lógica de negocio, la presentación y el manejo de datos. El sistema se estructura en una arquitectura **cliente–servidor**, donde el frontend y el backend funcionan como capas independientes que se comunican mediante API REST.

1. Arquitectura General

El sistema está dividido en tres capas principales:

a) Capa de Presentación (Frontend)

Desarrollada en HTML, CSS y JavaScript, es responsable de la interacción directa con el usuario. Incluye:

- Interfaces para cada rol (donante, hospital, receptor, administrador).
- Formulario de registro/login.
- Módulos de creación de solicitudes, gestión de disponibilidad y estadísticas.
- Integración de mapas y tablas dinámicas.

El frontend consume los endpoints del backend para obtener o enviar información, sin manejar lógica crítica por sí mismo.

b) Capa de Lógica de Negocio (Backend)

Implementada en Node.js con Express, esta capa:

- Gestiona la autenticación y autorización por rol.
- Procesa solicitudes hospitalarias (creación, actualización, compatibilidad).
- Ejecuta el algoritmo de matching entre donantes y solicitudes (tipo sanguíneo + proximidad + disponibilidad).
- Controla el ciclo de vida de notificaciones y convocatorias.
- Calcula estadísticas hospitalarias y globales.
- Mantiene la comunicación en tiempo real a través de sockets para métricas actualizadas.

El backend actúa como núcleo del sistema y garantiza la consistencia de los datos gestionados.

c) Capa de Datos (Base de Datos)

La base de datos almacena toda la información estructurada del proyecto, incluyendo:

- Usuarios, roles y estados de cuenta.
- Donantes, disponibilidad y citas.
- Solicitudes de sangre y donaciones.
- Notificaciones y su estado de lectura/aceptación.

- Inventario por hospital.
- Centros sanitarios y personal médico.

La estructura responde directamente al diagrama ER definido previamente, asegurando integridad referencial y soporte a las operaciones críticas descritas en los requisitos funcionales.

2. Arquitectura por Roles

La arquitectura de RedVital se diseñó para soportar flujos diferenciados según el tipo de usuario. Esta separación permite:

- Restringir accesos mediante middleware.
- Presentar interfaces adaptadas a cada necesidad.
- Optimizar el rendimiento evitando carga de datos innecesarios.

Los roles gestionados en la arquitectura son:

a) Donante

Interactúa principalmente con:

- Módulo de disponibilidad
- Sistema de notificaciones
- Historial personal
- Actualización de perfil

b) Hospital

Accede a módulos con mayor carga operativa:

- Gestión de solicitudes
- Filtrado y notificación a donantes
- Registro de donaciones
- Estadísticas hospitalarias

c) Receptor

Tiene un flujo más simple, centrado en:

- Consulta del estado de su solicitud
- Mapa de hospitales con stock compatible

d) Administrador

Tiene acceso global a:

- Gestión de usuarios
- Gestión de centros
- Estadísticas agregadas
- Control del flujo de notificaciones

Cada rol posee rutas, controladores y vistas específicas, garantizando seguridad y claridad en la experiencia de usuario.

3. Arquitectura de Comunicación

La comunicación entre capas se realiza mediante:

a) API REST

El backend expone endpoints REST organizados por módulos (donantes, hospitales, notificaciones, citas).

El frontend interactúa únicamente a través de estas rutas, lo que favorece:

- Independencia tecnológica
- Posibilidad de reemplazar el frontend sin afectar al backend
- Futuras mejoras como migrar a apps móviles

b) WebSockets

Usados para actualizaciones en tiempo real, especialmente en:

- Estadísticas hospitalarias

- Panel del administrador
- Confirmaciones inmediatas de solicitudes y donaciones

Esto permite que hospitales vean cambios sin necesidad de recargar la página.

4. Arquitectura de Datos

El diseño de datos sigue principios de normalización y relaciones uno-a-uno / uno-a-muchos según el ER.

Algunos puntos clave:

- **Usuarios** se relacionan 1–1 con Donante, Hospital o Admin.
- **Disponibilidad, Citas** y **Notificaciones** tienen relaciones 1–n.
- **Solicitudes** se relacionan con **Donaciones** para cubrir su ciclo completo.
- **Inventario_sangre** mantiene la información por hospital y tipo de sangre.

5. Arquitectura de Seguridad

Incluye:

- Sistema de autenticación mediante JSON Web Tokens (JWT).
- Autorización por rol (middleware).
- Encriptación de contraseñas (bcrypt).
- Validación de datos en todos los endpoints.
- Restricción de endpoints según tipo de usuario.
- Control de cuentas suspendidas o no válidas.

Esto garantiza cumplimiento de los requisitos RF12, RF15 y la protección de datos personales sensibles.

6. Arquitectura de Escalabilidad y Mantenibilidad

Para asegurar escalabilidad futura, el sistema usa:

- Separación completa frontend–backend.
- Rutas, controladores y servicios organizados por módulos.
- Uso de sockets solo en módulos que lo requieren.
- Posibilidad de integrar un sistema de colas (futuras versiones) para notificaciones masivas.

El diseño actual permite extender el sistema sin romper funcionalidades existentes.