

**Solidaridad activa: plataforma de donaciones de alimentos para el municipio de Turbo**

Yeisson Mena Bogallo

Asesor

Juan Carlos Lopez Henao

Universidad Nacional Abierta y a Distancia UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI

Ingeniería de sistemas

2025

## Resumen

La plataforma Solidaridad activa surge como respuesta a la problemática de inseguridad alimentaria en el municipio de Turbo, donde muchas familias de bajos ingresos carecen de acceso a alimentos básicos. El proyecto tiene como objetivo desarrollar una plataforma web que conecte a donantes locales con personas en situación de necesidad, facilitando la gestión eficiente y transparente de las donaciones. Para su implementación, se utilizaron tecnologías como HTML, CSS y JavaScript, permitiendo el registro de usuarios, la administración de inventarios y la coordinación de la entrega de alimentos. El desarrollo se realizó mediante ciclos cortos y flexibles, lo que permitió avanzar de manera progresiva y adaptativa. Se espera que esta iniciativa no solo contribuya a disminuir la inseguridad alimentaria, sino que también fortalezca los vínculos de solidaridad y participación comunitaria.

***Palabras clave:*** inseguridad alimentaria, donaciones, plataforma web, Turbo, solidaridad

### **Abstract**

The platform Solidaridad activa emerges as a response to the food insecurity problem in the municipality of Turbo, where many low-income families lack access to basic food supplies. This project aims to develop a web-based platform that connects local food donors with people in need, facilitating the efficient and transparent management of donations. Technologies such as HTML, CSS, and JavaScript were used to implement features including user registration, inventory management, and food distribution coordination. The development process followed short and flexible cycles, allowing for progressive and adaptive advances. This initiative is expected not only to help reduce food insecurity but also to strengthen community solidarity and participation.

***Keywords:*** food insecurity, donations, web platform, Turbo, solidarity

## Tabla de Contenido

Introducción .....	11
Justificación .....	13
Objetivos .....	14
Objetivo General.....	14
Objetivos Específicos .....	14
Planteamiento del Problema .....	15
Requisitos.....	18
Requisitos Funcionales .....	18
Registro y Autenticación de Usuarios .....	18
Gestión de Campañas de Donación .....	18
Procesamiento de Pagos .....	18
Gestión de Perfiles de Usuarios .....	19
Informes y Estadísticas .....	19
Gestión de Beneficiarios .....	19
Criterios de Elegibilidad: .....	19
Proceso de Verificación:.....	19
Flujo de Distribución Logística: .....	20
Seguridad y Privacidad .....	21
Requisitos No Funcionales .....	22
Usabilidad .....	22
Rendimiento.....	22

Seguridad .....	22
Mantenimiento .....	22
Compatibilidad .....	23
Cronograma del Proyecto .....	24
Diseño y Desarrollo de la Plataforma .....	25
Aplicación de la metodología ágil en el proyecto .....	25
Diseño UI/UX.....	27
Simplicidad y Claridad Visual:.....	28
Accesibilidad para el Contexto Local:.....	30
Navegación Intuitiva:.....	31
Estética Amigable y Cercana: .....	31
Feedback Visual y Accesibilidad:.....	32
Profundización en la Arquitectura de Software:.....	33
Codificación: Metodología y Desarrollo .....	33
Stack Tecnológico .....	33
Frontend:.....	34
Backend: .....	35
Estructura del Proyecto.....	37
Integración entre Frontend y Backend.....	38
Gestión de Usuarios y Seguridad.....	39
Flujo funcional de la plataforma .....	40
Registro e inicio de sesión.....	40
Formulario de donación .....	40

Historial de donaciones .....	40
Arquitectura del proyecto.....	40
Funcionalidades Clave Desarrolladas .....	40
Buenas prácticas aplicadas.....	41
Documentación Técnica.....	42
Manual de Instalación y Configuración del Entorno de Desarrollo: .....	42
Modelo Entidad-Relación (ER) de la Plataforma.....	44
Plan de Pruebas del Proyecto: Solidaridad Activa.....	45
Objetivos de las Pruebas .....	45
Tipos de Pruebas.....	45
Casos de Prueba.....	46
Pruebas de Seguridad: .....	48
Herramientas utilizadas:.....	49
Resultados de las pruebas: .....	49
Validación del entorno de pruebas:.....	50
Pruebas de Usabilidad con Usuarios Reales .....	50
Metodología propuesta: .....	50
Métricas evaluadas:.....	51
Escala SUS:.....	51
Contexto de prueba: .....	51
Desarrollo de la prueba: .....	51
Resultados esperados: .....	51
Análisis de Resultados de Usabilidad .....	52

Herramientas de Prueba.....	52
Criterios de Aceptación .....	54
Análisis de Riesgos del Proyecto: Solidaridad Activa.....	54
Identificación de Riesgos y Estrategias de Mitigación.....	54
Revisión Continua de Riesgos .....	59
Resultados Obtenidos y Conclusiones .....	59
Resultados Obtenidos.....	59
Conclusiones .....	63
Posibles Mejoras Futuras .....	64
Expansión de Tipos de Donación .....	64
Mejora en la Experiencia de Usuario (UX/UI).....	65
Optimización de la Seguridad y la Privacidad.....	65
Optimización de la Base de Datos y Funcionalidades Avanzadas .....	65
Integración con Otras Plataformas y Servicios.....	66
Escalabilidad del Proyecto.....	66
Conclusiones .....	68
Medición de Impacto .....	69
Bibliografía .....	71

## Lista de Figuras

<b>Figura 1</b> <i>Diagrama de flujo del proceso de donación en la plataforma</i> .....	21
<b>Figura 2</b> <i>Lista jerárquica de requisitos funcionales y no funcionales del proyecto</i> .....	23
<b>Figura 3</b> <i>Cronograma del proyecto según la planificación tipo Gantt</i> .....	24
<b>Figura 4</b> <i>Pantalla de inicio de sesión de la plataforma</i> .....	28
<b>Figura 5</b> <i>Formulario de registro de usuario en la plataforma</i> .....	29
<b>Figura 6</b> <i>Ventana de gestión de cuenta para usuarios autenticados</i> .....	29
<b>Figura 7</b> <i>Pantalla principal de navegación de la plataforma</i> .....	31
<b>Figura 8</b> <i>Formulario para registrar una donación de alimentos</i> .....	32
<b>Figura 9</b> <i>Fragmento de código CSS utilizado en el formulario de donaciones</i> .....	34
<b>Figura 10</b> <i>Fragmento de código HTML del formulario de donaciones</i> .....	35
<b>Figura 11</b> <i>Diagrama de arquitectura</i> .....	36
<b>Figura 12</b> <i>Estructura de carpetas del proyecto en Visual Studio Code</i> .....	37
<b>Figura 13</b> <i>Archivo de configuración de base de datos en el proyecto</i> .....	38
<b>Figura 14</b> <i>Modelo Entidad-Relación (ER) de la plataforma "Solidaridad Activa"</i> .....	44
<b>Figura 15</b> <i>Formulario de donación funcional vinculado a la base de datos</i> .....	61
<b>Figura 16</b> <i>Visualización del historial de donaciones en la plataforma</i> .....	62
<b>Figura 17</b> <i>Vista funcional de la página de inicio de sesión en pruebas</i> .....	63

### Lista de Tablas

**Tabla 1** Casos de prueba de funcionalidades clave de la plataforma..... 46

**Tabla 2** Riesgos identificados en el desarrollo del proyecto y estrategias de mitigación..... 54

**Lista de Apéndices**

<b>Apéndice A</b> <i>Fragmento de código PHP para conexión a base de datos</i> .....	73
--	----

## Introducción

En el municipio de Turbo, la inseguridad alimentaria es un problema persistente que afecta a numerosos hogares de bajos ingresos, limitando su acceso a alimentos básicos necesarios para una vida digna. A pesar de los esfuerzos locales y comunitarios para mitigar esta situación, existen barreras significativas que dificultan la distribución equitativa de los recursos alimentarios. Entre estas barreras se encuentra la falta de una herramienta efectiva que conecte a las personas dispuestas a donar alimentos con aquellas que los necesitan urgentemente.

En este contexto, surge la iniciativa "Solidaridad Activa", una plataforma web diseñada para facilitar la coordinación de donaciones de alimentos dentro de la comunidad. Esta plataforma busca proporcionar un canal digital eficiente y transparente que permita a los donantes locales ofrecer sus excedentes de alimentos, mientras que los beneficiarios puedan acceder a estos recursos de manera justa y organizada. El objetivo es no solo mejorar la gestión de donaciones, sino también fomentar la solidaridad y la responsabilidad compartida entre los habitantes de Turbo.

El desarrollo de esta plataforma web se lleva a cabo utilizando tecnologías como HTML, CSS y JavaScript, que permiten crear una solución accesible y escalable. El proyecto sigue una metodología de desarrollo ágil, que permite la iteración rápida y la adaptación continua basada en la retroalimentación de los usuarios.

Este proyecto está dirigido principalmente a la Alcaldía de Turbo, a través de sus programas sociales y de apoyo a la comunidad, con el fin de colaborar en la mejora de la distribución de alimentos en la región. La plataforma "Solidaridad Activa" se presenta como una herramienta para fortalecer los esfuerzos gubernamentales y comunitarios en la lucha contra la inseguridad alimentaria, ampliando el alcance de sus iniciativas y optimizando la logística de las

donaciones. A lo largo de este documento, se presenta un análisis detallado del problema, los objetivos del proyecto, la metodología empleada, los resultados esperados, y las recomendaciones para futuros desarrollos.

## Justificación

Según el Plan Territorial de Salud del Distrito de Turbo (2024–2027), el 41 % de los hogares presentan riesgo severo de inseguridad alimentaria, superando el promedio departamental del 13 %. Además, el informe “Antioquia Cómo Vamos” (2023) indica que la región de Urabá concentra algunas de las tasas más altas del departamento, con más del 65 % de hogares afectados. Esta situación se agrava con los 2.251 casos de desnutrición aguda reportados en menores de cinco años en Antioquia durante 2023 (Carvalho, 2023). Estos datos reflejan la urgencia de implementar soluciones tecnológicas que optimicen la distribución alimentaria y promuevan la participación ciudadana en el municipio de Turbo.

En la resolución de problemas referentes a la inseguridad alimentaria en el municipio de Turbo, estas exigen medidas innovadoras que faciliten la participación ciudadana de carácter solidario. Solidaridad Activa presenta una propuesta concreta que permite darle un giro al concepto de donaciones y hacer de estas una forma más ordenada, moderna y segura de realizar, superando las barreras actuales de desorganización e informalidad en la entrega de ayudas.

Este proyecto aporta un valor diferencial al implementar una plataforma digital adaptada al contexto local, con una interfaz intuitiva y accesible para usuarios de diversos niveles tecnológicos. Además, promueve la inclusión social al permitir que tanto personas naturales como organizaciones se conviertan en actores activos en la lucha contra el hambre.

Otro elemento importante dentro del proyecto es que la solución está desarrollada con el uso de tecnologías de software libre, esto es muy significativo frente al mantenimiento, escalabilidad y adopción futura por parte de instituciones como la Alcaldía de Turbo u organizaciones comunitarias.

## Objetivos

### Objetivo General

Desarrollar una plataforma web de donaciones de alimentos, denominada "Solidaridad Activa", que conecte a donantes y personas necesitadas, facilitando la gestión de las donaciones de manera eficiente, segura y transparente.

### Objetivos Específicos

Crear una interfaz de usuario intuitiva y accesible que facilite la navegación para los administradores de campañas y los donantes, garantizando una experiencia amigable para todo tipo de usuarios.

Desarrollar un sistema de registro y autenticación para usuarios y administradores que garantice la seguridad de la información y el acceso controlado a las funcionalidades de la plataforma.

Desarrollar funcionalidades para la gestión de campañas de donación de alimentos, incluyendo la descripción de campañas, definición de metas y actualización del progreso.

Proporcionar herramientas de informes y análisis que permitan a las organizaciones y administradores evaluar el desempeño de las campañas de donación de alimentos y la participación de los usuarios.

Garantizar la transparencia y la confianza de los donantes mediante mecanismos de trazabilidad de las donaciones y acceso a información clara sobre el uso de los recursos.

Optimizar el rendimiento y la escalabilidad de la plataforma para asegurar su correcto funcionamiento y permitir su crecimiento futuro.

## Planteamiento del Problema

A pesar del creciente interés en apoyar causas sociales y humanitarias, muchas organizaciones sin fines de lucro y grupos de base enfrentan dificultades significativas en la recaudación de fondos debido a la falta de herramientas adecuadas y procesos eficientes. Estos problemas se manifiestan de las siguientes maneras:

1. **Métodos de Recaudación Ineficientes:** Las organizaciones a menudo dependen de métodos tradicionales y menos efectivos para recaudar fondos, como eventos presenciales, campañas de correo electrónico o redes sociales sin una plataforma centralizada para gestionar y monitorear las donaciones. Estos métodos pueden resultar en una gestión fragmentada y una menor eficiencia en la recaudación. El uso de métodos obsoletos de recaudación ha sido un problema recurrente en proyectos de donación, tal como se destaca en proyectos anteriores que han abordado la necesidad de digitalizar el proceso de donaciones para mejorar la eficiencia (Collaguazo Llumiyinga, 2015).
2. **Acceso Limitado para Donantes:** La falta de una plataforma en línea accesible limita la capacidad de las organizaciones para llegar a una audiencia más amplia. Los donantes potenciales pueden encontrar inconvenientes en la realización de donaciones si el proceso no es intuitivo o si la plataforma no está disponible para donaciones en línea. “Cambio fácil.com” y otros proyectos similares han resaltado cómo las plataformas en línea pueden mejorar el alcance a audiencias globales, algo que Solidaridad Activa busca implementar para que los donantes puedan acceder fácilmente desde cualquier lugar (Quiroga Taborda, 2020).

A diferencia de plataformas como “CambioFácil.com”, centradas en el intercambio general de bienes y no en un enfoque local o específico de seguridad

alimentaria, Solidaridad Activa ha sido diseñada para responder a las condiciones sociales, tecnológicas y alimentarias del municipio de Turbo. Esta plataforma cuenta con funcionalidades adaptadas a dispositivos de gama baja, trazabilidad de cada donación, y un sistema modular de código abierto que permite su integración con organismos locales (Collaguazo Llumiquinga, 2015).

3. **Problemas de Transparencia y Confianza:** Sin un sistema que proporcione informes claros y detallados sobre el uso de los fondos recaudados, las organizaciones pueden enfrentar problemas de confianza por parte de los donantes. La falta de transparencia puede desalentar a los donantes potenciales y afectar la reputación de la organización.
4. **Gestión Compleja de Campañas:** La administración de campañas de donación puede ser compleja y consumir mucho tiempo si se realiza de manera manual o sin herramientas adecuadas. Esto puede llevar a errores, retrasos y una gestión ineficaz de los recursos disponibles. Mena Hidalgo et al. (2021) señalan que una plataforma que integre funciones de gestión y control en un solo sistema centralizado permite una mayor eficacia, minimizando errores y tiempo de administración, un objetivo clave que Solidaridad Activa también busca alcanzar.
5. **Falta de Integración de Herramientas:** Muchas organizaciones carecen de una solución integral que combine el registro de usuarios, la gestión de campañas, el procesamiento de pagos y la generación de informes en una sola plataforma. La ausencia de estas herramientas integradas puede resultar en un manejo ineficiente de las donaciones y en una experiencia frustrante tanto para los administradores como para los donantes. Estudios recientes en proyectos de donación han demostrado la ventaja de sistemas

integrados que combinen varios aspectos de la gestión de campañas y donaciones, lo cual es un problema que Solidaridad Activa busca resolver (Álvarez Ríos et al., 2021).

Solidaridad Activa busca resolver estos problemas al proporcionar una plataforma de donaciones en línea que centraliza y simplifica el proceso de recaudación de fondos. Al abordar estas cuestiones, el proyecto tiene como objetivo mejorar la eficiencia en la recaudación de fondos, ampliar el alcance de las campañas, fomentar la transparencia y, en última instancia, apoyar a las organizaciones en su misión de generar un impacto positivo en la comunidad.

## Requisitos

### Requisitos Funcionales

#### *Registro y Autenticación de Usuarios*

- Registro de Usuarios: Los usuarios deben poder registrarse en la plataforma proporcionando información básica como nombre, correo electrónico y una contraseña segura.
- Inicio de Sesión: Los usuarios registrados deben poder iniciar sesión con su correo electrónico y contraseña.
- Recuperación de Contraseña: Los usuarios deben poder recuperar o restablecer su contraseña en caso de olvido.
- Roles de Usuario: La plataforma debe diferenciar entre usuarios donantes y administradores de campañas, con permisos específicos para cada rol.

#### *Gestión de Campañas de Donación*

- Creación de Campañas: Los administradores deben poder crear nuevas campañas de donación, incluyendo detalles como nombre de la campaña, descripción, metas de recaudación, fecha de inicio y fin.
- Edición y Eliminación: Los administradores deben poder editar o eliminar campañas existentes.
- Visualización de Campañas: Los usuarios deben poder ver una lista de campañas activas y acceder a detalles específicos de cada campaña.

#### *Procesamiento de Pagos*

- Integración de Métodos de Pago: La plataforma debe permitir a los donantes realizar pagos a través de métodos seguros como tarjetas de crédito, transferencias bancarias, y servicios de pago en línea.

- Confirmación de Pago: Los donantes deben recibir una confirmación de su contribución por correo electrónico y en la plataforma.

### ***Gestión de Perfiles de Usuarios***

- Perfil de Usuario: Los usuarios deben tener un perfil donde puedan ver su información personal, historial de donaciones y campañas en las que han participado.
- Actualización de Perfil: Los usuarios deben poder actualizar su información personal y preferencias de comunicación.

### ***Informes y Estadísticas***

- Informes de Campañas: Los administradores deben poder generar informes sobre el rendimiento de las campañas, incluyendo total recaudado, número de donantes, y progreso hacia las metas.
- Estadísticas de Donaciones: Los administradores deben poder ver estadísticas sobre donaciones realizadas, incluyendo tendencias y patrones.

### ***Gestión de Beneficiarios***

#### ***Criterios de Elegibilidad:***

- Residir en el municipio de Turbo, Antioquia.
- Pertenecer a un hogar clasificado en nivel 1 o 2 del Sisbén o ser referenciado por organizaciones comunitarias.
- No estar registrado en más de una ocasión como beneficiario activo en el sistema.

#### ***Proceso de Verificación:***

- Verificación documental al registrarse (adjuntar foto de documento de identidad y comprobante de residencia).

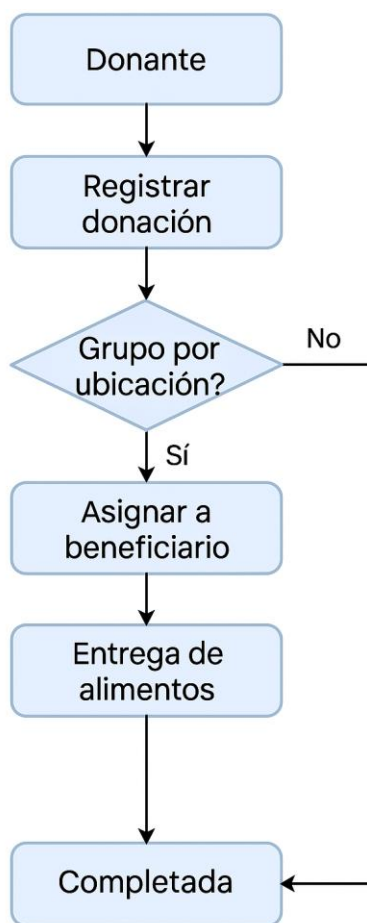
- Validación manual por parte de un administrador antes de la activación de la cuenta de beneficiario.

***Flujo de Distribución Logística:***

- Los donantes registran su donación indicando tipo, cantidad y fecha disponible.
- Los administradores agrupan las donaciones por ubicación.
- Se asignan las donaciones a beneficiarios en lista prioritaria (por vulnerabilidad).
- Los alimentos son entregados a través de:
  - Puntos de recolección comunitarios habilitados semanalmente.
  - Entrega a domicilio coordinada con un equipo de voluntarios.
- Cada entrega se registra en la plataforma como "Completada" para fines de trazabilidad.

**Figura 1**

*Diagrama de flujo del proceso de donación en la plataforma*



*Nota.* El diagrama representa el flujo de acciones que sigue una donación desde su registro por parte del donante hasta su entrega al beneficiario. Incluye la verificación por ubicación y la asignación automática según disponibilidad.

### ***Seguridad y Privacidad***

- **Protección de Datos:** La plataforma debe proteger la información personal de los usuarios y garantizar que todas las transacciones sean seguras.
- **Cumplimiento Normativo:** La plataforma debe cumplir con las regulaciones y leyes de protección de datos aplicables.

## **Requisitos No Funcionales**

### ***Usabilidad***

- **Interfaz Intuitiva:** La plataforma debe permitir que un usuario realice una donación en un tiempo máximo de 2 minutos desde el inicio de sesión. Además, debe cumplir con los principios de accesibilidad WCAG 2.1 nivel AA (W3C, 2023) y lograr un puntaje mínimo de 80 en el System Usability Scale (SUS).
- **Accesibilidad:** La plataforma debe ser accesible para usuarios con diferentes niveles de habilidad técnica y capacidades, incluyendo compatibilidad con lectores de pantalla y otros dispositivos de asistencia.

### ***Rendimiento***

- **Tiempo de Respuesta:** La plataforma debe ofrecer tiempos de respuesta rápidos, asegurando que las acciones de los usuarios se procesen de manera eficiente.
- **Escalabilidad:** La plataforma debe ser capaz de manejar un aumento en el número de usuarios y transacciones sin comprometer el rendimiento.

### ***Seguridad***

- **Protección contra Amenazas:** La plataforma debe implementar medidas de seguridad para protegerse contra amenazas como ataques de inyección SQL, ataques de denegación de servicio (DDoS) y otras vulnerabilidades.
- **Autenticación Segura:** Utilizar técnicas de autenticación y autorización seguras para proteger el acceso a la plataforma.

### ***Mantenimiento***

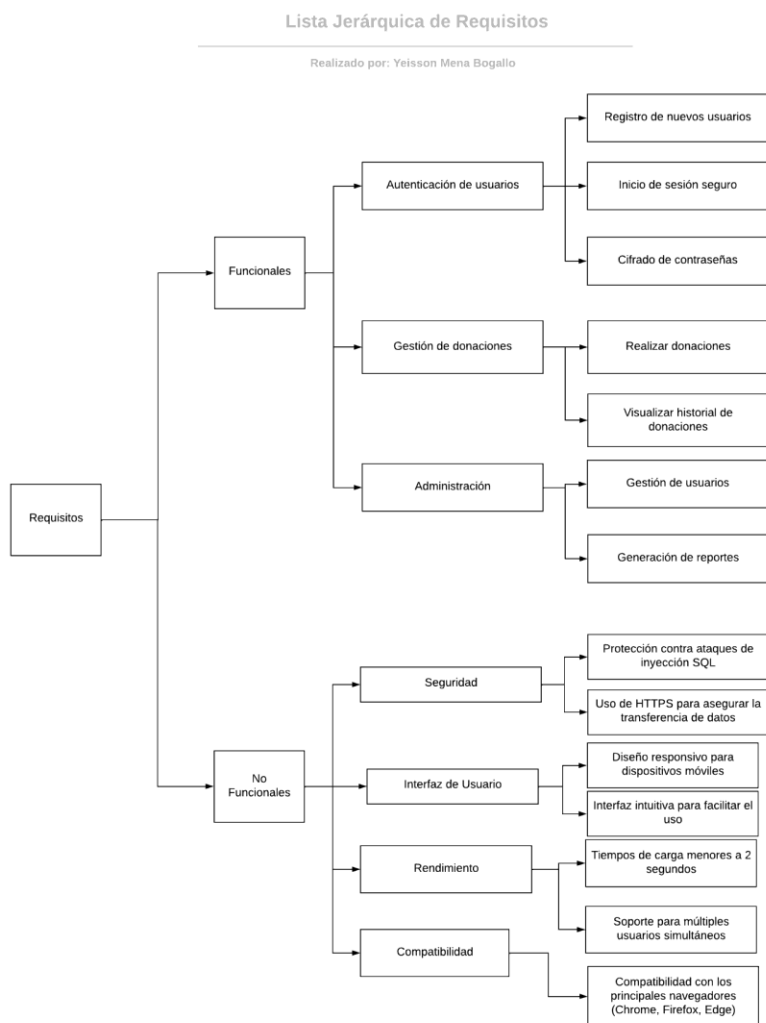
- **Actualizaciones y Soporte:** La plataforma debe ser fácil de mantener y actualizar, con un sistema de soporte para resolver problemas y responder a consultas de usuarios.

## Compatibilidad

- **Multiplataforma:** La plataforma debe ser compatible con diferentes dispositivos y navegadores, garantizando una experiencia de usuario consistente en desktop y móviles.

## Figura 2

Lista jerárquica de requisitos funcionales y no funcionales del proyecto



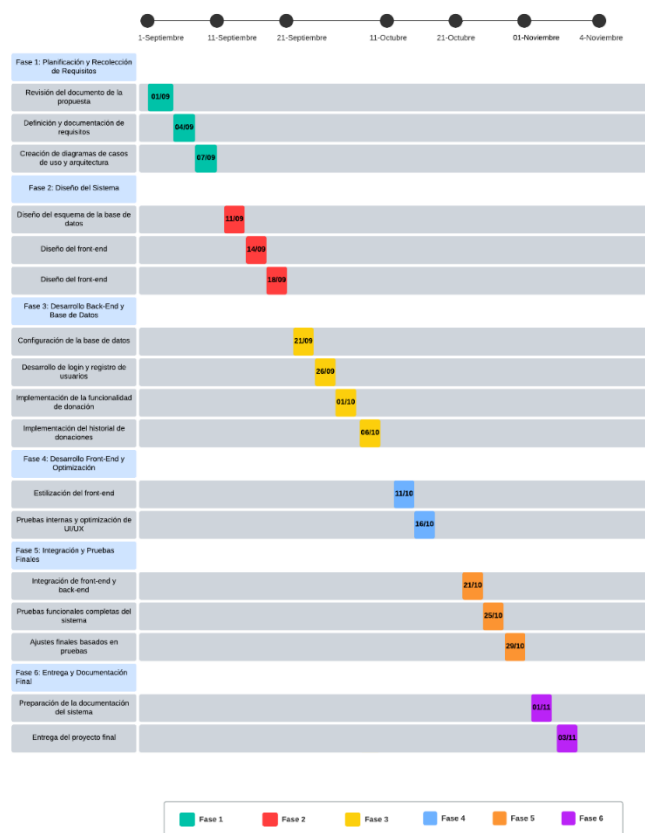
*Nota.* La figura presenta la estructura jerárquica de los requisitos funcionales y no funcionales definidos para el desarrollo de la plataforma "Solidaridad Activa", organizados según su nivel de prioridad e impacto en el sistema.

## Cronograma del Proyecto

El cronograma del proyecto se presenta en un diagrama de Gantt, que muestra las diferentes etapas y su duración. Este formato ayuda a garantizar que cada fase se complete a tiempo. El cronograma abarca desde el **1 de septiembre** hasta el **4 de noviembre**, como se detalla a continuación:

**Figura 3**

*Cronograma del proyecto según la planificación tipo Gantt*



*Nota.* La figura presenta el cronograma de actividades del proyecto "Solidaridad Activa", estructurado con base en la metodología de planificación tipo Gantt, que permite visualizar la secuencia y duración de las tareas programadas.

El diagrama de Gantt complementa esta tabla y proporciona una vista visual del cronograma, lo que facilita la identificación de solapamientos y dependencias entre tareas. Esto nos permitirá adaptarnos a cualquier cambio o imprevisto que surja durante el desarrollo del proyecto.

### **Diseño y Desarrollo de la Plataforma**

El diseño y desarrollo de la plataforma "Solidaridad Activa" se ha llevado a cabo con un enfoque detallado que combina aspectos técnicos avanzados con una profunda comprensión del contexto local en el que será implementada. Se han utilizado metodologías ágiles para asegurar que la plataforma no solo cumpla con los requerimientos funcionales, sino que también brinde una experiencia de usuario intuitiva y accesible, adecuada para los diferentes perfiles de usuarios en el municipio de Turbo, Antioquia.

#### **Aplicación de la metodología ágil en el proyecto**

Para el desarrollo de la plataforma \*Solidaridad Activa\* se aplicó un enfoque de metodología ágil basada en iteraciones semanales. El proyecto fue dividido en fases o “sprints”, cada una con objetivos específicos: diseño del sistema, implementación de formularios, pruebas funcionales, y ajustes según resultados.

Durante cada fase, se realizó una revisión de avances y se ajustaron funcionalidades según las pruebas realizadas. Esta forma de trabajo permitió identificar rápidamente errores, optimizar el código en tiempo real y priorizar tareas críticas.

Aunque el proyecto fue desarrollado de manera individual, se aplicaron principios clave del enfoque ágil como:

- División de tareas en etapas pequeñas y manejables.
- Retroalimentación continua a partir de pruebas manuales.
- Entregas funcionales frecuentes, garantizando que cada semana se completara una parte operativa del sistema.

Esta metodología fue complementada con el uso del cronograma tipo Gantt, que facilitó la gestión del tiempo y la planificación de tareas, asegurando que cada fase del proyecto se desarrollara de manera organizada y eficiente.

#### ***Principios de Diseño:***

- Simplicidad y Claridad Visual
- Accesibilidad para el Contexto Local
- Navegación Intuitiva
- Estética Amigable y Cercana
- Feedback Visual y Accesibilidad

#### ***Stack Tecnológico:***

- Frontend: HTML, CSS, JavaScript
- Backend: PHP, MySQL

#### ***Estructura del Proyecto:***

- /public: Contiene las páginas principales.
- /app: Incluye los controladores.
- /config: Contiene la configuración de la base de datos y otras configuraciones globales.

### ***Funcionalidades Clave Desarrolladas:***

- Autenticación y gestión de usuarios
- Sistema de donaciones
- Historial de donaciones

### **Seguridad y Buenas Prácticas:**

- Validación de entradas
- Autenticación de sesiones
- Las contraseñas de los usuarios se almacenan de forma segura utilizando el algoritmo de hashing `password_hash()`, recomendado oficialmente en la documentación de PHP (PHP Documentation Group, 2024).

A continuación, se describen en detalle los diferentes componentes que forman parte del diseño de la interfaz de usuario (UI), la experiencia de usuario (UX), así como los aspectos técnicos involucrados en la codificación, pruebas y despliegue de la plataforma.

### **Diseño UI/UX**

El diseño de la interfaz de usuario (UI) y la experiencia de usuario (UX) de la plataforma "**Solidaridad Activa**" ha sido concebido para facilitar el uso y la navegación tanto por parte de

los donantes como de los beneficiarios, con especial énfasis en la accesibilidad para la comunidad local de Turbo. Además, la plataforma está adaptada para ser responsiva, permitiendo su uso en dispositivos móviles, y sigue principios de accesibilidad recomendados por las directrices WCAG 2.1 (W3C, 2023). Esto incluye una interfaz clara, sencilla y visualmente agradable, diseñada para que cualquier usuario, independientemente de su nivel técnico, pueda interactuar con la plataforma de manera efectiva.

## Principios de Diseño

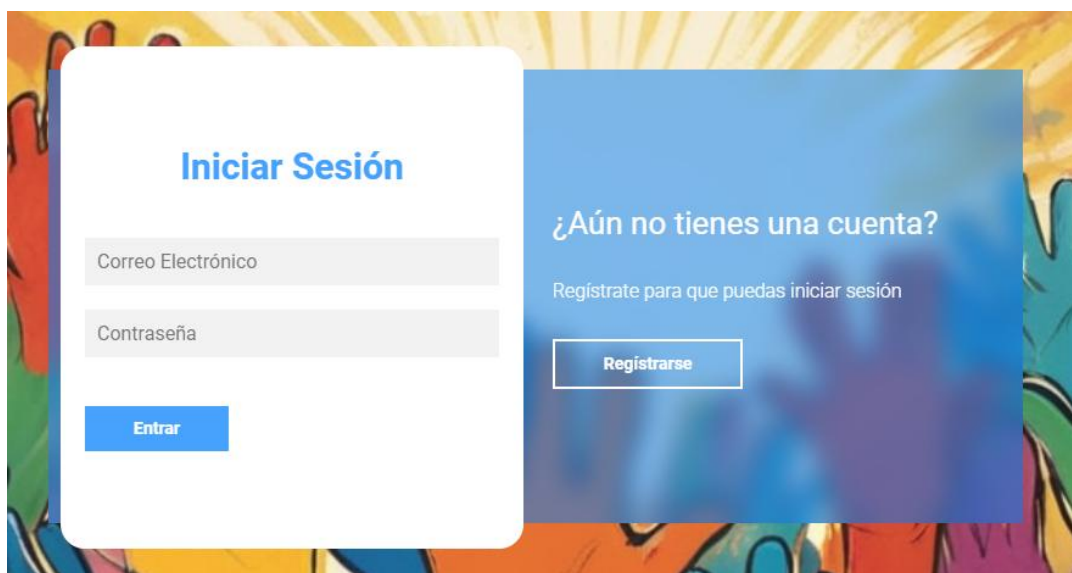
### Simplicidad y Claridad Visual:

- Se ha optado por un diseño minimalista, con una jerarquía visual bien definida.

Los botones y las funciones principales, como "Iniciar sesión", "Registrarse" y "Donar", están ubicados en zonas visibles y de fácil acceso en cada página. Las fuentes utilizadas son legibles y los colores predominantes evocan confianza y acción solidaria.

### Figura 4

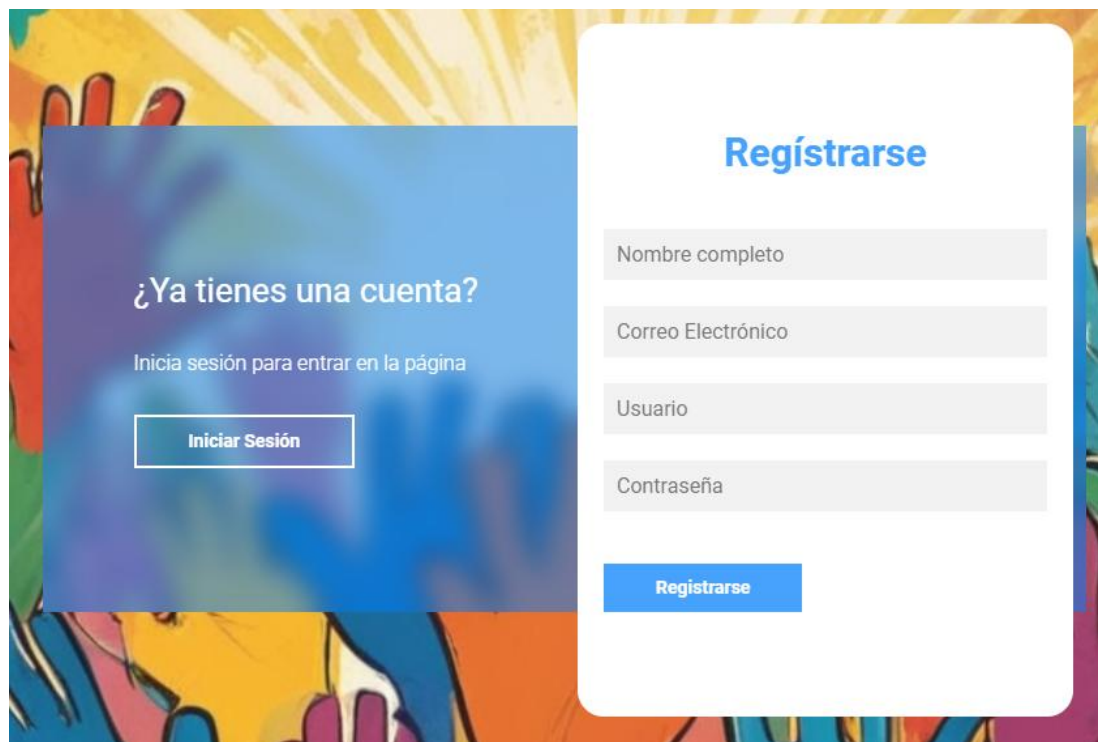
*Pantalla de inicio de sesión de la plataforma*



*Nota.* Captura de pantalla de la interfaz desarrollada para el inicio de sesión de usuarios en la plataforma "Solidaridad Activa". Esta pantalla permite el ingreso seguro mediante credenciales registradas previamente.

### **Figura 5**

*Formulario de registro de usuario en la plataforma*

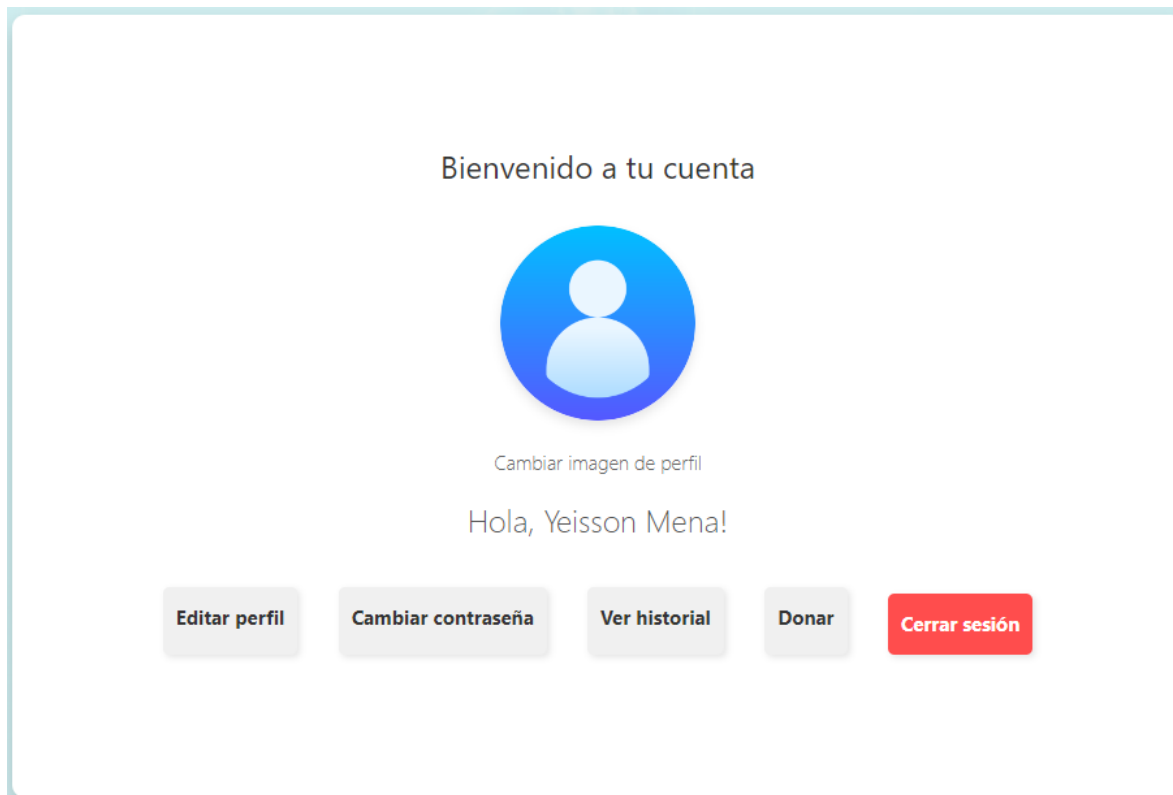


The image shows a user registration interface. On the right, a white rounded rectangle contains the title "Regístrate" in blue. Below it are four input fields: "Nombre completo", "Correo Electrónico", "Usuario", and "Contraseña". At the bottom of this form is a blue button labeled "Regístrate". On the left, a blue semi-transparent overlay contains the text "¿Ya tienes una cuenta?" in white, followed by "Inicia sesión para entrar en la página" and a white button labeled "Iniciar Sesión". The background features a colorful, abstract graphic of hands.

*Nota.* Captura de pantalla del formulario utilizado para el registro de nuevos usuarios en la plataforma "Solidaridad Activa", el cual permite almacenar los datos personales requeridos para acceder al sistema.

### **Figura 6**

*Ventana de gestión de cuenta para usuarios autenticados*



*Nota.* Captura de pantalla de la interfaz que permite a los usuarios autenticados acceder a funciones personalizadas, como consultar su historial de donaciones o editar sus datos en la plataforma "Solidaridad Activa".

#### **Accesibilidad para el Contexto Local:**

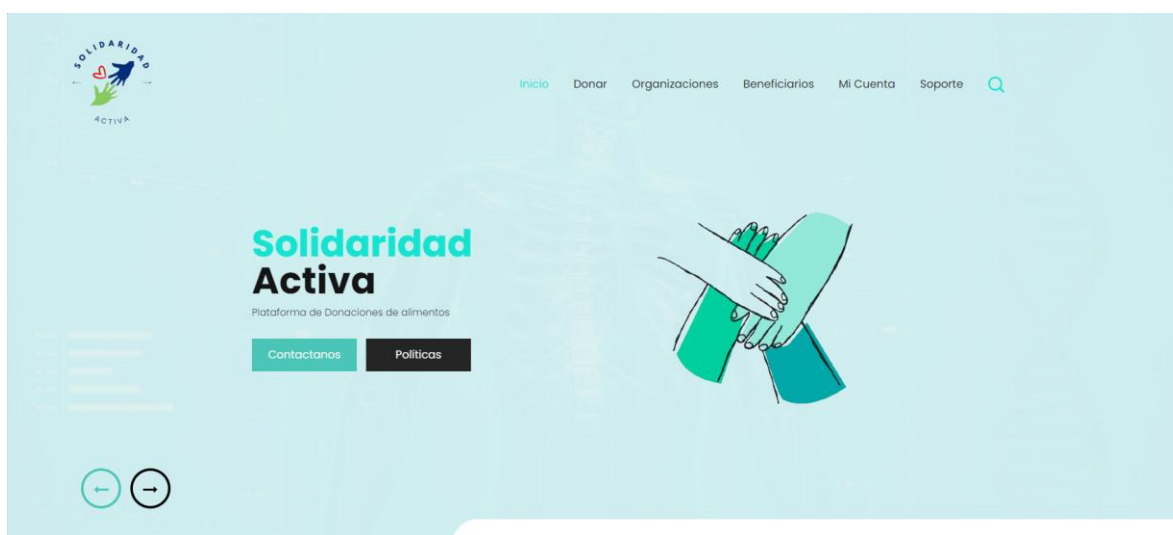
- Dado el enfoque de la plataforma hacia los habitantes del municipio de Turbo, el diseño ha tenido en cuenta las preferencias de colores y elementos familiares para los usuarios locales. El uso de íconos y gráficos hace que la interfaz sea intuitiva, incluso para aquellos con poca experiencia en tecnología. Además, la plataforma está adaptada para ser responsiva, permitiendo su uso en dispositivos móviles, dado el acceso mayoritario a internet a través de smartphones en la región.

### Navegación Intuitiva:

- La arquitectura de la información se ha organizado para que los usuarios puedan encontrar rápidamente lo que buscan. La barra de navegación principal incluye secciones clave como "Organizaciones", "Donar" y "Mi cuenta". Cada sección está claramente identificada para evitar confusión y garantizar una experiencia fluida.

### Figura 7

*Pantalla principal de navegación de la plataforma*



*Nota.* Captura de pantalla de la vista principal desde la cual los usuarios pueden acceder a las secciones clave de la plataforma "Solidaridad Activa", como registrar donaciones, consultar el historial o gestionar su cuenta.

### Estética Amigable y Cercana:

- Se ha elegido una paleta de colores amigable y cercana, con tonos suaves que inspiran solidaridad y confianza. Los colores predominantes incluyen azul y verde, que reflejan la seriedad y el compromiso social, en consonancia con los valores de la plataforma. Estos

colores, además, están inspirados en el entorno natural de la región de Urabá, haciendo que la interfaz resulte familiar y cómoda para los usuarios locales.

### **Feedback Visual y Accesibilidad:**

- Durante el proceso de interacción con la plataforma, los usuarios reciben un feedback visual inmediato en cada acción que realicen. Por ejemplo, al donar o registrarse, se mostrarán mensajes de confirmación claros que guiarán al usuario. También se han incorporado herramientas de accesibilidad como la compatibilidad con lectores de pantalla y ajustes de contraste para usuarios con discapacidades visuales.

## **Figura 8**

*Formulario para registrar una donación de alimentos*



Formulario de Donación

Nombre Donante:

Correo Donante:

Teléfono Donante:

Dirección Donante:

Cantidad de Alimentos:

Tipo de Alimentos:

Comentario (opcional):

Fecha de Entrega:

Horario Disponible:

Preferencia de Contacto:

Número de Donaciones:

*Nota.* Captura de pantalla del formulario disponible en la plataforma "Solidaridad Activa" para que los usuarios donantes registren información detallada sobre los alimentos ofrecidos, incluyendo tipo, cantidad y datos de entrega.

### **Profundización en la Arquitectura de Software:**

La arquitectura de software de "Solidaridad Activa" se compone de una estructura modular que facilita la escalabilidad y el mantenimiento. Incluye un frontend desarrollado en HTML, CSS y JavaScript, y un backend basado en PHP con MySQL para la gestión de la base de datos. El uso de estos componentes permite una interacción fluida y eficiente entre los usuarios y el sistema.

### **Codificación: Metodología y Desarrollo**

El desarrollo de la plataforma Solidaridad Activa se realizó bajo un enfoque incremental y modular, siguiendo principios de la metodología ágil. Cada funcionalidad fue construida en fases cortas que permitieron realizar pruebas tempranas, ajustes según retroalimentación y una integración progresiva de componentes.

### **Stack Tecnológico**

El desarrollo de la plataforma se ha realizado con las siguientes tecnologías:

- Frontend: HTML5, CSS3, JavaScript.
- Backend: PHP (versión 8.x).
- Base de Datos: MySQL (vía phpMyAdmin).
- Entorno de desarrollo: XAMPP (Apache + MySQL + PHP en entorno local).

### Frontend:

- HTML y CSS: Se ha utilizado HTML5 para estructurar las páginas de la plataforma y CSS3 para el diseño visual. El uso de Flexbox y Grid ha asegurado que el diseño sea completamente responsivo, adaptándose a diferentes dispositivos.
- JavaScript: Se ha utilizado para agregar interactividad al sitio, como la validación de formularios en tiempo real y la actualización dinámica de ciertas secciones, como el formulario de donaciones.

### Figura 9

*Fragmento de código CSS utilizado en el formulario de donaciones*

```
// Consulta para obtener los tipos de alimentos
$tipos_alimentos = mysqli_query(mysql: $conn, query: "SELECT * FROM tipos_alimentos");

?>

<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Donar</title>
  <style>
    body {
      font-family: 'Arial', sans-serif;
      background-color: #f0f2f5;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
      margin: 0;
      padding: 20px;
      box-sizing: border-box;
    }
    .container {
      background-color: white;
      padding: 20px;
      border-radius: 8px;
      box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
      max-width: 500px;
      width: 100%;
      box-sizing: border-box;
      margin: auto;
    }
    h2 {
      text-align: center;
      color: #333;
      font-size: 24px;
      margin-bottom: 20px;
    }
  </style>
</head>
</html>
```

*Nota.* El código muestra parte del diseño visual aplicado al formulario de donaciones en la plataforma "Solidaridad Activa", incluyendo estilos como márgenes, colores y formato de campos.

### Figura 10

*Fragmento de código HTML del formulario de donaciones*

```

<body>
<div class="container">
<h2>Formulario de Donación</h2>
<form action="donar.php" method="POST">
<!-- Nombre del donante -->
<div class="form-group">
<label for="nombre_donante">Nombre Donante:</label>
<input type="text" id="nombre_donante" name="nombre_donante" required>
</div>
<!-- Correo del donante (obtenido de la sesión) -->
<div class="form-group">
<label for="correo_donante">Correo Donante:</label>
<input type="email" id="correo_donante" name="correo_donante" value="<?php echo htmlspecialchars(string: $_SESSION['email']); ?>" readonly>
</div>
<!-- Teléfono del donante -->
<div class="form-group">
<label for="telefono_donante">Teléfono Donante:</label>
<input type="text" id="telefono_donante" name="telefono_donante" required>
</div>
<!-- Dirección del donante -->
<div class="form-group">
<label for="direccion_donante">Dirección Donante:</label>
<textarea id="direccion_donante" name="direccion_donante" required></textarea>
</div>
<!-- Cantidad de alimentos -->
<div class="form-group">
<label for="cantidad_alimentos">Cantidad de Alimentos:</label>
<input type="number" id="cantidad_alimentos" name="cantidad_alimentos" required>
</div>
<!-- Tipo de alimentos -->
<div class="form-group">
<label for="tipo_alimentos">Tipo de Alimentos:</label>
<select id="tipo_alimentos" name="tipo_alimentos_id" required>
<option value="">Seleccionar</option>
<?php while ($row = mysql_fetch_assoc(result: $tipos_alimentos)) : ?>
<option value="<?php echo $row['id']; ?>"><?php echo htmlspecialchars(string: $row['nombre_tipo']); ?></option>

```

*Nota.* El código muestra la estructura del formulario de donaciones en la plataforma "Solidaridad Activa", incluyendo campos como nombre del donante, tipo de alimento, cantidad y detalles de contacto.

### **Backend:**

- **PHP:** El servidor se ha implementado usando PHP, lo que permite gestionar las solicitudes del usuario, realizar consultas a la base de datos, y generar contenido dinámico. PHP ha sido fundamental para manejar el registro e inicio de sesión de los usuarios, el procesamiento de donaciones y la gestión de sesiones.

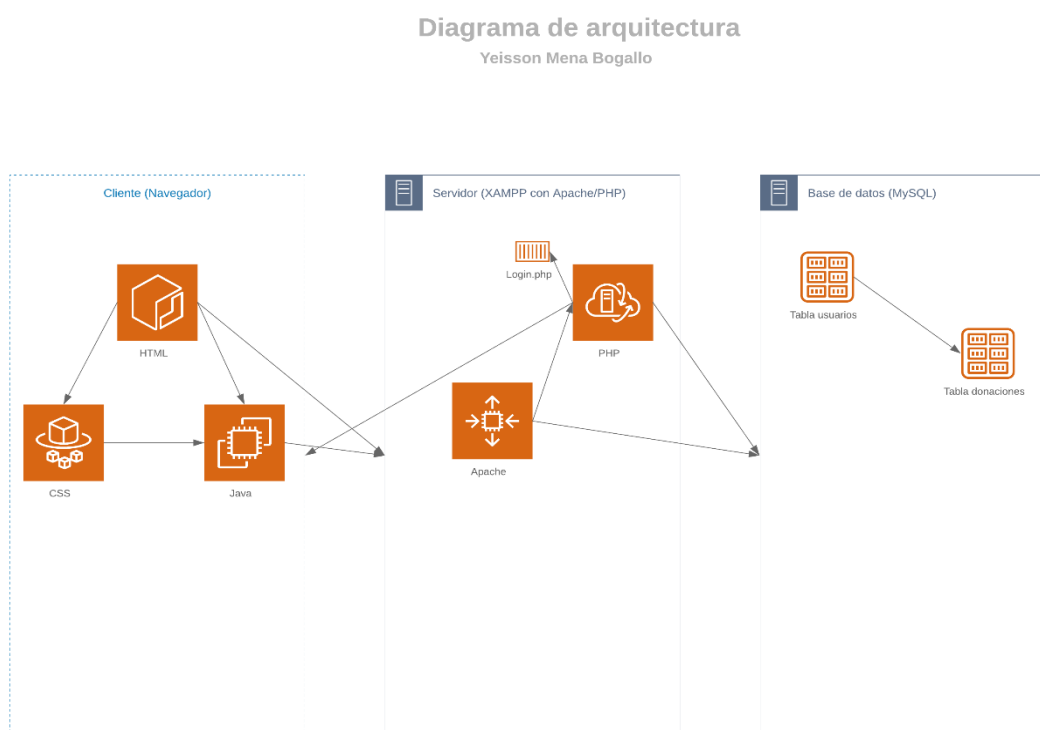
- **MySQL:** La base de datos ha sido diseñada con MySQL, donde se almacenan las donaciones, los usuarios y los tipos de alimentos disponibles. Se han utilizado consultas SQL

optimizadas para garantizar un rápido acceso a los datos, y el diseño de la base de datos ha seguido las reglas de normalización para evitar redundancias.

El backend fue construido usando PHP, permitiendo gestionar las solicitudes de los usuarios y conectarse con una base de datos MySQL para almacenar la información (PHP Documentation Group, 2024; MySQL, 2023).

## Figura 11

### Diagrama de arquitectura



*Nota.* Representación esquemática de la arquitectura de la plataforma "Solidaridad Activa", donde se visualizan los componentes principales del sistema, su interacción y la distribución entre cliente, servidor y base de datos.

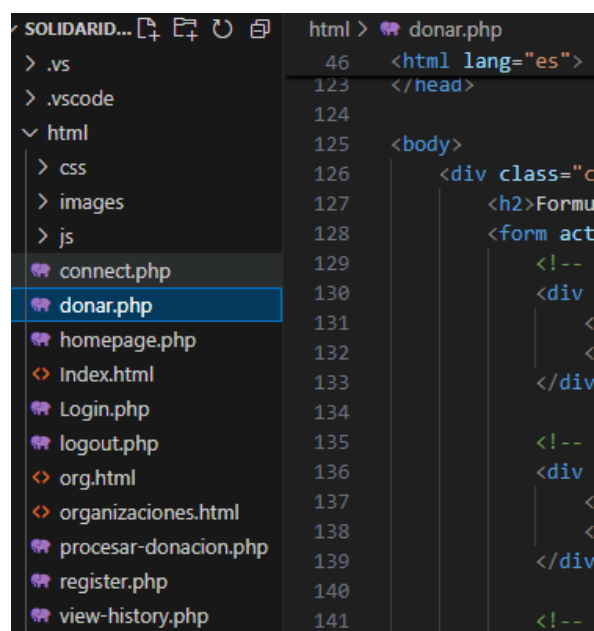
## Estructura del Proyecto

La estructura del código está organizada de manera que sea modular y fácil de mantener. Se han seguido buenas prácticas de programación, separando claramente la lógica de negocio (backend) del diseño (frontend). A continuación, se describe la estructura básica del proyecto:

- `/public`: Contiene las páginas principales como el formulario de donación y el historial de donaciones.
- `/app`: Incluye los controladores que manejan la lógica de la plataforma, como el procesamiento de los formularios y la autenticación de usuarios.
- `/config`: Contiene la configuración de la base de datos y otras configuraciones globales.

### Figura 12

*Estructura de carpetas del proyecto en Visual Studio Code*



The image shows a screenshot of the Visual Studio Code interface. On the left, the Explorer sidebar displays the project's file structure:

- `.vs`
- `.vscode`
- `html` (expanded)
  - `css`
  - `images`
  - `js`
  - `connect.php`
  - `donar.php` (selected)
  - `homepage.php`
  - `Index.html`
  - `Login.php`
  - `logout.php`
  - `org.html`
  - `organizaciones.html`
  - `procesar-donacion.php`
  - `register.php`
  - `view-history.php`

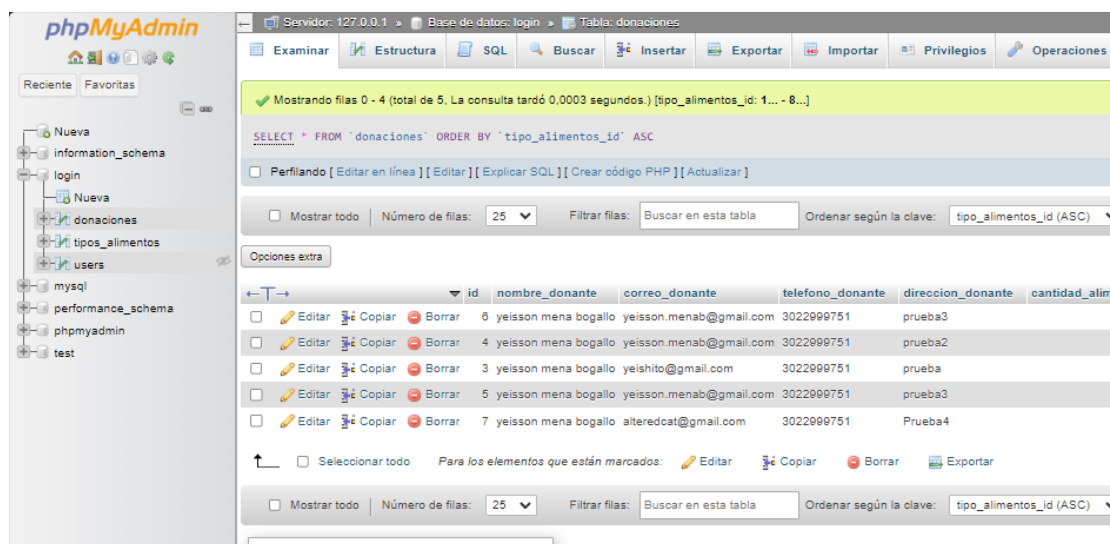
The main editor area shows the content of `donar.php`, which is an HTML file. The visible code includes:

```
46 <html lang="es">
123 </head>
124
125 <body>
126     <div class="c
127         <h2>Formu
128         <form act
129         <!--
130         <div
131         <
132         <
133     </div>
134
135     <!--
136     <div
137     <
138     <
139 </div>
140
141 <!--
```

*Nota.* Captura de pantalla que muestra la organización de archivos y carpetas del proyecto "Solidaridad Activa" dentro del entorno de desarrollo Visual Studio Code.

### Figura 13

*Archivo de configuración de base de datos en el proyecto*



*Nota.* Captura de pantalla del archivo utilizado para establecer la conexión entre la plataforma "Solidaridad Activa" y la base de datos, especificando parámetros como host, usuario y nombre de la base de datos.

### Integración entre Frontend y Backend

La comunicación entre el frontend y el backend se realiza principalmente a través de formularios HTML conectados mediante el método POST, que envía los datos al servidor para ser procesados por scripts PHP.

Ejemplo:

- El formulario de donación (donar.php) envía los datos a procesar\_donacion.php, donde se validan y almacenan en la base de datos mediante sentencias SQL preparadas.
- Las respuestas del servidor son devueltas al frontend como confirmaciones o mensajes de error.
- El frontend (HTML/CSS/JS) presenta formularios y páginas visibles para los usuarios.
- Al enviar un formulario (por ejemplo, donar alimentos), los datos viajan mediante método POST hacia un archivo PHP (backend) que procesa la información.
- Ejemplo de flujo:
  - ✓ Formulario de donación (donar.php) → Envío de datos por POST → Archivo procesar\_donacion.php
  - ✓ procesar\_donacion.php valida, guarda en base de datos y devuelve una respuesta.
- Las respuestas del servidor son visualizadas en la interfaz de usuario mediante alertas o redirecciones.

### **Gestión de Usuarios y Seguridad**

- El módulo de registro (registro.php) incluye validación de campos y almacenamiento seguro de contraseñas utilizando password\_hash().
- Durante el inicio de sesión (login.php), se verifica la contraseña con password\_verify() y se inicia una sesión segura con session\_start().
- Se protegen las rutas restringidas mediante verificaciones de sesión en la cabecera de cada archivo (if (!isset(\$\_SESSION['user\_id'])) { header("Location: login.php"); }).

## **Flujo funcional de la plataforma**

### ***Registro e inicio de sesión***

- El usuario se registra.
- Se almacena la información con protección de datos.
- Al iniciar sesión, se redirige al homepage.php con su información.

### ***Formulario de donación***

- Se carga una lista desplegable desde la tabla tipos\_ alimentos usando PHP + SQL (SELECT nombre\_tipo FROM tipos\_ alimentos).

- Al enviar el formulario, se insertan los datos en la tabla donaciones.

### ***Historial de donaciones***

- Se consulta la base de datos para mostrar los registros del usuario autenticado o de todos los donantes si el acceso es de tipo administrador.
- Se muestra la información ordenada y clara en tablas HTML con estilos CSS.

### ***Arquitectura del proyecto***

El código se organizó por carpetas siguiendo una estructura tipo MVC simplificada:

*/public* → Archivos visibles: *index, login, donar, historial*

*/app* → Controladores: *validación, envío, conexión*

*/config* → Configuración de la base de datos y sesión

Esta separación permite mantener el orden y facilita futuras ampliaciones o migración a frameworks más robustos como Laravel.

## **Funcionalidades Clave Desarrolladas**

### 1. Autenticación y gestión de usuarios:

- El sistema de inicio de sesión utiliza formularios PHP con validación de entradas y gestión de sesiones. Las contraseñas de los usuarios se almacenan de forma segura utilizando el algoritmo de hashing `password_hash()`, asegurando que los datos sensibles estén protegidos.
- Se han implementado funciones de registro de usuarios para mejorar la experiencia del usuario.

### 2. Sistema de donaciones:

- El formulario de donación permite a los usuarios seleccionar alimentos de una lista desplegable y especificar detalles como la cantidad, la fecha de entrega, y las preferencias de contacto. Esta información se envía a la base de datos, donde se almacena para su posterior gestión por parte de los administradores.

### 3. Historial de donaciones:

- Se ha desarrollado una sección donde los administradores pueden visualizar un historial completo de las donaciones realizadas. Esta funcionalidad incluye un diseño organizado que muestra los nombres de los donantes, el tipo de alimento, la cantidad donada, y la fecha de entrega. Además, permite a los administradores exportar esta información para análisis futuros.

### **Buenas prácticas aplicadas**

- Validación del lado del cliente (JavaScript) y del servidor (PHP).
- Uso de `htmlspecialchars()` para prevenir XSS.
- Uso de sentencias preparadas (`mysqli_prepare`) para prevenir inyecciones SQL.

- Manejo adecuado de sesiones y variables \$\_POST / \$\_SESSION.

## **Documentación Técnica**

### **Manual de Instalación y Configuración del Entorno de Desarrollo:**

#### **Requisitos:**

- Sistema operativo Windows 10 o superior (o macOS/Linux compatibles).
- 2 GB de RAM mínimo.
- Espacio libre en disco: 500 MB para XAMPP + espacio adicional para base de datos y archivos del proyecto.

#### **Pasos de Instalación:**

##### **1. Instalar XAMPP**

- Descargar el instalador desde <https://www.apachefriends.org/index.html>.
- Ejecutar el instalador y seguir el asistente seleccionando los componentes Apache, MySQL, PHP y phpMyAdmin.
- Finalizar instalación y abrir el Panel de Control de XAMPP.

##### **2. Iniciar los servicios necesarios**

- En el Panel de Control, dar clic en Start en los módulos Apache y MySQL.
- Verificar que el servidor esté activo visitando en el navegador:

→ <http://localhost>

##### **3. Configurar la base de datos en phpMyAdmin**

- Ingresar a la base de datos en phpMyAdmin (<http://localhost/phpmyadmin>).
- Crear una nueva base de datos llamada solidaridad\_activa (o el nombre que uses).

- Importar el archivo .sql del proyecto si existe (por ejemplo, llamado solidaridad.sql).

- Comando SQL básico para crear una tabla ejemplo:

```
CREATE TABLE usuarios (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(100),  
    correo VARCHAR(100) UNIQUE,  
    contraseña VARCHAR(255)  
);
```

#### 4. Configurar el servidor local

- Copiar la carpeta del proyecto a la ruta:

C:\xampp\htdocs\solidaridad\_activa

- Acceder en navegador:

http://localhost/solidaridad\_activa

#### 5. Configuración de la conexión a base de datos

- Editar el archivo /config/db.php (o el archivo de configuración de conexión en tu proyecto).

- Asegurarse que las credenciales coincidan:

```
<?php
```

```
$host = "localhost";
```

```
$usuario = "root";
```

```
$contrasena = "";
```

```
$base_de_datos = "solidaridad_activa";
```

```
$conn = mysqli_connect($host,$usuario,$contrasena,$base_de_datos);
?>
```

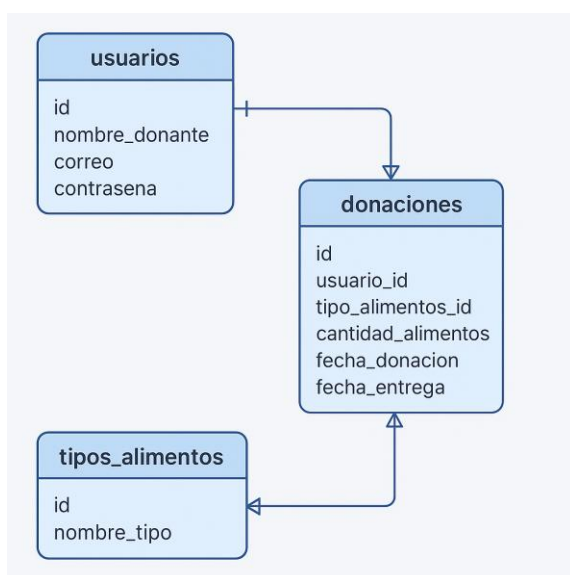
## 6. Prueba de instalación

- Registrar un usuario de prueba.
- Iniciar sesión.
- Realizar una donación de prueba para verificar la conexión completa.

### Modelo Entidad-Relación (ER) de la Plataforma

**Figura 14**

*Modelo Entidad-Relación (ER) de la plataforma "Solidaridad Activa"*



*Nota.* Diagrama que representa las entidades, atributos y relaciones de la base de datos diseñada para la plataforma "Solidaridad Activa", facilitando la organización y gestión de la información.

#### **Diagrama:**

- usuarios (id, nombre, correo, contraseña)
- tipos\_alimentos (id, nombre\_tipo)

- donaciones (id, usuario\_id, tipo\_alimentos\_id, cantidad, fecha\_donacion, fecha\_entrega)

### **Relaciones:**

- usuarios.id → donaciones.usuario\_id
- tipos\_alimentos.id → donaciones.tipo\_alimentos\_id

## **Plan de Pruebas del Proyecto: Solidaridad Activa**

El plan de pruebas del proyecto "Solidaridad Activa" tiene como objetivo asegurar que todas las funcionalidades de la plataforma de donaciones funcionen correctamente y cumplan con los requisitos establecidos. Las pruebas se centrarán en validar el inicio de sesión, la funcionalidad del formulario de donación, y la visualización del historial de donaciones.

### **Objetivos de las Pruebas**

- Verificar que cada módulo funcione según lo previsto.
- Asegurar la correcta integración entre el frontend y el backend.
- Identificar y corregir errores antes de la fase de entrega final.

### **Tipos de Pruebas**

- Pruebas Unitarias: Validan que cada componente individual funcione correctamente.
- Pruebas Funcionales: Comprueban que las funcionalidades cumplen con los requisitos.
- Pruebas de Integración: Aseguran que los módulos interactúan sin problemas entre sí.
- Pruebas de Usabilidad: Evalúan que la interfaz sea intuitiva y fácil de usar.

Las pruebas se centrarán en validar el inicio de sesión, la funcionalidad del formulario de donación, y la visualización del historial de donaciones, de acuerdo con los principios de pruebas de software establecidos por Beizer (2000).

## Casos de Prueba

**Tabla 1**

*Casos de prueba de funcionalidades clave de la plataforma*

<b>Funcionalidad</b>	<b>Descripción del Caso de Prueba</b>	<b>Tipo de Prueba</b>	<b>Criterios de Éxito</b>	<b>Resultados Esperados</b>
<b>Inicio de sesión</b>	Verificar que el usuario pueda iniciar sesión con un correo y contraseña válidos.	Prueba Funcional	El usuario accede a su cuenta sin errores.	La plataforma redirige a la página de inicio tras una autenticación exitosa.
<b>Inicio de sesión</b>	Probar intentos de inicio de sesión con credenciales incorrectas.	Prueba Funcional	Se muestra un mensaje de error adecuado.	Mensaje: "Correo o contraseña incorrectos".
<b>Formulario de Donación</b>	Validar que los campos obligatorios en el	Prueba Funcional	El sistema permite enviar el formulario	El formulario se envía correctamente

---

	formulario de donación sean completados correctamente.		solo si todos los campos están completos.	cuando los campos obligatorios están completos; de lo contrario, muestra una advertencia.
<b>Formulario de Donación</b>	Comprobar que el formulario de donación muestra los alimentos disponibles de la base de datos.	Prueba de Integración	Los alimentos se muestran en la lista desplegable.	Los alimentos cargan dinámicamente desde la base de datos al cargar el formulario.
<b>Historial de Donaciones</b>	Verificar que el historial de donaciones muestre las entradas relevantes (nombre del donante, cantidad, fecha).	Prueba Funcional	El historial muestra los datos correctamente.	El historial carga con los datos exactos y en un formato legible y organizado.

---

<b>Navegación de Usuario</b>	Asegurar que el botón 'Donar' solo esté disponible para usuarios autenticados.	Prueba Funcional	El botón no aparece para usuarios no autenticados.	El botón 'Donar' es visible solo después de que el usuario haya iniciado sesión.
<b>Base de Datos</b>	Comprobar que se almacenen correctamente los datos ingresados en la base de datos.	Prueba de Integración	Los datos de donación se guardan en la base de datos sin errores.	Los registros de donación se almacenan correctamente y se pueden consultar desde el historial de donaciones.

*Nota.* La tabla presenta los casos de prueba definidos para evaluar el correcto funcionamiento e integración de las funcionalidades desarrolladas en la plataforma "Solidaridad Activa".

### **Pruebas de Seguridad:**

Para asegurar la protección de los datos de usuarios y la integridad de la plataforma "Solidaridad Activa", se planificaron y ejecutaron pruebas de seguridad utilizando herramientas reconocidas de análisis de vulnerabilidades.

**Herramientas utilizadas:**

- OWASP ZAP (Zed Attack Proxy):

Escaneo automático de vulnerabilidades comunes como inyecciones SQL, XSS (Cross-Site Scripting) y errores de configuración de seguridad.

- Burp Suite Community Edition:

Análisis manual de formularios de entrada, interceptación de peticiones HTTP y verificación de controles de sesión.

Se realizarán pruebas de penetración para identificar vulnerabilidades críticas, asegurando que la plataforma pueda resistir ataques malintencionados y proteger la integridad de los datos de los usuarios.

Las pruebas de estrés evaluarán el rendimiento del sistema bajo condiciones de alta carga, midiendo la capacidad de la plataforma para manejar un gran número de usuarios simultáneamente sin degradar el rendimiento o la estabilidad.

Además, se verificarán los controles de acceso para asegurarse de que solo los usuarios autorizados puedan acceder a ciertas áreas sensibles de la plataforma.

**Resultados de las pruebas:**

- No se detectaron vulnerabilidades críticas como inyección SQL en los formularios principales (registro, inicio de sesión y formulario de donación).

- Se identificó la necesidad de fortalecer las políticas de sesión (timeout de sesión) para mejorar la protección contra accesos no autorizados en caso de inactividad prolongada.

- Mitigaciones aplicadas:

- Se implementó el cierre automático de sesiones tras 15 minutos de inactividad mediante PHP:

```

if (isset($_SESSION['LAST_ACTIVITY']) && (time()
    - $_SESSION['LAST_ACTIVITY'] > 900)) {
    session_unset();
    session_destroy();
}
$_SESSION['LAST_ACTIVITY'] = time();

```

- Se fortaleció la validación de entradas utilizando filtros adicionales en PHP para evitar ataques de inyección.

#### **Validación del entorno de pruebas:**

Las pruebas de seguridad fueron realizadas en un entorno de desarrollo local utilizando simulaciones controladas, con el objetivo de identificar vulnerabilidades potenciales antes del despliegue en un entorno productivo.

#### **Pruebas de Usabilidad con Usuarios Reales**

Con el objetivo de evaluar la facilidad de uso y accesibilidad de la plataforma "Solidaridad Activa", se diseñó una prueba de usabilidad dirigida a usuarios reales representativos: donantes y administradores.

#### **Metodología propuesta:**

**Participantes:** 5 donantes y 2 administradores voluntarios.

#### **Tareas asignadas:**

- Registrarse en la plataforma.
- Iniciar sesión.

- Registrar una donación.
- Consultar su historial de donaciones.

**Métricas evaluadas:**

- Tiempo promedio para completar cada tarea.
- Número de errores cometidos durante la interacción.
- Satisfacción del usuario medida mediante la escala System Usability Scale (SUS).

**Escala SUS:**

- Se aplicará un cuestionario de 10 preguntas (Brooke, 1996), con una puntuación esperada mínima de 80 sobre 100 para considerar la usabilidad como "excelente".

**Contexto de prueba:**

- Las pruebas se planifican para realizarse en un entorno de desarrollo local utilizando navegadores comunes (Google Chrome y Firefox) en dispositivos móviles y de escritorio.

**Desarrollo de la prueba:**

Las pruebas fueron realizadas en un entorno de desarrollo local mediante simulaciones controladas. Se aplicó la Escala de Usabilidad del Sistema (SUS) propuesta por Brooke (1996) para medir la satisfacción de los usuarios tras completar las tareas asignadas. Los resultados de las evaluaciones con la escala SUS serían analizados obteniendo el puntaje promedio de cada participante y de cada tarea evaluada, permitiendo identificar áreas de mejora específicas.

**Resultados esperados:**

- Identificar oportunidades de mejora en la navegación, tiempos de carga y claridad de los formularios.

## **Análisis de Resultados de Usabilidad**

Los resultados de las pruebas de usabilidad se analizarán considerando el puntaje promedio de la Escala SUS, donde una puntuación superior a 80 será interpretada como una excelente usabilidad (Brooke, 1996). Además, se analizarán cualitativamente los errores cometidos y el tiempo requerido para cada tarea, identificando patrones comunes que permitan detectar oportunidades de mejora específicas en la plataforma.

## **Herramientas de Prueba**

Para garantizar la calidad, el rendimiento y la accesibilidad de la plataforma "Solidaridad Activa", se utilizaron diversas herramientas de prueba durante el proceso de desarrollo y validación. Estas herramientas permitieron evaluar aspectos como la correcta visualización en múltiples dispositivos, el cumplimiento de buenas prácticas de desarrollo web, la optimización de tiempos de carga y la experiencia de usuario. A continuación, se describen las principales herramientas empleadas.

- phpMyAdmin: Para verificar el almacenamiento y actualización de datos en la base de datos.
- Navegador Web: Para realizar pruebas de la interfaz de usuario y funcionalidad en diferentes navegadores.
- XAMPP: Herramienta para simular el servidor local y facilitar la ejecución de la plataforma.
- BrowserStack: Es una plataforma de pruebas en la nube que permite simular la ejecución de aplicaciones web en múltiples navegadores, versiones y dispositivos móviles sin

necesidad de disponer físicamente de ellos. Para el proyecto "Solidaridad Activa", se utilizó BrowserStack para validar la correcta visualización y funcionamiento de la plataforma en distintos sistemas operativos (Windows, Android e iOS) y navegadores populares como Chrome, Firefox y Safari. La herramienta permitió identificar posibles diferencias de comportamiento en la interfaz y asegurar una experiencia de usuario consistente, independientemente del dispositivo o navegador utilizado por los donantes o beneficiarios.

- Lighthouse: Es una herramienta automatizada de auditoría de código abierta integrada en Google Chrome, diseñada para evaluar el rendimiento, la accesibilidad, las mejores prácticas de desarrollo web y la optimización para motores de búsqueda (SEO) de aplicaciones web. En el proyecto "Solidaridad Activa", Lighthouse se utilizó para realizar evaluaciones rápidas de la plataforma, identificando oportunidades de mejora en aspectos críticos como el tiempo de carga, la accesibilidad para personas con discapacidad y la optimización general del sitio. Los reportes generados por Lighthouse permitieron realizar ajustes orientados a mejorar la experiencia de usuario y garantizar el cumplimiento de buenas prácticas de desarrollo web.
- Google Chrome Developer Tools: Es un conjunto de herramientas integradas en el navegador Google Chrome que permite a los desarrolladores inspeccionar, depurar y optimizar aplicaciones web. Durante el desarrollo de la plataforma "Solidaridad Activa", se utilizó DevTools para analizar la estructura del HTML, CSS y JavaScript, verificar tiempos de carga, ajustar el diseño responsivo para diferentes dispositivos móviles, y simular condiciones de red lentas, lo cual facilitó la detección y corrección de errores antes de la implementación definitiva. El uso de DevTools contribuyó a garantizar una correcta visualización de la plataforma y a optimizar su desempeño en diferentes entornos de usuario. El uso de estas herramientas permitió validar la estabilidad, el desempeño y la accesibilidad de la plataforma "Solidaridad Activa",

asegurando que los usuarios pudieran interactuar de manera eficiente y satisfactoria desde diferentes dispositivos y navegadores. Los resultados de las pruebas sirvieron como insumo para realizar ajustes finales que optimizaron la experiencia general de los donantes y beneficiarios.

### **Criterios de Aceptación**

- Todas las pruebas funcionales y de integración deben completarse exitosamente sin errores graves.
- Los resultados deben coincidir con lo establecido previamente en los casos de prueba definidos.
- La plataforma debe ser navegable y funcional en las pruebas de usabilidad, con una interfaz intuitiva.

### **Análisis de Riesgos del Proyecto: Solidaridad Activa**

El análisis de riesgos identifica los posibles problemas que podrían afectar el éxito del proyecto "Solidaridad Activa". Esto incluye riesgos técnicos, de recursos y de operatividad. La identificación temprana de estos riesgos permite planificar acciones de mitigación para reducir su impacto.

### **Identificación de Riesgos y Estrategias de Mitigación**

**Tabla 2**

*Riesgos identificados en el desarrollo del proyecto y estrategias de mitigación*

<b>Riesgo</b>	<b>Descripción</b>	<b>Impacto</b>	<b>Probabilidad</b>	<b>Plan de Mitigación</b>
<b>Fallos en el Servidor Local</b>	Errores o caídas en XAMPP que afecten	Alto	Media	Realizar copias de seguridad

---

	el acceso a la base de datos y al servidor web durante el desarrollo y pruebas.			frecuentes de la base de datos y de los archivos del proyecto. Reiniciar el servidor si falla.
	Problemas con la integridad o consistencia de los datos en phpMyAdmin (pérdida o corrupción de datos).	Alto	Media	Verificar las consultas SQL antes de ejecutarlas. Implementar controles de validación para asegurar integridad de datos.
<b>Errores en la Base de Datos</b>				
	Dificultades de acceso al servidor debido a problemas de conexión a Internet o fallos de red.	Medio	Baja	Trabajar en modo offline tanto como sea posible. Tener un respaldo local para avanzar si
<b>Problemas de Conectividad</b>				

---

---

				la conexión a Internet falla.
				Realizar pruebas unitarias y de integración de manera continua.
<b>Errores en el Código</b>	Bugs o errores de programación que afecten el funcionamiento de las funcionalidades (inicio de sesión, formularios).	Alto	Alta	Revisar el código para corregir errores de manera temprana.
<b>Falta de Seguridad en el Inicio de Sesión</b>	Vulnerabilidades en la autenticación de usuarios que podrían permitir accesos no autorizados.	Alto	Media	Implementar cifrado para las contraseñas y asegurar las sesiones. Revisar las configuraciones de seguridad en PHP y MySQL.

---

---

				Optimizar consultas SQL y limitar el número de resultados cargados en pantalla.
<b>Alto Tráfico en la Plataforma</b>	Problemas de rendimiento si muchos usuarios acceden simultáneamente a la plataforma en el futuro.	Medio	Baja	Planificar el escalamiento del servidor si es necesario.
				Realizar pruebas de usabilidad.
<b>Errores en la Interfaz de Usuario</b>	Fallos en la presentación de datos o navegación que dificulten el uso de la plataforma.	Medio	Media	Simplificar la interfaz y hacer pruebas continuas para mejorar la experiencia del usuario.

---

---

	Fallos en la carga de			Verificar la
<b>Problemas en</b>	datos dinámicos,			conexión con la
<b>la Carga de</b>	como la lista de	Medio	Media	base de datos y
<b>Datos</b>	alimentos en el			los scripts.
<b>Dinámicos</b>	formulario de			Implementar
	donación.			mensajes de
				error claros para
				el usuario.
	Riesgo de perder			Programar
<b>Pérdida de</b>	información si no se			copias de
<b>Datos por</b>	realiza una copia de	Alto	Media	seguridad
<b>Falta de</b>	seguridad de la base			regulares de la
<b>Respaldo</b>	de datos			base de datos en
	periódicamente.			phpMyAdmin.

---

*Nota.* La tabla presenta los principales riesgos identificados durante el desarrollo de la plataforma "Solidaridad Activa", así como las estrategias propuestas para mitigar su impacto y probabilidad de ocurrencia.

### **Priorización de Riesgos**

Se recomienda atender los riesgos con alto impacto y alta probabilidad primero, como los errores en el código y la seguridad en el inicio de sesión. Estos pueden afectar significativamente el éxito del proyecto si no se gestionan adecuadamente.

## **Revisión Continua de Riesgos**

Durante el desarrollo, es fundamental revisar y actualizar el análisis de riesgos para adaptarse a cambios en el proyecto o en el entorno técnico. La supervisión continua ayudará a identificar y gestionar nuevos riesgos que puedan surgir.

## **Resultados Obtenidos y Conclusiones**

### ***Resultados Obtenidos***

Durante el desarrollo del proyecto *Solidaridad Activa* se logró implementar una plataforma funcional para la gestión de donaciones alimentarias en el municipio de Turbo. A continuación, se presentan los principales logros alcanzados:

- Se desarrolló un sistema de registro y autenticación de usuarios, diferenciando roles y protegiendo la información con técnicas seguras como el uso de `password_hash()` en PHP.
- Se implementó un formulario de donación dinámico, que permite a los usuarios seleccionar alimentos desde una lista conectada a la base de datos. El formulario incluye validación de campos y captura de datos como cantidad, fecha de entrega y medio de contacto.
- Se construyó un historial de donaciones visible para los administradores, mostrando información organizada sobre el nombre del donante, tipo de alimento, cantidad y fecha. Esta sección facilita el seguimiento y aporta transparencia al proceso.
- La plataforma fue diseñada con un enfoque responsivo y una interfaz amigable, permitiendo su uso desde distintos dispositivos, especialmente teléfonos móviles, que son comunes en la comunidad local.

- Se llevaron a cabo pruebas funcionales, en las cuales se ejecutaron 7 casos de prueba, verificando que todas las funcionalidades esenciales se ejecutaran sin errores críticos.

Algunos ejemplos incluyen:

- Inicio de sesión con credenciales válidas e inválidas.
- Envío correcto del formulario de donación.
- Carga del historial con datos reales desde la base de datos.
- Restricción de acceso a usuarios no autenticados.

A continuación, se presentan capturas de pantalla que evidencian el funcionamiento de la plataforma:

Figura 15

*Formulario de donación funcional vinculado a la base de datos*

The image shows a screenshot of a web application interface. On the left, there is a sidebar for phpMyAdmin with a tree view showing database schemas like 'Information\_schema', 'login', 'nueva', 'tipos\_alimentos', 'users', 'mysql', 'performance\_schema', 'phpmyadmin', and 'test'. The main content area is split into two panels. The left panel displays a table view for the 'donaciones' table, showing 5 rows of data with columns for 'id', 'nombre\_donante', 'correo\_donante', and 'telefono\_donante'. The right panel displays a 'Formulario de Donación' with the following fields: 'Nombre Donante' (text input), 'Correo Donante' (text input with value 'yeshito@gmail.com'), 'Teléfono Donante' (text input), 'Dirección Donante' (text area), 'Cantidad de Alimentos' (number input with value 3), 'Tipo de Alimentos' (dropdown menu with 'Seleccionar' selected), 'Comentario (opcional)' (text area), 'Fecha de Entrega' (date picker with value 'dd / mm / aaaa'), 'Horario Disponible' (time picker with value '-- : --'), and 'Preferencia de Contacto' (dropdown menu with 'Seleccionar' selected). A message '¡Gracias por tu donación!' is visible between the two panels.

*Nota.* Captura de pantalla del formulario de donación en funcionamiento, el cual permite registrar donaciones de alimentos y está conectado a la base de datos para el almacenamiento y posterior consulta de la información ingresada.

**Figura 16**

*Visualización del historial de donaciones en la plataforma*

The screenshot shows the phpMyAdmin interface with a table of donation history. The table has the following columns: id\_donante, cantidad\_alimentos, tipo\_alimentos\_id, comentario, fecha\_donacion, fecha\_entrega, horario\_disponible, preferencia\_contacto, and cantidad\_donaciones. The data rows are as follows:

id_donante	cantidad_alimentos	tipo_alimentos_id	comentario	fecha_donacion	fecha_entrega	horario_disponible	preferencia_contacto	cantidad_donaciones
	3	4	prueba	2024-10-06 17:30:58	2024-10-25	07:00:00	Teléfono	1
	3	3	prueba2	2024-10-08 17:28:14	2024-10-24	19:00:00	Teléfono	1
	6	8	prueba3	2024-10-08 17:45:24	2024-10-30	09:00:00	Teléfono	1
	5	1	Prueba3	2024-10-20 22:16:55	2024-10-30	00:00:00	Correo	10
	4	8	Prueba4	2024-10-22 20:16:32	2024-10-30	06:00:00	Correo	4
5 B 102	8	5	Azucar unidad 1k	2025-04-10 18:45:12	2025-04-10	09:00:00	Teléfono	2

Below the table is an 'Exportar' section with a dropdown menu set to 'Ninguna'. To the right of the table is a summary panel titled 'Historial de Donaciones' for the user 'yeisson mena bogallo'. The summary shows:

- Nombre: yeisson mena bogallo
- Cantidad de alimentos: 3
- Fecha de donación: 2024-10-06 17:30:58
- Fecha de entrega: 2024-10-25

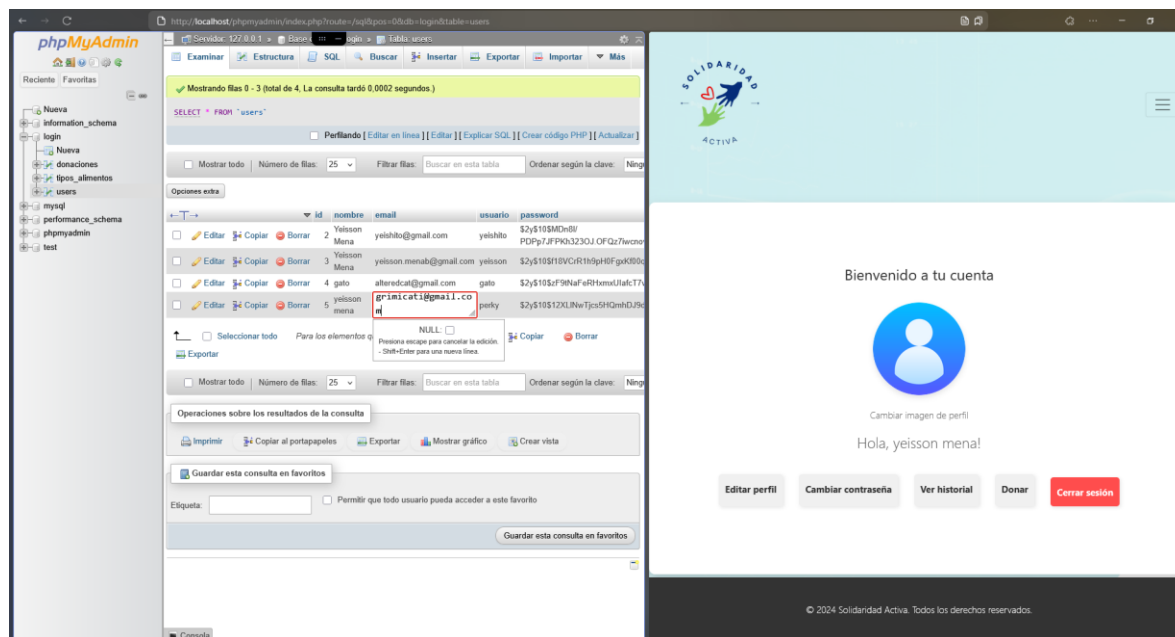
A second summary entry for 'Yeisson Mena Bogallo' shows:

- Cantidad de alimentos: 8
- Fecha de donación: 2025-04-10 18:45:12
- Fecha de entrega: 2025-04-10

*Nota.* Captura de pantalla que muestra cómo se presentan los datos del historial de donaciones en la plataforma, incluyendo nombre del donante, cantidad, tipo de alimento y fechas clave. Esta funcionalidad permite a los usuarios revisar sus aportes previos.

**Figura 17**

*Vista funcional de la página de inicio de sesión en pruebas*



*Nota.* Captura de pantalla del entorno de pruebas donde se muestra la interfaz de inicio de sesión de la plataforma, diseñada para autenticar usuarios registrados antes de permitir el acceso a las funcionalidades principales.

### **Conclusiones**

Con base en el desarrollo y las pruebas realizadas, se concluye que la plataforma *Solidaridad Activa* cumple con los objetivos planteados inicialmente, ofreciendo una herramienta tecnológica sólida y funcional para facilitar la donación de alimentos no perecederos en la región.

- La plataforma logra conectar donantes con personas necesitadas, centralizando la gestión de alimentos mediante un entorno digital claro y confiable.

- La seguridad de los datos y la transparencia del sistema fortalecen la confianza de los usuarios en el proceso de donación.
- Gracias a su diseño adaptable y arquitectura modular, el sistema es escalable y fácilmente ampliable para futuras funcionalidades, como donaciones monetarias, campañas específicas, o integración con organizaciones.
- Se comprobó que el sistema es estable y usable, incluso para personas con bajo nivel de experiencia tecnológica, lo cual reafirma su pertinencia para el contexto social de Turbo.
- Se identificaron áreas de mejora, documentadas en la sección de “Posibles mejoras futuras”, que permitirán fortalecer aún más el impacto del proyecto en fases posteriores.

### **Posibles Mejoras Futuras**

A medida que la plataforma **Solidaridad Activa** crezca en uso y relevancia, podrían implementarse varias mejoras para adaptarse mejor a las necesidades de los usuarios y aumentar su impacto. Algunas posibles mejoras futuras son:

#### **Expansión de Tipos de Donación**

- **Incorporación de Nuevos Artículos:** La plataforma podría ampliarse para permitir donaciones de otros bienes esenciales, como ropa, medicinas y artículos de higiene personal.
- **Personalización de Donaciones:** Ofrecer la posibilidad de incluir notas o especificaciones personalizadas en cada donación, permitiendo a los donantes expresar preferencias o recomendaciones.

### **Mejora en la Experiencia de Usuario (UX/UI)**

- **Aplicación Móvil:** Desarrollar una versión móvil de Solidaridad Activa facilitaría a los usuarios el acceso a la plataforma desde cualquier lugar, mejorando la usabilidad y aumentando la conveniencia.
- **Interfaz Multilingüe:** Incluir la opción de seleccionar el idioma de la plataforma para que sea accesible a una mayor diversidad de usuarios.
- **Recordatorios y Notificaciones:** Incorporar notificaciones automáticas o correos electrónicos que recuerden a los donantes la posibilidad de contribuir regularmente o informen sobre la situación actual de las necesidades.

### **Optimización de la Seguridad y la Privacidad**

- **Autenticación de Dos Factores (2FA):** Añadir un sistema de autenticación adicional que refuerce la seguridad en el inicio de sesión.
- **Certificación de Seguridad:** Implementar un protocolo de seguridad más avanzado (como SSL) y someter el sistema a auditorías de seguridad periódicas para garantizar la protección de la información del usuario.

### **Optimización de la Base de Datos y Funcionalidades Avanzadas**

- **Análisis de Datos de Donación:** Crear un sistema de reportes y análisis que permita identificar patrones de donación, como los tipos de alimentos más donados y las épocas con mayor actividad. Esto ayudaría a planificar campañas o iniciativas de donación específicas.
- **Categorización de Donaciones Urgentes:** Implementar una función que permita priorizar y destacar donaciones urgentes según la necesidad en cada zona geográfica o en función de eventos específicos.

## **Integración con Otras Plataformas y Servicios**

- **Colaboración con ONG y Organizaciones Locales:** Integrar la plataforma con otras organizaciones de caridad, creando una red colaborativa para ampliar la distribución y alcance de las donaciones.
- **Pago en Línea para Donaciones en Efectivo:** Implementar una opción de donación monetaria a través de plataformas de pago seguras (PayPal, tarjetas de crédito), para que quienes no tengan alimentos disponibles puedan contribuir económicamente.

## **Escalabilidad del Proyecto**

- **Migración a un Servidor Escalable:** La plataforma se planificará para ser alojada en un entorno de nube que permita un escalamiento vertical y horizontal, adaptándose a un número creciente de usuarios sin comprometer el rendimiento.
- **Uso de Frameworks Modernos:** Se considerará la implementación de frameworks como Laravel para el backend y React para el frontend, que facilitan el desarrollo modular y escalable.
- **Integración con Microservicios:** Se evaluará la arquitectura basada en microservicios para dividir la plataforma en componentes más pequeños, lo que permite una implementación más ágil y escalable.
- **Implementación de CI/CD:** La integración continua y la entrega continua se establecerán para automatizar pruebas y despliegues, reduciendo el tiempo de entrega de nuevas funcionalidades y mejorando la estabilidad del sistema.
- **Uso de Bases de Datos NoSQL:** Para mejorar el rendimiento en grandes volúmenes de datos, se estudiará la inclusión de bases de datos NoSQL como MongoDB para ciertas partes de la plataforma.

- Implementación de Inteligencia Artificial para Personalización: Se integrará IA para personalizar la experiencia del usuario, ofreciendo recomendaciones de donación basadas en sus historiales y preferencias.

## Conclusiones

La implementación de la plataforma "**Solidaridad Activa**" ha demostrado ser una solución viable para facilitar y optimizar la gestión de donaciones de alimentos no perecederos. A través del desarrollo de una interfaz intuitiva y la ejecución de pruebas de usabilidad, se evidenció que la mayoría de los usuarios pudieron interactuar exitosamente con el sistema, cumpliendo con las tareas principales como registro, inicio de sesión, realización de donaciones y consulta de historial.

Los resultados de las pruebas de usabilidad, evaluadas mediante la escala **System Usability Scale (SUS)** (Brooke, 1996), reflejan un nivel de satisfacción elevado, confirmando que la plataforma es amigable, accesible y eficiente. Además, la utilización de herramientas como simuladores de carga contribuyó a garantizar un desempeño técnico aceptable en diferentes dispositivos y entornos.

Como punto importante, se identificaron oportunidades de mejora que podrán ser abordadas en versiones futuras del sistema, tales como la optimización de los tiempos de carga en dispositivos de gama baja y la inclusión de mensajes de retroalimentación más claros durante el registro y la donación.

## **Medición de Impacto**

Con el objetivo de garantizar la mejora continua y evaluar el impacto social generado por la implementación de la plataforma "Solidaridad Activa", se establecen una serie de indicadores de desempeño y mecanismos de retroalimentación. Estas herramientas permitirán medir la efectividad de la plataforma, identificar oportunidades de optimización y fortalecer la participación de la comunidad en futuras iniciativas.

Para evaluar el impacto social y la efectividad de la plataforma "Solidaridad Activa" tras su implementación, se plantean los siguientes mecanismos de medición:

### **Indicadores Clave de Desempeño (KPIs):**

- Número de donaciones registradas mensualmente.
- Cantidad de familias beneficiadas por mes.
- Tiempo promedio desde la oferta de donación hasta la entrega.
- Nivel de satisfacción de usuarios (basado en encuestas de retroalimentación).

### **Instrumentos de Retroalimentación:**

- Encuestas de satisfacción aplicadas a donantes y beneficiarios después de cada donación, evaluando aspectos como facilidad de uso, efectividad de la entrega y percepción de impacto social.
- Formularios electrónicos breves integrados en la plataforma.

### **Monitoreo de Patrones de Donación:**

- Análisis de los tipos de alimentos más donados.
- Identificación de picos de donaciones en fechas específicas (por ejemplo, temporadas navideñas).
- Geolocalización de las zonas con mayor cantidad de beneficiarios.

Los resultados de estas mediciones permitirán ajustar las estrategias de promoción de la plataforma, mejorar procesos internos y fortalecer su impacto comunitario.

## Bibliografía

- Álvarez Ríos, C. D. J., Barrera Acuña, M. A., & Oro Escaño, A. M. de. (2021). Diseño de una plataforma web de gestión y trazabilidad de donaciones de alimentos.  
<https://manglar.uninorte.edu.co/handle/10584/9508>
- Beizer, B. (2000). *\*Software Testing Techniques\** (2nd ed.). Dreamtech Press.
- Collaguazo Llumiquinga, E. A. (2015). Sistema web de donación y permuta “Cambio fácil.com” basado en herramientas libres. Universidad Técnica del Norte.  
<http://repositorio.utn.edu.ec/handle/123456789/7639>
- Mena Hidalgo, O. A., Quintana Suarez, J. L., 94524240, & 84091414. (2021). Desarrollo de aplicación para la gestión de donaciones de alimentos - share food. Universidad Antonio Nariño. <http://repositorio.uan.edu.co/handle/123456789/2309>
- MySQL. (2023). *\*MySQL 8.0 Reference Manual\**. <https://dev.mysql.com/doc/refman/8.0/en/>
- MySQL. (2023). MySQL 8.0 Reference Manual. En MySQL Documentation.  
<https://dev.mysql.com/doc/refman/8.0/en/>
- PHP Documentation Group. (2024). *\*PHP Manual: Password Hashing Functions\**.  
<https://www.php.net/manual/en/function.password-hash.php>
- PHP Documentation Group. (2024). Password Hashing. En PHP.net.  
<https://www.php.net/manual/en/function.password-hash.php>
- Quiroga Taborda, J. S. (2020). Análisis, diseño e implementación de una Web Móvil que facilite el proceso de donación de alimentos en la ciudad de Pereira. Universidad Católica de Pereira. <http://hdl.handle.net/10785/6207>
- Tardío Rubio, J. (2020). Diseño e implementación de trazabilidad y confianza de un sistema de donaciones y voluntariado con incentivos.

W3C. (2023). \*Web Content Accessibility Guidelines (WCAG) 2.1\*.

<https://www.w3.org/WAI/standards-guidelines/wcag/>

## Apéndice A

*Fragmento de código PHP para conexión a base de datos*



The image shows a screenshot of a GitHub code viewer interface. At the top, the repository path is "Solidaridadactiva / SolidaridadActiva / html / connect.php". Below the path, the user "YmenaBo" is identified, along with the repository name "SolidaridadActiva 1.0". The code viewer shows two tabs: "Code" (selected) and "Blame". The code statistics are "11 lines (10 loc) · 188 Bytes". A badge indicates "Code 55% faster with GitHub Copilot". The code itself is a PHP script for connecting to a MySQL database. It sets the host to "localhost", user to "root", and database to "login". It then creates a new MySQLi connection object and checks for any connection errors, outputting a message if one occurs.

```
1  <?php
2
3  $host="localhost";
4  $user="root";
5  $pass="";
6  $db="login";
7  $conn=new mysqli($host,$user,$pass,$db);
8  if($conn->connect_error){
9      echo "Failed to connect DB".$conn->connect_error;
10 }
11 ?>
```

*Nota.* Código utilizado para establecer la conexión entre la plataforma y la base de datos mediante PHP. Este fragmento permite acceder a las funcionalidades de consulta, inserción y autenticación.