Desarrollo de Software Articulo

Arquitectura de software para una aplicación web que gestiona donaciones online y divulgación de campañas para la fundación hogar divino amor de la ciudad de Aguachica – Cesar

by Karina Villamizar Sánchez y Saul Augusto Montaño Zapata – noviembre de 2023

**INTRODUCCIÓN**

El proyecto de desarrollo de una aplicación web para gestionar donaciones en línea y la divulgación de campañas para la Fundación Hogar Divino Amor en la ciudad de Aguachica, Cesar, representa una iniciativa valiosa y significativa. Esta aplicación se concibe como una herramienta esencial para la Fundación, permitiéndole recaudar fondos de manera eficiente y llegar a un público más amplio. La misión de la Fundación de ayudar a niños y niñas de escasos recursos del municipio de Aguachica se fortalecerá y se cumplirá con mayor efectividad a través de esta plataforma tecnológica.

El proyecto se basa en una arquitectura de software de tres capas que garantiza la funcionalidad, la seguridad y la escalabilidad. La capa de presentación proporciona una interfaz de usuario atractiva y fácil de usar para donantes y administradores. La capa lógica de negocio maneja la lógica detrás de las donaciones, la gestión de campañas y la generación de informes. La capa de almacenamiento garantiza la integridad de los datos y su acceso eficiente.

La seguridad y el control de datos son una prioridad en este proyecto, y se aplican prácticas sólidas para proteger la privacidad de los usuarios y la integridad de los datos. Además, se integran APIs externas como PayU Latam, Facebook Graph, Google Analytics y SendGrid para ampliar la funcionalidad y la conectividad de la aplicación.

Este proyecto se desarrollará siguiendo la metodología Scrum, lo que garantiza la adaptabilidad y la flexibilidad en el proceso de desarrollo. Se han definido 17 requerimientos funcionales y no funcionales que guiarán la implementación, y se han identificado las herramientas y tecnologías necesarias para el éxito del proyecto.

**ALCANCE**

El aplicativo web para gestionar donaciones online y divulgación de campañas para la Fundación Hogar Divino Amor de la  ciudad de Aguachica -Cesar  permitirá a la Fundación recaudar o registrar los recaudos financieros, alcanzar un mayor impacto en la comunidad y gestionar campañas de manera más efectiva, que permitan cumplir con la misión de seguir ayudando a niños y niñas de escasos recursos del municipio de Aguachica, además el aplicativo web estará desarrollado bajo los siguientes módulos: gestión de la información de las empresas o de los donantes, gestión de anuncios de campañas, seguimiento de impacto, gestión de donaciones, historial de donaciones, generación de recibos, informes financieros, privacidad y seguridad de la información e integración con redes sociales.

**METODOLOGÍA DE DESARROLLO DE SOFTWARE**

Para el desarrollo del proyecto, como se muestra en la figura 1, se usará el marco de trabajo SCRUM, es un marco ligero que ayuda a las personas, equipos y organizaciones a generar valor a través de soluciones adaptables para problemas complejos. (Ken Schwaber & Jeff Sutherland, noviembre 2020)

Scrum se basa en el empirismo y el pensamiento Lean. El empirismo afirma que el conocimiento proviene de la experiencia y la toma de decisiones basadas en lo que se observa. El pensamiento Lean reduce los desperdicios y se centra en lo esencial.

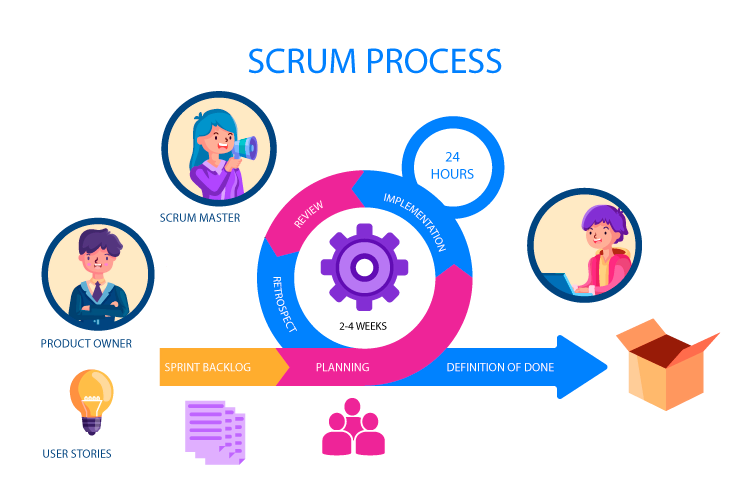


Figura 1 Marco de trabajo SCRUM

En pocas palabras, Scrum requiere un Scrum Master para fomentar un entorno donde:

* Un propietario del producto (Product Owner) ordena el trabajo de un problema complejo en un Product Backlog.
* El equipo de Scrum convierte una selección del trabajo en un Incremento de valor durante un Sprint.
* El equipo de Scrum y sus partes interesadas (stakeholders) inspeccionan los resultados y realizan los ajustes necesarios para el próximo Sprint.
* Repetir

**DEFINICIÓN Y ANÁLISIS DE REQUERIMIENTOS**

|  |  |  |
| --- | --- | --- |
| **REQUERIMIENTO** | **PRIORIDAD** | **DESCRIPCIÓN** |
| Registro de Usuarios | Media | El sistema debe permitir a los usuarios registrarse de manera sencilla y práctica, con campos como nombre, apellido y correo electrónico. |
| Visualización de Imágenes de Campañas | Alta | El sistema debe permitir al donante visualizar imágenes de la campaña para posteriormente donar por la campaña elegida. |
| Donaciones a través de Pasarelas de Pagos | Alta | El sistema debe permitir a los usuarios hacer donaciones por medio de pasarelas de pagos (PayU) para sus respectivas donaciones. |
| Registro de Datos de Donaciones | Alta | El sistema debe almacenar el registro de los datos, valores, fecha y bancos de todas las donaciones realizadas. |
| Certificados Digitales de Donación | Alta | El sistema debe permitir emitir certificados digitales de donación para las personas que hagan la donación. |
| Boletín de Impacto | Media | El sistema debe permitir informar a los donantes recurrentes el boletín del impacto de las donaciones a la comunidad. |
| Gestión de Campañas | Alta | El sistema debe permitir a los administradores agregar, editar y eliminar campañas. |
| Requerimientos de Seguridad | Alta | El sistema debe garantizar la seguridad de los datos y los usuarios, incluyendo autenticación segura, protección contra ataques de inyección de SQL y XSS, y cifrado de datos sensibles. |
| Requerimientos de Reportes | Media | El sistema debe generar informes financieros y de impacto según especificaciones detalladas, con formatos definidos. |
| Gestión de Roles y Permisos | Alta | El sistema debe permitir la gestión de roles y permisos para usuarios, garantizando que los administradores tengan acceso a funciones específicas. |
| Requerimientos de Pruebas | Media | El sistema debe definir casos de prueba, escenarios de prueba y criterios de aceptación para todas las funcionalidades, asegurando una prueba exhaustiva. |
| Métodos de Comunicación con Donantes | Alta | El sistema debe permitir la comunicación con los donantes a través de métodos como correo electrónico o notificaciones, bajo condiciones específicas. |
| Requerimientos de Copias de Seguridad | Alta | El sistema debe definir un proceso de copias de seguridad de datos críticos, garantizando la recuperación en caso de pérdida de información. |
| Requerimientos de Escalabilidad | Alta | El sistema debe ser escalable y capaz de manejar un crecimiento en el volumen de donaciones y usuarios. |
| Requerimientos de Rendimiento | Alta | El sistema debe cumplir con los niveles de rendimiento definidos, incluyendo tiempos de respuesta máximos y capacidad de procesamiento. |
| Requerimientos de Accesibilidad | Alta | El sistema debe cumplir con estándares de accesibilidad web, como WCAG, para garantizar la accesibilidad para personas con discapacidades. |
| Requerimientos de Auditoría y Cumplimiento | Alta | El sistema debe cumplir con regulaciones y estándares específicos, incluyendo requerimientos de auditoría y cumplimiento. |

**DEFINICIÓN DE REQUERIMIENTOS**

**Registro de Usuarios (Prioridad: Media):** La aplicación debe permitir a los usuarios registrarse de manera sencilla y práctica, utilizando campos como nombre, apellido y correo electrónico. Este requisito es esencial para identificar y gestionar a los donantes y usuarios.

**Visualización de Imágenes de Campañas (Prioridad: Alta):** La aplicación debe permitir a los donantes visualizar imágenes de las campañas, lo que facilita la comprensión de los objetivos y motiva las donaciones.

**Donaciones a través de Pasarelas de Pagos (Prioridad: Alta):** Los usuarios deben poder realizar donaciones de forma segura a través de pasarelas de pago, como PayU. Este requisito garantiza la facilidad y seguridad en las transacciones financieras.

**Registro de Datos de Donaciones (Prioridad: Alta):** La aplicación debe registrar de manera segura los datos relacionados con las donaciones, incluyendo información sobre montos, fechas y entidades bancarias involucradas. Esto es esencial para el seguimiento y la transparencia.

**Certificados Digitales de Donación (Prioridad: Alta):** La aplicación debe ser capaz de generar certificados digitales de donación para las donaciones realizadas por personas jurídicas. Esto permite la documentación y el cumplimiento de requisitos legales y fiscales.

**Boletín de Impacto (Prioridad: Media):** La aplicación debe permitir la entrega de un boletín que informe a los donantes recurrentes sobre el impacto de sus donaciones en la comunidad. Esto fomenta la retención y la participación continua de los donantes.

**Gestión de Campañas (Prioridad: Alta):** La aplicación debe permitir a los administradores agregar, editar y eliminar campañas. Esto garantiza la flexibilidad y la eficacia en la gestión de campañas.

**Requerimientos de Seguridad (Prioridad: Alta):** La aplicación debe garantizar la seguridad de los datos y de los usuarios, incluyendo la implementación de autenticación segura, protección contra ataques de inyección de SQL y XSS, y cifrado de datos sensibles. Esto es fundamental para proteger la integridad de la información.

**Requerimientos de Reportes (Prioridad: Media):** La aplicación debe ser capaz de generar informes financieros y de impacto de acuerdo con especificaciones detalladas, con formatos definidos. Esto es esencial para el seguimiento y la presentación de informes a los interesados.

**Gestión de Roles y Permisos (Prioridad: Alta):** La aplicación debe permitir la gestión de roles y permisos para usuarios, asegurando que los administradores tengan acceso a funciones específicas. Esto garantiza la seguridad y la eficacia de la administración.

**Requerimientos de Pruebas (Prioridad: Media):** La aplicación debe definir casos de prueba, escenarios de prueba y criterios de aceptación para todas las funcionalidades, asegurando una prueba exhaustiva. Esto es fundamental para garantizar la calidad del software.

**Métodos de Comunicación con Donantes (Prioridad: Alta):** La aplicación debe permitir la comunicación con los donantes a través de métodos como correo electrónico o notificaciones, bajo condiciones específicas. Esto facilita la interacción y la relación con los donantes.

**Requerimientos de Copias de Seguridad (Prioridad: Alta):** La aplicación debe definir un proceso de copias de seguridad de datos críticos, garantizando la recuperación en caso de pérdida de información. Esto asegura la integridad de los datos.

**Requerimientos de Escalabilidad (Prioridad: Alta):** La aplicación debe ser escalable y capaz de manejar un crecimiento en el volumen de donaciones y usuarios. Esto es importante para adaptarse a las necesidades cambiantes del proyecto.

**Requerimientos de Rendimiento (Prioridad: Alta):** La aplicación debe cumplir con los niveles de rendimiento definidos, incluyendo tiempos de respuesta máximos y capacidad de procesamiento. Esto asegura una experiencia de usuario óptima.

**Requerimientos de Accesibilidad (Prioridad: Alta):** La aplicación debe cumplir con estándares de accesibilidad web, como WCAG, para garantizar la accesibilidad para personas con discapacidades. Esto promueve la inclusión.

**Requerimientos de Auditoría y Cumplimiento (Prioridad: Alta):** La aplicación debe cumplir con regulaciones y estándares específicos, incluyendo requerimientos de auditoría y cumplimiento. Esto es fundamental para el cumplimiento de las obligaciones legales.

**ANÁLISIS DE REQUERIMIENTOS**

**Registro de Usuarios:** Asegurar que el proceso de registro sea eficiente y que se establezcan medidas de seguridad para proteger la información personal de los usuarios.

**Visualización de Imágenes de Campañas:** Garantizar que la aplicación admita la carga y visualización de imágenes de alta calidad, y que las imágenes sean compatibles con múltiples dispositivos y navegadores.

**Donaciones a través de Pasarelas de Pagos:** Implementar medidas de seguridad sólidas para proteger la información financiera de los donantes y asegurarse de que las transacciones sean fáciles de realizar.

**Registro de Datos de Donaciones:** Definir una estructura de base de datos que almacene de manera segura los registros de donaciones, asegurando la integridad y la disponibilidad de los datos.

**Certificados Digitales de Donación:** Especificar el formato y los detalles de los certificados digitales, y establecer un proceso automatizado para su generación.

**Boletín de Impacto:** Definir la frecuencia y el contenido del boletín, así como los métodos de entrega, para mantener a los donantes recurrentes informados sobre el impacto de sus donaciones.

**Gestión de Campañas:** Asegurar que los administradores tengan acceso a funciones de gestión de campañas que les permitan agregar, editar y eliminar campañas de manera eficiente.

**Requerimientos de Seguridad:** Implementar medidas de seguridad, como autenticación segura, protección contra ataques de inyección de SQL y XSS, y cifrado de datos sensibles, para proteger la integridad de los datos y la privacidad de los usuarios.

**Requerimientos de Reportes:** Definir los detalles de los informes financieros y de impacto, así como los formatos requeridos, para asegurar que los informes generados cumplan con las necesidades de la Fundación.

**Gestión de Roles y Permisos:** Asegurar que la aplicación permita la gestión de roles y permisos para usuarios, garantizando que los administradores tengan acceso a funciones específicas y que se respeten los principios de seguridad y privacidad.

**Requerimientos de Pruebas:** Definir casos de prueba, escenarios de prueba y criterios de aceptación para todas las funcionalidades, lo que es fundamental para garantizar la calidad del software y su correcto funcionamiento.

**Métodos de Comunicación con Donantes:** Especificar los métodos de comunicación, como correo electrónico o notificaciones, y las condiciones específicas bajo las cuales se enviarán las comunicaciones, para mantener una relación efectiva con los donantes.

**Requerimientos de Copias de Seguridad:** Definir un proceso de copias de seguridad de datos críticos para garantizar la recuperación en caso de pérdida de información y asegurar la continuidad del servicio.

**Requerimientos de Escalabilidad:** Asegurarse de que la aplicación sea escalable y pueda manejar un mayor volumen de donaciones y usuarios a medida que el proyecto crece.

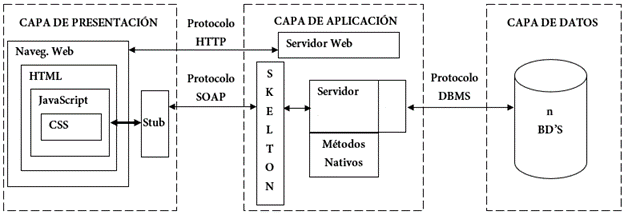
**Requerimientos de Rendimiento:** Establecer niveles de rendimiento específicos, incluyendo tiempos de respuesta máximos y capacidad de procesamiento, para garantizar una experiencia de usuario óptima.

**Requerimientos de Accesibilidad:** Asegurarse de que la aplicación cumpla con estándares de accesibilidad web, como WCAG, para garantizar la accesibilidad a personas con discapacidades.

**Requerimientos de Auditoría y Cumplimiento:** Definir requerimientos específicos para el cumplimiento de regulaciones y estándares aplicables, incluyendo requerimientos de auditoría y cumplimiento para garantizar la legalidad y la transparencia del proyecto.

**ARQUITECTURA DE SOFTWARE PARA EL DESARROLLO DEL PROYECTO.**

El desarrollo de software se refiere a un conjunto de actividades informáticas dedicadas al proceso de creación, diseño, despliegue y compatibilidad de software[[1]](#footnote-1), entendido esto, el software se desarrollará bajo la arquitectura de tres capas que consiste en dividir el desarrollo en tres capas: la Capa de Presentación (Interfaz de Usuario), la Capa Lógica de Negocio y la Capa de Almacenamiento y Mantenimiento tal y como lo muestra figura 2



**JDBC**

**JAVA**

Figura 2 Arquitectura de 3 Capas

**Herramientas tecnológicas, lenguajes de programación y base de datos.**

**Capa de Presentación:**

**Frontend (Interfaz de Usuario):**

* Visual Studio Code
* HTML y CSS
* Bibliotecas de estilos: Bootstrap (sass)
* Framework de JavaScript: React
* Gestión de estado: Redux

**Herramientas de Diseño:**

* Figma

**Capa Lógica de Negocio:**

**Backend (lógica de negocio):**

* Lenguaje de programación: Java (con Spring)
* Framework de servidor: Spring Boot (para Java)

**Seguridad:**

* Spring Security

**APIs:**

* pasarela de pago: API de PayU Latam
* redes sociales: Facebook Graph API
* Análisis y Seguimiento: Google Analytics API
* Notificaciones y Comunicación: SendGrid API

**Capa de Almacenamiento y Mantenimiento:**

**Base de Datos:**

* MySQL

**Herramientas de Gestión de Base de Datos:**

* MySQL Workbench

**Seguridad y Control:**

* HTTPS
* OWASP Java HTML Sanitizer
* Logback

**Control de Calidad del Código:**

* ESLint (para JavaScript)
* Checkstyle (para Java)

**Pruebas:**

* Jest y React Testing Library

**Gestión de Versiones:**

* Git y GitHub

**Comunicación y Colaboración:**

* Zoom

**Gestión de Proyectos:**

* Jira

**IDENTIFICACIÓN DE LOS COMPONENTES CLAVES DEL SOFTWARE**

En este apartado se identifican los componentes para el desarrollo del software, los cuales nos permiten tener un enfoque más claro y detallado.

* Sistema de pagos
* Sistema de autenticación
* Sistema de Generación de Certificados Digitales
* Sistema de Comunicación con Donantes
* Sistema de Copias de Seguridad
* Sistema de Reportes
* Sistema de Campañas
* Sistema Seguimiento

**MEDIOS DE COMUNICACIÓN Y MODELACIÓN DE LOS COMPONENTES EN EL SISTEMA.**

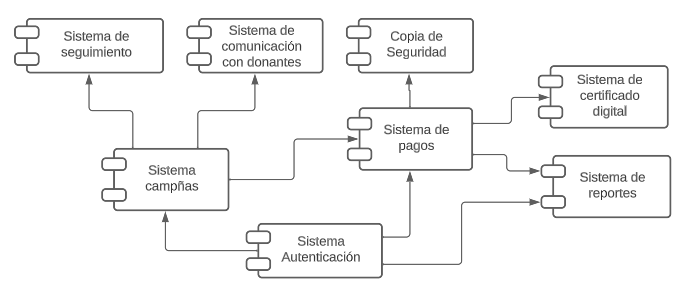


Figura 3 Medios de comunicación

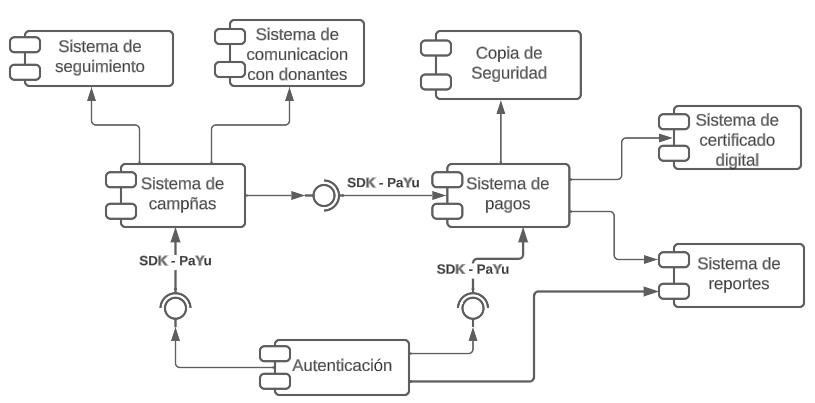


Figura 4 Modelación de componentes

**EQUIPO DE DESARROLLO**

* **Gerente de Proyecto:** Responsable de la gestión general del proyecto, utilizando la metodología Scrum para la gestión del proyecto.
* **Desarrollador Frontend React:** Crea la interfaz de usuario (UI) y (UX) utilizando React, HTML, CSS y JavaScript, Colabora en el diseño de la interfaz de usuario para garantizar un aspecto atractivo y una experiencia de usuario intuitiva.
* **Desarrollo Backend Java:** Construye la lógica empresarial y la capa de servidor utilizando Java, Spring y Spring Boot, administra la base de datos y garantiza un almacenamiento eficiente y seguro de los datos.
* **Seguridad y Control:** Especialista en Seguridad y Pruebas, Implementa medidas de seguridad y protección, asegurando la integridad de la aplicación y Realiza pruebas de seguridad y calidad para identificar vulnerabilidades y garantizar un funcionamiento seguro.

### ARQUITECTURA INTERNA DE LOS COMPONENTES

* **Módulo de Donaciones:**

**Frontend (Cliente):** Desarrollado con React y Redux para una interfaz de usuario interactiva y fácil de usar, se utiliza HTML y CSS para la estructura y el diseño, Bootstrap con SASS proporciona estilos modernos y responsivos, Se implementa la validación de datos en tiempo real para garantizar la precisión de la información ingresada por el donante.

**Backend (Servidor):** Implementado en Java con Spring Boot para un rendimiento eficiente y escalabilidad, los controladores REST gestionan las solicitudes de donación, validan los datos y dirigen la lógica de negocio los servicios de Java encapsulan la lógica empresarial y la interacción con la base de datos, Spring Security garantiza la autenticación segura de los usuarios y la autorización adecuada para las funciones críticas, los datos de donaciones se almacenan de manera segura en una base de datos MySQL utilizando Spring Data JPA.

**Almacenamiento:** Se utiliza una base de datos MySQL para almacenar los detalles de las donaciones, proporcionando confiabilidad y rendimiento y MySQL Workbench facilita la administración y el diseño de la base de datos.

**Seguridad y Control:** La comunicación entre el frontend y el backend se asegura mediante certificados SSL/TLS (HTTPS), OWASP Java HTML Sanitizer se utiliza para proteger contra ataques de inyección de HTML y XSS, Spring Security garantiza la autenticación segura y la autorización y Logback se encarga del registro y la auditoría de eventos para el control.

**Comunicación y Notificaciones:** Se utiliza la API de **SendGrid** para enviar correos electrónicos de agradecimiento y confirmación de donación a los donantes.

* **Módulo de Campañas:**

**Frontend (Cliente):** Se desarrolla en React y React Router para la navegación, permitiendo a los administradores gestionar campañas de manera intuitiva.

**Backend (Servidor):** Java y Spring Boot se emplean para la lógica de negocio relacionada con las campañas, controladores REST permiten a los administradores agregar, editar y eliminar campañas de manera eficaz.

* **Módulo de Generación de Informes:**

**Frontend (Cliente):** React se utiliza para crear un panel de informes que permite a los administradores generar informes financieros y de impacto de manera personalizada.

**Backend (Servidor):** La generación de informes se implementa en el backend, accediendo a la base de datos para recopilar datos y crear informes en formatos definidos.

* **Módulo de Comunicación con Donantes:**

**Frontend (Cliente):** Componentes permiten la comunicación con donantes recurrentes a través de métodos como correo electrónico o notificaciones, siguiendo condiciones específicas definidas en el backend.

**Backend (Servidor):** Se procesa y envía notificaciones a los donantes recurrentes utilizando métodos como el envío de correos electrónicos.

* **Integración de APIs Externas:**

**PayU Latam API:** Se integra para procesar donaciones y pagos en línea de manera segura.

**Facebook Graph API:** Se utiliza para compartir contenido y atraer a más donantes.

**Google Analytics API:** Se integra para analizar y medir el impacto de las campañas y el tráfico del sitio web.

**SendGrid API:** Se utiliza para enviar notificaciones y correos electrónicos a los donantes.

* **Seguridad y Control:**

La arquitectura se centra en la seguridad y el control, garantizando la protección de datos y usuarios, así como la autenticación segura y la autorización adecuada.

* **Gestión de Calidad y Desarrollo:**

La calidad del código se mantiene a través de herramientas como ESLint (para JavaScript) y Checkstyle (para Java).

Se realizan pruebas exhaustivas utilizando Jest y React Testing Library para asegurar la calidad del código y funcionalidad del sistema.

### CONCLUSIÓN

En conclusión, el proyecto de desarrollo de la aplicación web para la Fundación Hogar Divino Amor representa un paso significativo hacia el fortalecimiento de su misión de ayudar a niños y niñas necesitados en Aguachica, Cesar. La aplicación ofrecerá una plataforma eficiente para la recaudación de donaciones en línea, la gestión de campañas y la divulgación de su impacto en la comunidad.

La arquitectura de tres capas proporciona una base sólida para la aplicación, asegurando un rendimiento eficiente, seguridad de datos y escalabilidad. La seguridad y el control se consideran elementos esenciales en el diseño, garantizando que los datos de los donantes y las donaciones estén protegidos de manera efectiva.

La metodología Scrum permitirá un desarrollo ágil y adaptable, lo que es fundamental en un entorno en constante cambio. La integración de APIs externas agrega valor a la aplicación al proporcionar conectividad con pasarelas de pago, redes sociales y herramientas de análisis.

Este proyecto se presenta como una oportunidad para la Fundación de avanzar en su misión y lograr un mayor impacto en la comunidad. El trabajo conjunto entre el equipo de desarrollo y la Fundación será fundamental para el éxito de esta iniciativa, y esperamos que esta aplicación web se convierta en una herramienta poderosa en su lucha por un mundo mejor.

**REFERENCIAS**

**[1]** Ken Schwaber & Jeff Sutherland, La guía definitiva de SCRUM: las reglas del juego, 2020, [2020-Scrum-Guide-Spanish-European.pdf (scrumguides.org)](https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf)

1. [¿Qué es el desarrollo de software? | IBM](https://www.ibm.com/es-es/topics/software-development) [↑](#footnote-ref-1)