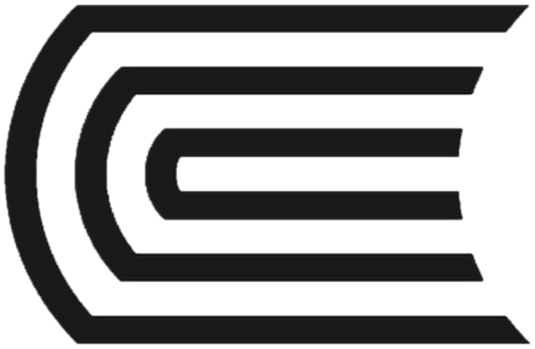
Universidad Continental

“Año de la recuperación y consolidación de la economía peruana”



INFORME

Plataforma inteligente de voluntariado ambiental comunitario

INTEGRANTES DEL GRUPO

Izarra Flores Jewilson Jenkens
Tomailla Contreras Alexis Anyelo

NRC: 62151

Huancayo – Perú 2025 - 10

1. Descripción general del sistema

El sistema propuesto es una plataforma digital web y móvil que conecta a ciudadanos interesados en participar en campañas ambientales como limpieza, reforestación y reciclaje con organizaciones locales que lideran estos proyectos.

Actualmente, los registros se realizan manualmente mediante formularios en papel, lo que genera duplicidad de datos, falta de seguimiento y escasa comunicación entre organizadores y voluntarios.

Con la nueva plataforma, se busca automatizar el registro, la asignación de tareas, la comunicación y el seguimiento del impacto ambiental generado por las actividades. El sistema permitirá medir indicadores como la cantidad de residuos recolectados, árboles plantados y horas de voluntariado realizadas.

El proyecto responde a la necesidad de una herramienta eficiente, moderna y accesible que mejore la gestión de campañas ecológicas y promueva una participación ciudadana más activa.

2. Actores principales

- **Voluntario:** persona que se registra en la plataforma para participar en campañas ambientales.
- **Organizador:** Encargado de crear campañas, asignar tareas y validar los resultados de los voluntarios.
- **Administrador:** Supervisa el correcto funcionamiento del sistema, gestiona usuarios y genera reportes globales.

3. Requerimientos del sistema

a) Requerimientos funcionales

1. El sistema debe permitir el registro y autenticación de usuarios
2. Los organizadores deben poder crear, editar y eliminar campañas ambientales.
3. Los voluntarios deben poder inscribirse en campañas disponibles.
4. Se debe permitir la asignación de tareas específicas a los voluntarios.
5. El sistema debe registrar métricas de impacto ambiental, residuos recolectados, árboles plantados, horas de trabajo.
6. Los usuarios deben poder comunicarse en tiempo real mediante chat o notificaciones.
7. El sistema debe generar reportes e indicadores visuales del impacto ambiental.

b) Requerimientos no funcionales

1. **Disponibilidad:** Accesible 24/7 desde la web y dispositivos móviles.

2. **Seguridad:** Autenticación con contraseñas cifradas y validación de roles.
3. **Escalabilidad:** Posibilidad de crecer en cantidad de usuarios y campañas sin afectar el rendimiento.
4. **Usabilidad:** Interfaz intuitiva y amigable, adaptada a distintos tipos de usuarios.
5. **Rendimiento:** Respuesta rápida (< 3 segundos por transacción promedio).
6. **Mantenibilidad:** Código modular con documentación técnica.
7. **Portabilidad:** Compatible con sistemas Android, iOS y navegadores modernos.

4. Casos de uso representativos

Actor	caso de uso	descripción
Voluntario	Registrarse y Iniciar sesión	Permite crear una cuenta y acceder a la plataforma
voluntario	Inscribirse en campaña	Solicita participar en una campaña disponible.
organizador	Crear campaña ambiental	Define los datos de una nueva campaña ecológica.
organizador	Asignar tareas	Asigna responsabilidades específicas a los voluntarios.
voluntario	Registrar impacto ambiental	Reporta sus resultados (residuos, árboles, horas).
organizador	Validar impacto	Revisa y aprueba los registros enviados.
administrador	Generar reportes globales	Obtiene estadísticas y métricas consolidadas.

5. Diseño arquitectónico del sistema

a) Tipo de arquitectura seleccionada: Arquitectura multicapa (n-tier)

b) Justificación técnica:

Se eligió una arquitectura multicapa porque permite organizar el sistema en módulos independientes que facilitan la escalabilidad, mantenimiento y reutilización del código.
Esta estructura separa claramente las responsabilidades de cada parte del sistema:

1. **Capa de presentación:**
   Interfaces web y móvil que interactúan con el usuario final.
   (ReactJS / Flutter).

2. **Capa de negocio:**
   Lógica de aplicación, validaciones y reglas de negocio.
   (Node.js / Java Spring Boot).

3. **Capa de datos:**
   Base de datos relacional (PostgreSQL o MySQL) que almacena la información de usuarios, campañas, tareas e impactos.

4. **Capa de servicios externos:**
   APIs de geolocalización, mensajería en tiempo real (Firebase o WebSocket).

Esta arquitectura facilita la integración futura de **microservicios** o módulos adicionales como gamificación o ranking de voluntarios.

6. Modelo relacional del sistema (para Erwin Data Modeler)

Entidades principales:

- **Usuario:** usuario_id, nombre, apellido, email, rol, estado, fecha_registro
- **Perfil Voluntario:** perfil id, usuario_id, dirección, fecha_nacimiento, certificaciones
- **Campaña:** campaña id, organizador id, título, descripción, ubicación, fecha_inicio, fecha_fin, cupo, estado
- **Tarea:** tarea id, campaña _id, responsable_id, título, descripción, estado
- **Inscripción:** inscripción id, usuario_id, campaña _id, estado, fecha_inscripción
- **RegistroImpacto:** registro_id, campaña_id, usuario_id, kg_residuos, árboles_plantados, horas_voluntariado, fecha_registro

- **Notificación:** notificación_id, usuario_id, mensaje, leido, fecha

Relaciones principales:

- Un Usuario tiene un solo Perfil Voluntario.
- Un Usuario puede participar en varias Campañas.
- Una Campaña puede tener muchas Inscripciones, Tareas y Registros de impacto.
- Un Usuario puede generar múltiples Notificaciones.

Justificación del modelo:

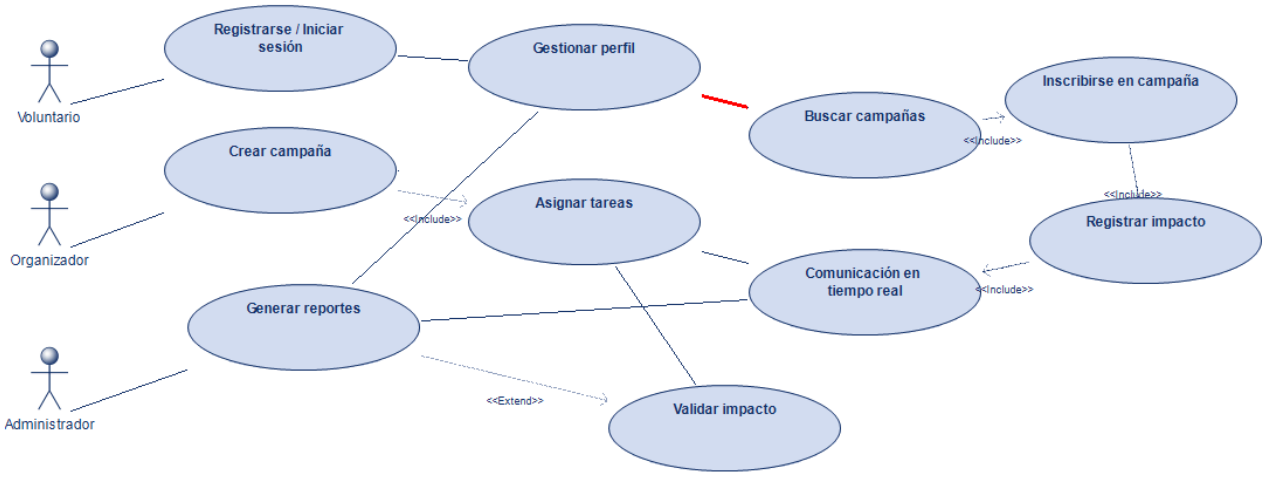
El diseño sigue la Tercera Forma Normal (3FN), evitando redundancias y garantizando la integridad referencial entre las tablas.
Cada entidad tiene una clave primaria única (PK) y claves foráneas (FK) para mantener la coherencia entre datos.
Este modelo es flexible y permite añadir nuevas métricas o funcionalidades sin modificar la estructura central.

7. Diagramas UML (para Modelio)

a) Diagrama de Casos de Uso

Muestra las interacciones entre los actores (Voluntario, Organizador, Administrador) y el sistema.
Incluye casos principales: registro, inscripción, gestión de campañas, comunicación y reportes.
Relaciones de tipo include y extend permiten identificar dependencias y acciones opcionales.

diagrama caso uso



b) Diagrama de Clases

Representa las entidades lógicas del sistema Usuario, Campaña, Tarea, Inscripción, Registro Impacto, Notificación, Chat con sus atributos y relaciones.
Permite visualizar la estructura de datos y las dependencias entre objetos.

c) Diagrama de Secuencia

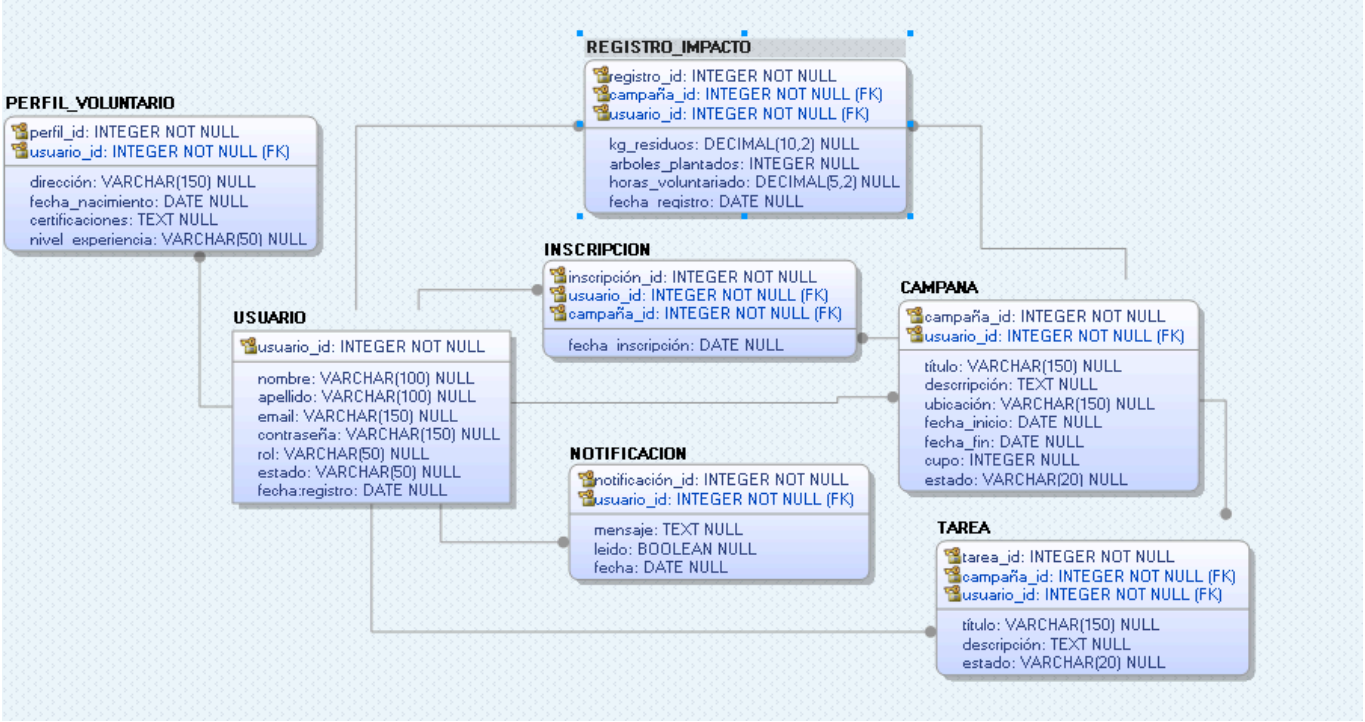
Ejemplo: flujo de inscripción de

voluntario ⟶ validación ⟶ confirmación ⟶ notificación

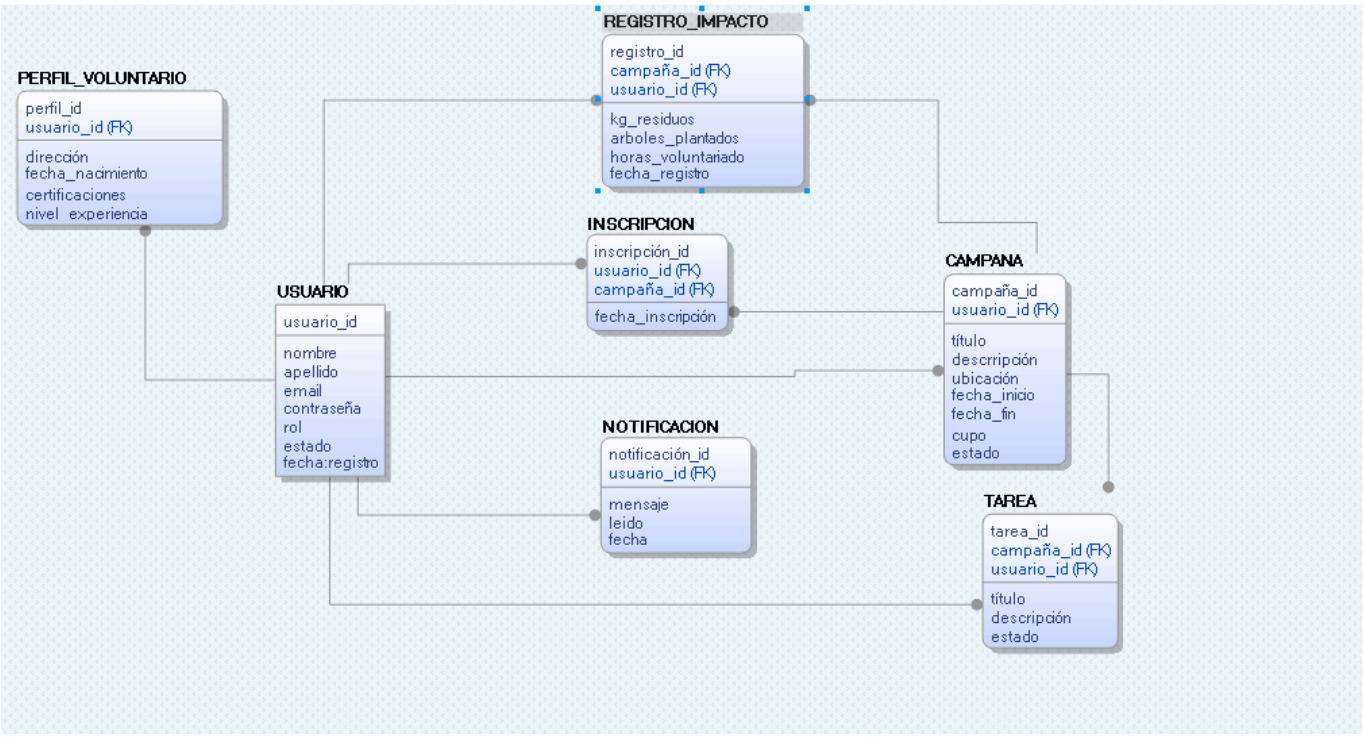
Ayuda a comprender cómo se comunican los componentes a lo largo de un proceso.

hecho en erwin data modeler

Logical



Physical



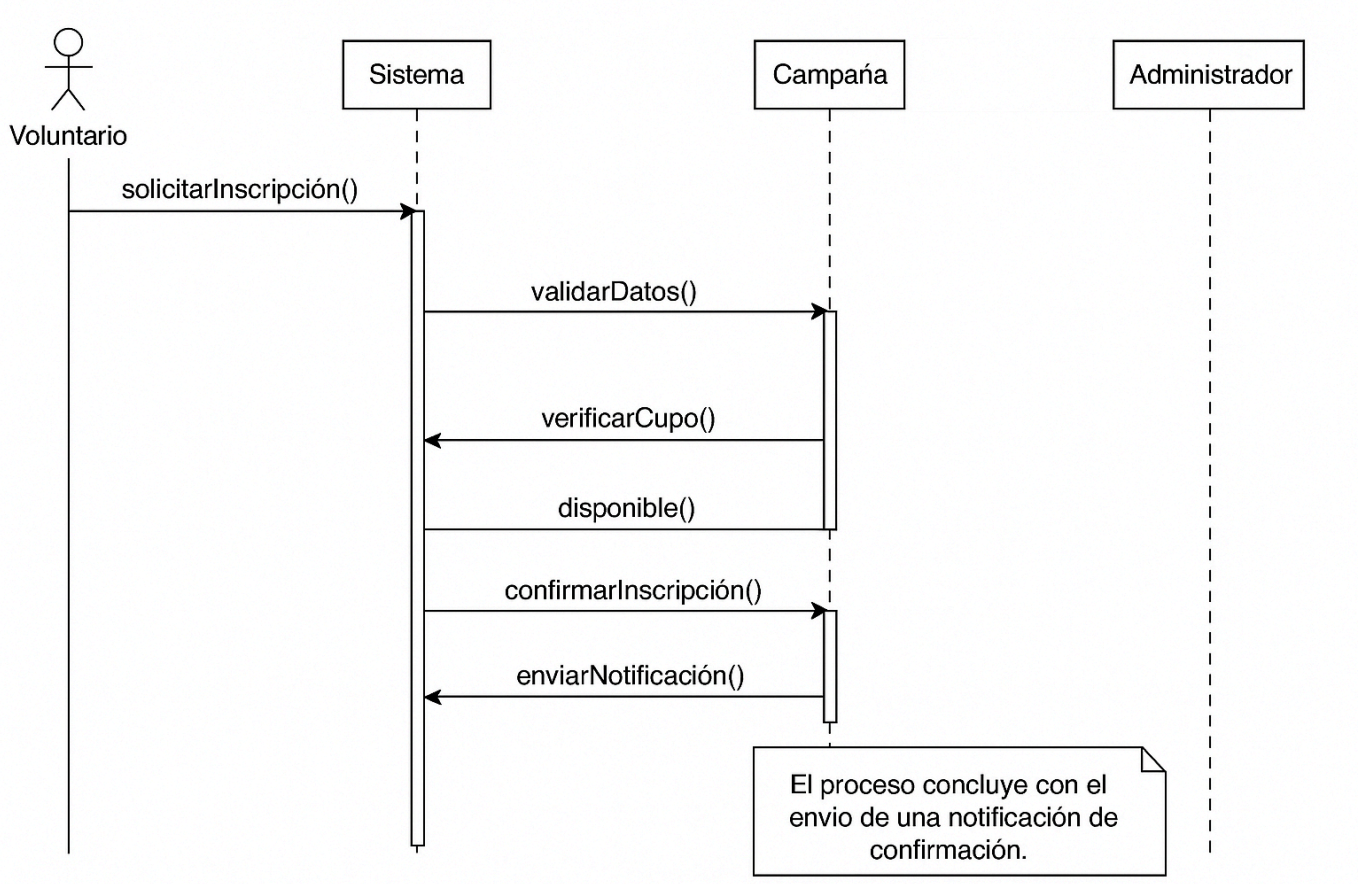
d) Diagrama de Componentes

Incluye módulos principales: API Gateway, Servicios de Usuarios, Campañas, Reportes, Chat, y Base de Datos.
Facilita visualizar la comunicación entre las partes del sistema.

e) Diagrama de Despliegue

Muestra la estructura física:

- Dispositivos de usuario web/móvil
- Servidor de aplicaciones.
- Servidor de base de datos.
- Servicios externos (Google Maps, Firebase).



8. Diseño de red y móvil

a) Diseño en red

- Los clientes (web y móvil) acceden a través de Internet a un servidor web seguro (HTTPS).
- El servidor de aplicaciones maneja la lógica del negocio y se comunica con la base de datos central.
- Se utilizan servicios en la nube (AWS, Azure o Firebase) para almacenar archivos y notificaciones.
- Se implementa autenticación JWT para la validación de usuarios.

b) Diseño móvil

- Aplicación híbrida (Flutter o React Native) conectada a la API central.
- Permite geolocalización en tiempo real y sincronización offline.
- Interfaz adaptable con menús simples para registro, campañas y tareas.

9. Justificación de decisiones de diseño

- **Arquitectura multicapa:** separa responsabilidades, mejora la mantenibilidad y permite escalar fácilmente.
- **Modelo relacional:** garantiza consistencia de datos e integridad referencial.
- **UML:** facilita la comprensión global del sistema antes de codificar.
- **Diseño de red y móvil:** prioriza la seguridad y conectividad entre los componentes.

10. Conclusiones

El diseño propuesto de la Plataforma Inteligente de Voluntariado Ambiental Comunitario permite una gestión moderna, escalable y eficiente de campañas ecológicas.

El uso de una arquitectura multicapa, junto con un modelo relacional bien normalizado y diagramas UML coherentes, garantiza un sistema sólido, mantenible y adaptable a futuras ampliaciones.

La propuesta facilita la interacción fluida entre organizadores y voluntarios, promueve la participación ciudadana activa y permite medir de forma objetiva el impacto ambiental logrado por cada campaña.