

22-8-2025

Proyecto Integrador



EcoCycle

Equipo 4
GENERATION

CONSUELO PINTO TORO
JOSÉ DÍAZ LILLO
CARLOS PEÑA MELLADO
FERNANDO GALDAMES VILCHES
CONSTANZA VALENZUELA BADILLA

Tabla de contenido

Introducción.....	2
Desarrollo: Frontend.....	3
Desarrollo: Backend	4
Desarrollo: Arquitectura General del Sistema	5
Flujo de Autenticación	6
Sistema de Reciclaje y Recompensas.....	7
API y Endpoints.....	8
Conclusión.....	10
Bibliografía.....	11

Introducción

El proyecto Ecocycle, alojado en el repositorio de GitHub bajo el usuario Conybri, y el backend se encuentra alojado en el repositorio CarlosAntonio representa una iniciativa innovadora en el ámbito de la sostenibilidad ambiental. Se trata de una plataforma tecnológica avanzada que integra un proyecto integrador y gamificación para revolucionar la gestión de residuos y promover prácticas ecológicas. Lanzado en 2025, el proyecto busca conectar a usuarios, educarlos y recompensarlos por acciones sostenibles, alineándose con los Objetivos de Desarrollo Sostenible (ODS) de las Naciones Unidas. Nuestra visión es transformar la gestión global de residuos en un ecosistema digital inclusivo y eficiente, reduciendo la huella ambiental mientras fomenta la participación comunitaria.

El proyecto **Ecocycle** se plantea como un **ecosistema digital integral para la gestión sostenible del reciclaje**, diseñado con una arquitectura modular y escalable que combina un **frontend en React/TypeScript**, un **backend en Spring Boot** y un sistema de **persistencia en MySQL**.

La **capa de presentación** está implementada en **React**, con dos versiones diferenciadas (1.0 y 2.0) que representan la evolución del sistema en términos de **usabilidad, diseño de interfaces y experiencia de usuario**. Estas aplicaciones actúan como puente entre los distintos perfiles del sistema —usuarios finales, recolectores y administradores— y la lógica de negocio central.

El **backend**, desarrollado en **Spring Boot**, responde a un modelo multicapa que organiza el código en **controladores, servicios y repositorios**. Este enfoque garantiza el desacoplamiento, la mantenibilidad y la escalabilidad del sistema. A través de **endpoints REST**, expone la funcionalidad principal del ecosistema:

- **Gestión de usuarios y autenticación** mediante **JWT**, que asegura control de acceso y sesiones seguras.
- **Registro y trazabilidad de residuos**, permitiendo documentar y supervisar cada proceso de reciclaje.
- **Sistema de recompensas y gamificación**, donde los usuarios obtienen **EcoTokens** como incentivo por su participación activa.
- **Módulo de impacto ambiental**, que mide indicadores sostenibles basados en la información registrada en la base de datos.

En la **capa de persistencia**, Ecocycle emplea **MySQL** como gestor de base de datos relacional. A través de **Spring Data JPA**, se definen entidades, relaciones y repositorios que permiten realizar operaciones **CRUD** de manera eficiente, asegurando la **integridad referencial** de los datos y optimizando la escalabilidad del sistema.

El ecosistema también contempla **pruebas unitarias e informes de validación**, que verifican el correcto funcionamiento de los endpoints y la integridad de los datos persistidos. Además, se aplican **buenas prácticas de ingeniería de software**, como modularización del código, seguridad mediante validación de entradas y CORS, y preparación para despliegues distribuidos en contenedores (Docker).

En conjunto, **Ecocycle** representa una solución tecnológica sólida y en constante evolución, cuyo objetivo es **digitalizar y optimizar los procesos de reciclaje**, garantizando **transparencia, trazabilidad y sostenibilidad ambiental** a través de una infraestructura moderna y robusta.

Desarrollo: Frontend

El frontend de **Ecocycle** está desarrollado con un stack moderno y escalable:

- **React y Next.js**: permiten crear una interfaz dinámica con soporte para Server-Side Rendering (SSR) y optimización de rendimiento.
- **TypeScript (TS)**: aporta tipado estático y mayor robustez en el desarrollo.
- **Tailwind CSS**: facilita la creación de interfaces modernas, responsivas y consistentes.
- **Framer Motion**: se emplea para animaciones fluidas e interactivas, mejorando la experiencia de usuario (UX).

Este conjunto de tecnologías garantiza que la aplicación sea **rápida, accesible y escalable**, alineándose con los objetivos de sostenibilidad y participación ciudadana que persigue el proyecto.



Desarrollo: Backend

El backend de **Ecocycle** se encuentra desarrollado principalmente en **Java**, utilizando el framework **Spring Boot**, lo que permite implementar una arquitectura modular, escalable y orientada a microservicios. Existen dos repositorios relacionados que trabajan de forma complementaria:

1. Back-render-eco

Este repositorio constituye una primera versión de la **API de usuarios demo**.

- Implementa endpoints básicos para el registro y gestión de usuarios.
- Incluye un archivo de inicialización (register_admin.json) que permite crear un usuario administrador al levantar el sistema por primera vez, facilitando la gestión inicial.
- La aplicación está preparada para su despliegue en contenedores mediante un Dockerfile, lo que garantiza portabilidad y estandarización entre entornos.
- Su estructura basada en **Maven** (pom.xml, mvnw, mvnw.cmd) asegura la correcta administración de dependencias y la facilidad de compilación.

Este módulo demuestra las capacidades iniciales de autenticación y gestión de usuarios que más tarde se integran en el ecosistema completo de Ecocycle.

2. Spring ecocycle

Este módulo corresponde al **backend principal del proyecto Ecocycle**, también desarrollado en **Spring Boot**.

- Su estructura sigue el patrón estándar de una aplicación Spring:
 - **Controladores** (@RestController) que exponen los endpoints REST al frontend.
 - **Servicios** (@Service) que contienen la lógica de negocio.
 - **Repositorios** (@Repository) para la interacción con la base de datos.
- Permite la comunicación con el **frontend** (desarrollado en React + Next.js) a través de un **API Gateway** que distribuye solicitudes a distintos microservicios.
- Entre las funcionalidades más relevantes se encuentran:
 - **Gestión de usuarios y roles**, con autenticación basada en **JWT**.
 - **Operaciones CRUD** para entidades propias del ecosistema (usuarios, puntos de reciclaje, logros de gamificación, etc.).
 - Integración con otros módulos:
 - **IA y Machine Learning** (predicciones de rutas, reconocimiento de materiales).
 - **Gamificación** (gestión de recompensas y badges).
 - **Geolocalización** (mapas y optimización de rutas).
- Al igual que back-render-eco, cuenta con un Dockerfile que facilita su despliegue en entornos de producción junto con el resto de microservicios del sistema.

3. Rol del Backend en el Ecosistema Ecocycle

En el contexto del **Ecocycle Project**, ambos backends cumplen una función esencial:

- **Autenticación y seguridad** → mediante usuarios, roles y JWT.

- **Gestión de datos** → permiten almacenar, consultar y modificar información sobre usuarios y actividades del sistema.
- **Comunicación con el frontend** → actúan como intermediarios entre la interfaz de usuario y los servicios de inteligencia artificial, gamificación y geolocalización.
- **Despliegue escalable** → gracias a Docker y la modularidad de Spring Boot, el backend puede crecer e integrarse con nuevas tecnologías como blockchain y AR en fases futuras del roadmap.

Podemos decir que back-render-eco es una **implementación demo** de la API de usuarios. Sprint ecocycle es el **backend central** que integra controladores, servicios y repositorios para todo el sistema. Ambos garantizan que el proyecto Ecocycle tenga una base sólida de **seguridad, modularidad y escalabilidad**, permitiendo la comunicación fluida entre el frontend y los microservicios inteligentes.

Desarrollo: Arquitectura General del Sistema

El sistema Ecocycle es una **plataforma web integral** diseñada para gestionar la participación ciudadana en iniciativas de reciclaje y educación ambiental, combinando interactividad, seguimiento de usuario y gamificación.

El **usuario** accede a la plataforma mediante un **frontend desarrollado en React.js**, que funciona como la interfaz principal para interactuar con las distintas funcionalidades del sistema. Desde el frontend, el usuario puede:

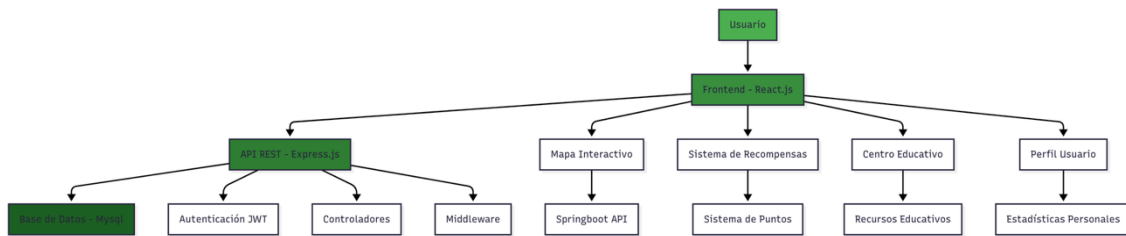
- Consultar un **mapa interactivo** que muestra ubicaciones relevantes, utilizando la **API de SpringBoot**.
- Acceder a un **sistema de recompensas**, donde acumula puntos y logros por su participación.
- Consultar el **centro educativo**, que proporciona recursos y material de aprendizaje sobre reciclaje y sostenibilidad.
- Revisar su **perfil de usuario**, incluyendo estadísticas personales y progresos en el sistema.

El **backend**, implementado mediante **API REST con Express.js**, se encarga de procesar todas las solicitudes del frontend. Incluye:

- **Autenticación JWT**, que garantiza la seguridad y el control de acceso de los usuarios.
- **Controladores**, que gestionan la lógica de negocio y la interacción con la base de datos.
- **Middleware**, que permite validar datos, manejar errores y ejecutar funciones intermedias antes de procesar las solicitudes principales.

Los datos del sistema se almacenan en **Dbeaver**, lo que permite mantener información estructurada y no estructurada sobre usuarios, puntos acumulados, recursos educativos y estadísticas.

En conjunto, este sistema permite **una experiencia interactiva, segura y personalizada**, integrando educación ambiental, seguimiento de participación y recompensas gamificadas, garantizando la trazabilidad y el correcto funcionamiento de todas las funcionalidades.



Flujo de Autenticación

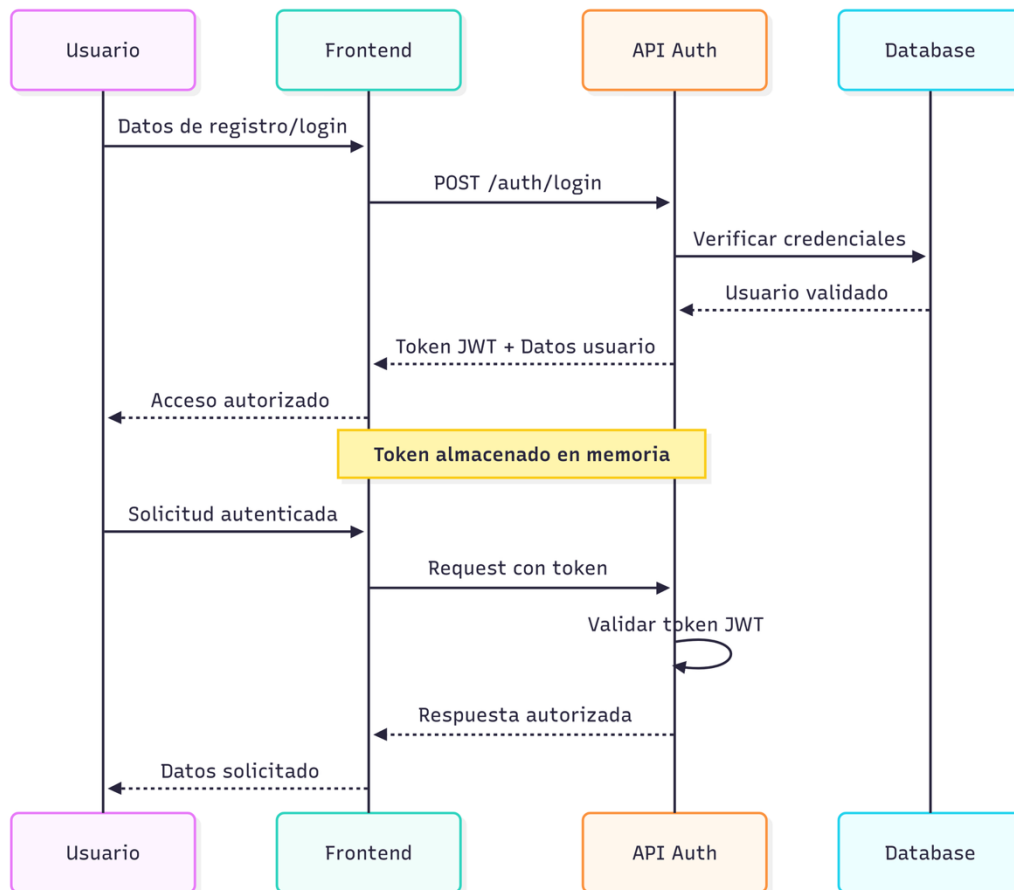
El diagrama muestra el **flujo de autenticación de un usuario** en el sistema, paso a paso:

1. Registro o login:

- El usuario envía sus datos (usuario y contraseña) al frontend.
- El frontend hace un POST /auth/login a la API de autenticación.
- La API verifica las credenciales en la base de datos.
- Si son correctas, la base de datos confirma que el usuario es válido.
- La API genera un **token JWT** y lo envía al frontend junto con los datos del usuario.
- El frontend permite el acceso al usuario y almacena temporalmente el token en memoria.

2. Solicitud autenticada:

- El usuario realiza una acción que requiere autorización, enviando el token al frontend.
- El frontend envía la solicitud a la API incluyendo el token JWT.
- La API valida el token para comprobar que sea válido y vigente.
- Si el token es correcto, la API responde con los datos solicitados.
- El frontend entrega los datos al usuario.



Sistema de Reciclaje y Recompensas

1. Registro de Actividad del Usuario

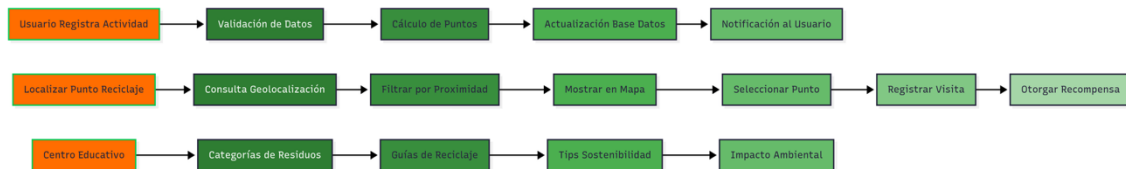
- **Usuario Registra Actividad (A):** El usuario ingresa información sobre su acción de reciclaje.
- **Validación de Datos (B):** El sistema verifica que la información ingresada sea correcta.
- **Cálculo de Puntos (C):** Se calculan los puntos o recompensas correspondientes a la actividad.
- **Actualización Base Datos (D):** Los puntos y la actividad se almacenan en la base de datos.
- **Notificación al Usuario (E):** El usuario recibe confirmación y detalle de los puntos obtenidos.

2. Localización de Puntos de Reciclaje

- **Localizar Punto Reciclaje (F):** El usuario busca un punto de reciclaje cercano.
- **Consulta Geolocalización (G):** El sistema obtiene la ubicación del usuario.
- **Filtrar por Proximidad (H):** Se seleccionan los puntos más cercanos al usuario.
- **Mostrar en Mapa (I):** Se despliegan los puntos en un mapa interactivo.
- **Seleccionar Punto (J):** El usuario elige un punto específico.
- **Registrar Visita (K):** El sistema registra la visita del usuario al punto.
- **Otorgar Recompensa (L):** Se asignan puntos adicionales por la visita.

3. Centro Educativo

- **Centro Educativo (M):** Área de aprendizaje dentro del sistema.
- **Categorías de Residuos (N):** Clasificación de los tipos de residuos.
- **Guías de Reciclaje (O):** Instrucciones para reciclar correctamente cada tipo de residuo.
- **Tips Sostenibilidad (P):** Consejos para reducir el impacto ambiental.
- **Impacto Ambiental (Q):** Información sobre cómo las acciones de reciclaje benefician al medio ambiente.



API y Endpoints

1. API REST Ecocycle

La **API central** funciona como intermediaria entre el frontend y la base de datos. Todos los módulos del sistema dependen de ella para procesar solicitudes y devolver datos.

2. Módulo de Autenticación (/api/auth/)

- **POST /register:** Permite registrar un nuevo usuario en el sistema.
- **POST /login:** Inicia sesión y devuelve un token JWT para que el usuario pueda autenticarse en futuras solicitudes.
- **POST /logout:** Cierra la sesión del usuario.
- **POST /refresh:** Renueva el token JWT cuando está a punto de expirar.

Este módulo asegura que **solo usuarios autenticados puedan acceder** a las funcionalidades protegidas.

3. Módulo de Usuarios (/api/users/)

- **GET /profile:** Devuelve los datos del perfil del usuario.
- **PUT /profile:** Permite actualizar información personal del usuario.
- **GET /stats:** Devuelve estadísticas del usuario, como puntos acumulados y actividades registradas.

Este módulo permite **gestionar la información personal y el seguimiento del progreso** de cada usuario.

4. Módulo de Puntos de Reciclaje (/api/recycling-points/)

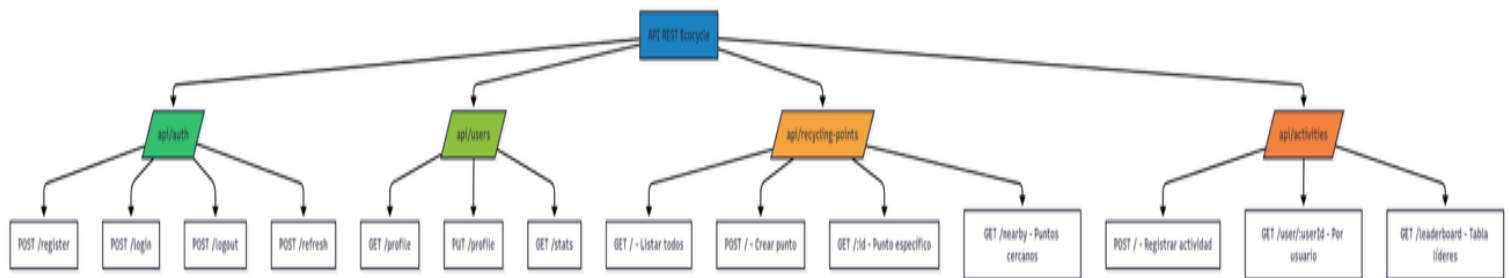
- **GET /:** Lista todos los puntos de reciclaje disponibles.
- **POST /:** Permite crear un nuevo punto de reciclaje.

- **GET /:id:** Obtiene información detallada de un punto específico.
- **GET /nearby:** Devuelve los puntos de reciclaje más cercanos a la ubicación del usuario.

Este módulo facilita **localizar, agregar y consultar puntos de reciclaje**.

5. Módulo de Actividades (/api/activities/)

- **POST /:** Registra una nueva actividad de reciclaje realizada por un usuario.
- **GET /user/:userId:** Muestra todas las actividades de un usuario específico.
- **GET /leaderboard:** Devuelve la tabla de líderes del sistema, mostrando los usuarios con más puntos.



Conclusión

El proyecto **Ecocycle** constituye una plataforma tecnológica completa y escalable que facilita la gestión de residuos, fomenta la educación ambiental y promueve la participación ciudadana mediante un sistema de recompensas y gamificación. Su arquitectura modular, que integra frontend, backend y base de datos, garantiza seguridad, trazabilidad y eficiencia en el registro y seguimiento de actividades de reciclaje.

Gracias a sus funcionalidades, como el flujo de autenticación seguro, la localización de puntos de reciclaje y el centro educativo, Ecocycle no solo optimiza los procesos de reciclaje, sino que también incentiva hábitos sostenibles y responsables. En conjunto, el proyecto demuestra cómo la tecnología puede contribuir de manera significativa a la sostenibilidad ambiental y al compromiso comunitario.



Bibliografía

Conybri. (2024). Ecocycle-Project. GitHub. Recuperado de <https://github.com/Conybri/Ecocycle-Project>

Naciones Unidas. (s.f.). Objetivos de Desarrollo Sostenible. Recuperado de <https://www.un.org/sustainabledevelopment/es/> (Referencia general a ODS mencionados en el proyecto).