



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

Universidad Nacional de Colombia - sede Bogotá  
Facultad de Ingeniería  
Departamento de Sistemas e Industrial  
Curso: Ingeniería de Software 1 (2016701)

Grupo 11

Alejandro Arguello Muñoz  
Juan Luis Vergara Novoa  
Steven David Alfonso Galindo  
Daniel Santiago Delgado Pinilla

### **Taller requerimientos y clean code para proyecto**

El objetivo principal del proyecto es aplicar los conceptos vistos en los módulo de requerimientos y clean code, llevando a cabo un proceso integral para **documentar** las necesidades de un sistema de software. Este ejercicio está diseñado para fortalecer sus habilidades en la conceptualización, análisis y diseño inicial de sistemas de software.

Al finalizar el taller, ustedes deben tener avanzado el MVP del proyecto final puesto que entramos a la fase de testing en 2 semanas (primera semana de febrero).

Esta entrega se hará en dos fases, **la primera es el 31 de Enero a las 23:59**, esta será preliminar, es decir, aunque no habrá nota deben entregar todo lo que hayan desarrollado hasta ahora. **Es obligatoria, quienes NO la hagan la nota de los dos módulos será evaluada sobre 4.0**, les daré retroalimentación y tendrán 1 semana para corregirlo, **el día 7 de febrero a las 23:59** será la entrega que tendrá su calificación.

## La estructura de entrega es la siguiente:

La mayor parte de la información del taller debe estar centralizada en un único documento PDF llamado **Proyecto\_Final.pdf**, que contendrá los puntos principales del proyecto (detallados más abajo). Sin embargo, algunos puntos requieren archivos adicionales que deben incluirse en carpetas específicas.

/Raiz..

```

|-- README.md          <<-- Archivo de presentación del proyecto.
|-- .gitignore         <<-- Configuración para evitar subir archivos innecesarios.
|-- Proyecto/         <<-- Carpeta con el código fuente del proyecto.
|   |-- (código fuente)
|
|-- Documentación/    <<-- Carpeta principal de la documentación.
|   |-- Proyecto_Final.pdf <<-- Documento central del proyecto.
|   |-- Casos_de_Uso/   <<-- Carpeta con los casos de uso individuales.
|       |-- Caso_de_uso_<nickname>_01.pdf
|       |-- Caso_de_uso_<nickname>_02.pdf
|
|   |-- Historias_de_Usuario/ <<-- Carpeta con historias de usuario individuales.
|       |-- historia_de_usuario_<nickname>_01.pdf
|       |-- historia_de_usuario_<nickname>_02.pdf
|
|   |-- Diagramas/     <<-- Carpeta con todos los diagramas del sistema.
|       |-- Diagrama_Arquitectura.pdf
|       |-- Modelo_Base_Datos.pdf
|       |-- Diagrama_Casos_de_Uso.pdf
|
|-- Clean_Code/       <<-- Carpeta con el informe de código limpio.
|   |-- Informe_codigo_limpio.pdf

```

**\*\*<nickname>** hace referencia a su usuario unal, por ejemplo el mio es “oalvarezr” por lo que los archivos quedan: Caso\_de\_uso\_oalvarezr\_01.pdf

Punto	Ubicación de calificable
Punto 1	/README.md
Punto 2	Documentación/Proyecto_Final.pdf
Punto 3	Documentación/Proyecto_Final.pdf
Punto 4	Documentación/Proyecto_Final.pdf
Punto 5	Documentación/Casos_de_Uso/* Documentación/Diagramas/Diagrama_casos_de_uso
Punto 6	Documentación/Historias_de_Usuario/*
Punto 7	Documentación/Clean_Code/Informe_codigo_limpio.

	pdf
Punto 8	Documentación/Proyecto_Final.pdf Documentación/Diagramas/*
Punto 9	Documentación/Proyecto_Final.pdf

## Punto 1: Pre Requerimientos

### SIN ESTE PUNTO NO SE REVISARÁN LOS SIGUIENTES.

En el archivo **README.md** del repositorio, incluyan lo siguiente:

1. **Título del repositorio:**
  - Ejemplo: Repositorio grupal - Ingeniería de Software 1 - 2024-2 Grupo #.
2. **Nombres de los integrantes del equipo.**
3. **Nombre del proyecto.**
4. **Icono o logo del proyecto (opcional).**
5. **Descripción del objetivo del proyecto:**  
Descripción de lo que trata el proyecto.
6. **Tecnologías a utilizadas:**
  - Lenguajes de programación.
  - Frameworks.
  - Servicios externos (bases de datos, servidores, APIs).
7. **Archivo .gitignore:**  
implementar el [.gitignore](https://www.toptal.com/developers/gitignore), les recomiendo esta pagina 😊  
<https://www.toptal.com/developers/gitignore>

## Punto 2: Levantamiento de requerimientos

Este punto se enfoca en identificar y documentar las necesidades del proyecto desde sus fundamentos. Para estructurar este apartado, utilicen las siguientes preguntas orientadoras a para redactar su texto a [manera de narración](#):

- **¿De dónde surge la idea?**  
Explicar cómo identificaron la necesidad del sistema. ¿Fue por observación, entrevistas, investigación o una combinación de estas? ¿Qué problemática busca resolver el software?
- **¿Cómo se pusieron de acuerdo para elegir la idea?**  
Describan el proceso de consenso dentro del equipo: reuniones, votaciones, asignación de roles, etc.
- **¿Cuáles son las problemáticas que buscan resolver?**  
Contextualicen los problemas que enfrenta el público objetivo. Justifiquen la relevancia de abordarlos con un sistema de software.
- **¿Qué expectativas tienen los usuarios potenciales del sistema?**  
Identifiquen las principales funcionalidades esperadas y los beneficios que el sistema debe proporcionar.
- **¿Qué beneficios esperan ustedes al desarrollar este proyecto?**  
Reflexionen sobre lo que cada integrante espera aprender y cómo el proyecto contribuye a su formación profesional.

El levantamiento debe hablar muy brevemente sobre las funcionalidades, identificarlas es el primer paso y estas deben estar conectadas con las problemáticas mencionadas.

Inicialmente, no teníamos ni idea de qué proyecto realizar. Para ser sinceros, no queríamos hacer algo muy complejo, principalmente por experiencias previas en otras asignaturas, donde por cuestiones de tiempo siempre terminamos realizando las cosas sobre el tiempo. Para contextualizar cómo surgió esta idea, al principio formamos un grupo de tres integrantes que ya nos conocíamos por algunas asignaturas anteriores y en general, porque habíamos ingresado a la carrera al mismo tiempo. Como ya mencionamos, ninguno de nuestros proyectos anteriores nos entusiasmaba y buscábamos en general queríamos realizar algo diferente, algo que nos pudiera motivar sin complicarnos mucho la vida. En realidad, no exploramos muchas posibilidades, simplemente terminamos evaluando dos opciones.

La primera idea consistía en desarrollar una aplicación para el manejo y control de parqueaderos. Su origen se remonta a que uno de los miembros, un día sin nada que hacer, se puso a vagar por Google Maps y, al leer reseñas de distintos centros comerciales, notó que una queja recurrente se refería al manejo de los parqueaderos (aunque, como el autor de esta idea y persona que está escribiendo esto, no recuerdo con exactitud qué decían). La otra idea en la que se fundamenta nuestro proyecto fue propuesta por otro integrante del grupo; un día, mientras se disponía a botar la basura, se percató de que no sabía en qué contenedor debía hacerlo, dejando finalmente dicha decisión en manos del azar. Por otro lado, el tercer miembro nunca quiso aportar una idea >:(.

La elección de cuál idea desarrollar se basó en lograr un consenso entre los tres. Aunque la idea del parqueadero no nos disgustaba, consideramos que la aplicación relacionada con la basura nos permitiría adaptarla más a nuestro estilo. Además, se nos ocurrían más funcionalidades para esa propuesta que para la otra, probablemente debido a nuestro desconocimiento en el tema, y quizá porque trabajar en la idea del parqueadero podría terminar afectando más nuestra salud mental.

Probablemente, a estas alturas de la historia, se estará preguntando porque no hemos mencionado al cuarto integrante, o tal vez no. En todo caso, se trata de una persona que conocimos este semestre, en un acto casi irresponsable de nuestra parte, jamás socializamos de manera amplia la idea del proyecto entre los cuatro. Sin embargo, a él no le pareció mala la idea 👍.

Aunque no hemos investigado estudios o estadísticas, basándonos en nuestra experiencia empírica, desde el colegio, se nos enseña a ser responsables con lo que desechamos. Sin embargo, en la práctica, muchas personas desconocen dónde depositar correctamente la basura. Hemos observado que, para concientizar e informar, se utilizan centros educativos, talleres y medios de comunicación, sin embargo, jamás hemos visto (o escuchado, probablemente por nuestra nula investigación) estrategias que hagan un gran uso de la tecnología. En pocas palabras, si tuviéramos que especificar la problemática que queremos abordar, se trataría de facilitar el acceso a la información sobre el desecho de residuos al público en general, mediante una aplicación que, además de simplificar este proceso, motive a las personas a ser más responsables con sus desechos, como por ejemplo con un sistema de rachas como Duolingo (Si no es muy difícil de hacer xd).

Queremos que la aplicación sea sencilla de usar para todo tipo de público, de modo que desde un niño hasta una persona mayor pueda manejarla sin complicaciones. Además, esperamos que los usuarios sientan que aprenden algo nuevo cada vez que la utilizan.

Entre las funcionalidades, algo que teníamos claro desde el principio es que queríamos usar IA, por lo que el escáner debe ser capaz de identificar la categoría de basura a la que pertenece haciendo uso de la inteligencia artificial. No solo queremos que los usuarios sepan dónde depositar la basura, sino también que, al escanear un objeto, reciban información básica sobre el tipo de residuo y el motivo por el que debe ir en ese contenedor.

Otro aspecto importante es que la información no se pierda al cerrar la aplicación. Por ello, consideramos esencial contar con un registro de los últimos desechos escaneados, de forma que se pueda acceder a esta información en cualquier momento o por lo menos durante cierto periodo de tiempo.

Finalmente, teniendo en cuenta que no todas las personas disponen de conexión a internet en todo momento, nuestra intención es que la aplicación funcione también sin conexión, aunque esto implique la pérdida de algunas funcionalidades.

Probablemente, a estas alturas, la aplicación, pese a contar con una premisa muy bonita, lo cierto es que probablemente muy poca gente la descargaría (ni siquiera nosotros jajaja). Por ello, hemos pensado en incorporar funcionalidades adicionales que la hagan más llamativa. Nos gustaría incluir un sistema de rachas, desafíos y estadísticas que incentiven a los usuarios a ser más responsables con sus desechos. Aunque esto sería algo mucho más opcional. Si bien el objetivo principal de la aplicación es ofrecer un acceso sencillo a la información sobre la clasificación de residuos, contribuyendo así al cuidado del medio ambiente, también nos interesa hacerlo de una forma entretenida. De esta manera, la app no solo informará, sino que también motivará y divertirá a sus usuarios.

Hablando de lo que esperamos obtener al elaborar este proyecto, como grupo consideramos que esta asignatura es fundamental para nuestra carrera. Aunque en cursos anteriores hemos trabajado en distintos proyectos, este es, sin duda, el que posee un mayor detalle técnico en el que ha participado cualquiera de los cuatro. Por ello, más allá de aprender a utilizar las tecnologías que aplicaremos, lo que más nos interesa es comprender en profundidad la estructura que debe tener un proyecto de este tipo y todo lo que ello implica. Además claro, esperamos ampliar nuestros conocimientos sobre la clasificación de residuos :).

(El texto en general fue mejorado con IA)

### Punto 3: Análisis de requerimientos

Realicen un análisis detallado de cada funcionalidad utilizando el método **MoSCoW (Must, Should, Could, Won't)**. Además, para cada funcionalidad, añadan una **estimación de esfuerzo con la secuencia de Fibonacci** (1, 2, 3, 5, 8, 13, etc.). Argumenten por qué llegaron a cada estimación considerando factores como:

- Complejidad técnica.
- Recursos disponibles.

- Impacto en la experiencia del usuario.
- Relación con los objetivos del proyecto.

**NOTA:** utilice tablas, gráficos, alguna técnica para organizar la información, piensen que le van a presentar esto a alguien que los quiere contratar...

## Punto 4: Análisis gestión de software

Analicen la **triada de gestión de proyectos** para su proyecto. Investigar y documentar lo siguiente:

- **Tiempo:**  
¿Cuánto tiempo estiman que tomará el desarrollo de cada funcionalidad o módulo? Detallen cómo se dividirán las etapas del desarrollo (por ejemplo, diseño, desarrollo, pruebas).
- **Costo:**  
Investigar cuánto cuesta desarrollar software considerando factores como:
  - Sueldos de desarrolladores (junior, senior, arquitectos, ux, testers).
  - Licencias, servicios en la nube, bases de datos.
  - Herramientas externas (APIs, frameworks, software de desarrollo).
  - Infraestructura (servidores, almacenamiento).

Saquen un total en una tabla organizada partiendo en que solo mostrarán lo que necesiten dentro de los costos (por ejemplo solo ocupamos un desarrollador junior)

- **Alcance:**  
Definan los límites del sistema. ¿Qué características o funcionalidades están incluidas y cuáles quedan fuera del alcance del MVP? Sean claros para evitar ambigüedades en futuras iteraciones del proyecto.

**NOTA:** Aunque esto no lo vimos en clase, lo que quiero que intenten es que investiguen en internet cuánto vale realmente su trabajo, esto lo deben siempre en el mundo laboral real, es la intención del ejercicio, no usen IA, o terminarán regalándose por 50k a la semana (?).

## Punto 5: Casos de uso del sistema

Elaboren casos de uso exclusivamente para las funcionalidades clasificadas como **MUST** en el análisis MoSCoW. Cada estudiante deberá realizar un mínimo de **tres casos de uso** es decir, si no son suficientes los MUST, utilicen los SHOULD.

Documenten los casos de uso con los siguientes elementos como mínimo:

- Nombre del caso de uso.
- Descripción breve.
- Actor(es) involucrado(s).
- Flujo principal.
- Flujos alternativos (si aplica).
- Precondiciones y postcondiciones (si aplica).

Además, complementan los casos de uso con su diagrama (**ES SÓLO UNO POR EL SISTEMA**), para ambos grupos su diagrama tiene que verse parecido a esto... [URL](#) aquellos que se equivocaron en el taller, espero lo corrijan, los revisaré con detalle 🤔🤔

**NOTA:** Sé que anteriormente había dicho que todas las funcionalidades (MoSCoW) pero hubo grupos que han hecho demasiados casos de uso, la idea es que aprendan a como hacerlos, no ahogarlos en repetir el proceso, a los que hayan avanzado y hayan hecho más de los que les pido, me excuso desde ya.

### **Punto 6: Historias de usuario**

Para cada caso de uso generado en el punto anterior, cada estudiante debe crear su respectiva **historia de usuario**, con un mínimo de **tres historias por persona**. Utilicen la plantilla proporcionada. Hasta ahora, todos utilizan correctamente la plantilla.



## Punto 2

### Requerimientos Funcionales

A continuación, se detallan los requerimientos funcionales identificados para EcoScan, organizados por prioridad y asociados a procesos específicos dentro del software:

#### Escáner Inteligente

- **Descripción:** Permite a los usuarios tomar una foto de un objeto para que la aplicación lo clasifique automáticamente en la categoría correcta (reciclaje, orgánico, basura general, etc.).
- **Proceso Asociado:** Procesamiento de imágenes y clasificación automática.
- **Especificaciones Técnicas:**
  - Integración de tecnología de visión por computadora.
  - Algoritmos de aprendizaje automático para mejorar la precisión del reconocimiento.

#### Historial de Escaneos

- **Descripción:** Permite a los usuarios registrados guardar un historial de los objetos escaneados para evaluar sus hábitos de reciclaje y recibir consejos personalizados.
- **Proceso Asociado:** Almacenamiento y gestión de datos de usuario.
- **Especificaciones Técnicas:**
  - Interfaz de usuario para visualizar el historial.

#### Registro Opcional de Usuarios (Opcional)

- **Descripción:** Los usuarios pueden registrarse para acceder a funcionalidades adicionales como el historial de escaneos.
- **Proceso Asociado:** Gestión de autenticación y perfiles de usuario.
- **Especificaciones Técnicas:**
  - Integración con servicios de autenticación (correo electrónico, redes sociales).
  - Seguridad en el almacenamiento de credenciales.

#### Recordatorios y Desafíos Ambientales (Opcional)

- **Descripción:** Envía recordatorios de reciclaje y propone desafíos ambientales, como "Recicla 10 botellas esta semana".
- **Proceso Asociado:** Gestión de notificaciones y motivación del usuario.
- **Especificaciones Técnicas:**
  - Sistema de notificaciones push.
  - Módulo para crear y gestionar desafíos.

#### Reconocimiento de Materiales con Consejos Específicos

- **Descripción:** Además de clasificar un objeto, detalla el material del que está hecho y ofrece consejos específicos para su limpieza o preparación para reciclaje.
- **Proceso Asociado:** Análisis detallado de materiales.
- **Especificaciones Técnicas:**
  - Base de datos de materiales y sus características.
  - Algoritmos para identificar y asociar materiales con consejos específicos.

### Racha de Usuario (Opcional)

- **Descripción:** Implementa un sistema de puntos basado en la cantidad de objetos escaneados correctamente y se acumula cada día pero se pierde tras varios días sin uso.
- **Proceso Asociado:** Gamificación y motivación del usuario.
- **Especificaciones Técnicas:**
  - Interfaz para visualizar la racha.

### Notificaciones Educativas

- **Descripción:** Envía notificaciones con tips ambientales, datos curiosos o recordatorios de hábitos sostenibles.
- **Proceso Asociado:** Distribución de contenido educativo mediante notificaciones.
- **Especificaciones Técnicas:**
  - Módulo de gestión de notificaciones.

### Detección de Contaminación Cruzada(opcional)

- **Descripción:** Advierte sobre materiales que contaminan el reciclaje y sugiere cómo limpiarlos antes de desecharlos.
- **Proceso Asociado:** Análisis de calidad de residuos.
- **Especificaciones Técnicas:**
  - Algoritmos para detectar residuos contaminados.
  - Sistema de recomendaciones para limpieza.

### Personalización del Perfil (para Usuarios Registrados)

- **Descripción:** Permite a los usuarios establecer objetivos ambientales personalizados, como reducir la producción de plástico.
- **Proceso Asociado:** Gestión de perfiles y objetivos de usuario.
- **Especificaciones Técnicas:**
  - Interfaz de usuario para establecer y modificar objetivos.
  - Almacenamiento seguro de preferencias de usuario.

### Estadísticas Ambientales (Opcional)

- **Descripción:** Muestra estadísticas sobre el impacto de las acciones de reciclaje de los usuarios, así como estadísticas globales.
- **Proceso Asociado:** Análisis y visualización de datos.
- **Especificaciones Técnicas:**
  - Módulo de recopilación y procesamiento de datos.
  - Herramientas de visualización gráfica para estadísticas.

### Reconocimiento Sin Conexión

- **Descripción:** Permite realizar escaneos básicos sin necesidad de conexión a internet, almacenando los datos para sincronizarse posteriormente.
- **Proceso Asociado:** Operación offline y sincronización de datos.
- **Especificaciones Técnicas:**
  - Módulo de reconocimiento local con capacidad limitada.
  - Sistema de almacenamiento temporal y sincronización en la nube.

### Requerimientos No Funcionales

#### Accesibilidad y Usabilidad

- **Descripción:** La plataforma debe ser accesible desde dispositivos móviles, con una interfaz intuitiva y adaptada para usuarios con poca experiencia tecnológica.
- **Especificaciones Técnicas:**
  - Diseño responsivo compatible con Android.

#### Disponibilidad y Rendimiento

- **Descripción:** La aplicación debe funcionar predominantemente online, con opciones básicas disponibles offline y tiempos de carga razonables para una experiencia fluida.
- **Especificaciones Técnicas:**
  - Infraestructura en la nube para alta disponibilidad.
  - Optimización de recursos para minimizar tiempos de carga.

#### Personalización y Diseño

- **Descripción:** Diseño visual atractivo que refleje una identidad ambiental, utilizando una paleta de colores naturales y evitando elementos molestos para el usuario.
- **Especificaciones Técnicas:**
  - Uso de colores y símbolos universales para la clasificación de residuos.
  - Diseño UI/UX centrado en la simplicidad y la facilidad de uso.

## Punto 3 Análisis de requerimientos

Funcionalidad	Prioridad	Estimación	Tecnologías	Comentarios
---------------	-----------	------------	-------------	-------------

		(Fibonacci)	Necesarias	
Escáner Inteligente	Must Have	13	TensorFlow, OpenCV, Flutter	Función principal de la app.
Historial de Escaneos	Must Have	5	Firebase, Flutter	Permite a usuarios ver su actividad.
Registro de Usuarios	Could Have	5	Firebase Auth, Flutter	Facilita personalización y almacenamiento de datos.
Recordatorios y Desafíos Ambientales	Could Have	8	Firebase Cloud Messaging	Aumenta el engagement del usuario.
Reconocimiento de Materiales	Must Have	8	TensorFlow, OpenCV	Detalla composición del objeto escaneado.
Rachas de Usuarios	Could Have	8	Firebase, Flutter	Sistema de puntuación y competencia.
Notificaciones Educativas	Should Have	3	Firebase Cloud Messaging	Envía consejos y recordatorios.
Detección de Contaminación Cruzada	Should Have	8	TensorFlow	Identifica residuos contaminantes.
Personalización del Perfil	Should Have	5	Firebase, Flutter	Permite establecer objetivos personalizados.
Estadísticas Ambientales	Could Have	8	Firebase, D3.js	Visualización del impacto del usuario.
Reconocimiento Sin Conexión	Could Have	8	TensorFlow Lite	Permite uso offline con sincronización posterior.
Lector de Código de Barras	Won't Have	-		Identificar el tipo de material y obtener instrucciones

## Estimación de Recursos

### Equipo:

- **Desarrollador Backend:** Encargado de implementar la lógica de procesamiento de imágenes, integración de tecnologías así como la comunicación con la base de datos y servicios externos.
- **Desarrollador Frontend:** Responsable del desarrollo de la aplicación móvil y de la integración de las varias interfaces, y notificaciones.

- **Product Manager / Analista (rol dual):** Rol dual o compartido con los anteriores y quién supervisa la integración de requerimientos, la calidad del código y la comunicación con el cliente.

### Equipo:

### Duración Total del Desarrollo

Se estima que el proyecto se desarrollará en aproximadamente 30 días.

1. **Investigación y Diseño:** 1 semanas
  - Definición de la arquitectura, elaboración de prototipos y validación de requerimientos.
2. **Desarrollo e Integración:** 3 semanas
  - Implementación de las funcionalidades principales identificadas durante el análisis para tener un MVP.
3. **Pruebas y Ajustes:** 1 semanas
  - Pruebas funcionales, ajustes en la interfaz y optimización del rendimiento.

### Estimación de Costos

Basándonos en información actual de plataformas laborales en Colombia, y considerando un equipo de 4 personas trabajando durante 30 días, la estimación es la siguiente:

Recurso	Cantidad	Costo Unitario (COP)	Total Estimado
Desarrollador Backend	1	COP 150,000/día	COP 4,500,000
Desarrollador Frontend	1	COP 150,000/día	COP 4,500,000
Product Manager / Analista / QA	2	COP 150,000/día	COP 4,500,000

### Alternativas de Reducción de Costos

1. **Reutilización de Código y Componentes Open Source:**
  - Utilizar plantillas y módulos ya desarrollados para Python y TensorFlow y plantillas de React puede reducir el tiempo de desarrollo.
  - Utilización de Transfer Learning.
2. **Priorización de Funcionalidades (MVP):**
  - En la fase inicial se implementarán únicamente las funcionalidades **Must Have**. Las funcionalidades opcionales se desarrollarán en iteraciones futuras.
3. **Roles Dualizados:**

- En algunas etapas, se puede aprovechar la versatilidad del equipo (por ejemplo, un desarrollador Fullstack que asuma tareas tanto de backend como de frontend) para optimizar recursos.

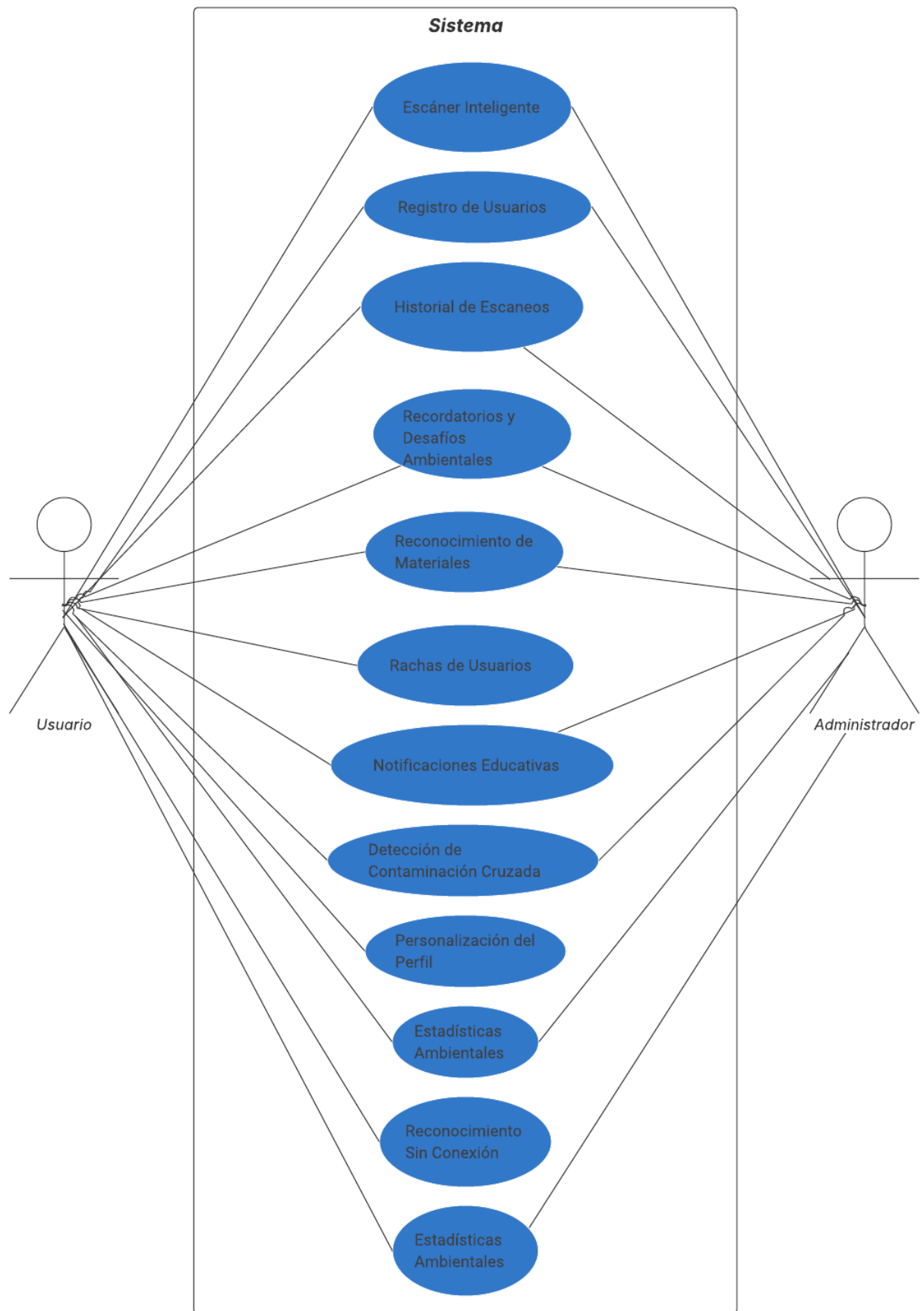
### Tecnologías Propuestas

- **Backend:**
  - **Lenguaje y Frameworks:** Python con TensorFlow y OpenCV para el procesamiento de imágenes y reconocimiento de materiales.
  - **APIs y Servicios:** Integración con Firebase Cloud Messaging para notificaciones y Firebase Auth para la autenticación.
- **Frontend:**
  - **Framework:** Flutter para el desarrollo de una aplicación móvil multiplataforma (Android, y opcionalmente iOS).
  - **UI/UX:** Diseño responsivo e intuitivo, basado en las últimas tendencias de diseño para aplicaciones móviles.
- **Base de Datos y Servicios en la Nube:**
  - **Firebase:** Para autenticación, almacenamiento de datos (historial de escaneos, perfiles de usuario) y mensajería.
  - **Infraestructura:** Servidores en la nube (por ejemplo, Google Cloud Platform o AWS) para asegurar alta disponibilidad y escalabilidad.

### Alcance

- **Incluido en el MVP:**
  - **Escáner Inteligente:** Permite capturar una imagen y clasificarla automáticamente en categorías (reciclaje, orgánico, basura general, etc.).
  - **Historial de Escaneos:** Registro de actividades que permite a los usuarios evaluar sus hábitos de reciclaje.
  - **Registro de Usuarios:** Permite la creación de perfiles mediante autenticación con correo electrónico o redes sociales.
- **Fuera del Alcance y para actualizaciones futuras:**
  - Funcionalidades opcionales como Recordatorios y Desafíos Ambientales, Racha de Usuarios, Notificaciones Educativas avanzadas, Detección de Contaminación Cruzada, Personalización del Perfil y Estadísticas Ambientales.
- **Consideraciones Adicionales:**
  - La arquitectura debe ser escalable y modular para permitir la integración de nuevas funcionalidades.
  - Se priorizará una experiencia de usuario óptima, con tiempos de carga reducidos y un diseño intuitivo.

## Diagrama de Casos de Uso





Escaneo de objetos.
Actor: usuario registrado o no registrado
Breve descripción: El usuario puede subir una imagen del objeto y obtener una clasificación y recomendaciones para el reciclaje
<p><b>Precondiciones:</b></p> <ul style="list-style-type: none"> <li>• La aplicación está instalada y abierta.</li> <li>• Se tienen los permisos necesarios para acceder a la cámara del dispositivo.</li> </ul>
<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción “<b>Escanear objeto</b>” en la pantalla principal.</li> <li>2. La aplicación activa la cámara para que el usuario pueda capturar la imagen del objeto.</li> <li>3. El usuario toma una foto del objeto.</li> <li>4. La aplicación procesa la imagen a partir de un modelo entrenado de inteligencia artificial.</li> <li>5. Se muestra la clasificación del objeto (por ejemplo, reciclaje, orgánico, basura general) junto con consejos específicos para su tratamiento o preparación.</li> <li>6. Si el usuario está registrado             <ol style="list-style-type: none"> <li>a. El sistema guarda la información:                 <ol style="list-style-type: none"> <li>i. Tiempo de escaneo</li> <li>ii. Tipo de objeto escaneado</li> <li>iii. Imagen</li> <li>iv. nombre de usuario</li> </ol> </li> <li>b. El sistema actualiza la base de datos.</li> </ol> </li> </ol>
<p><b>Flujos alternos</b></p> <p><b>A. Error en la captura:</b></p> <ul style="list-style-type: none"> <li>• <b>Si la imagen no es clara o el objeto no se reconoce, el sistema muestra un mensaje de error solicitando repetir el escaneo.</b></li> </ul>

Registro de usuario
Actor: usuario no registrado
Breve descripción: Permitir que un usuario se registre para acceder a funcionalidades adicionales, como guardar el historial de escaneos y personalizar objetivos ambientales.
<p><b>Precondiciones:</b></p>

<ul style="list-style-type: none"> <li>El usuario no tiene una cuenta activa en la aplicación.</li> </ul>
<ol style="list-style-type: none"> <li>El usuario selecciona el botón “Registrarse” desde la pantalla de inicio o durante el primer escaneo.</li> <li>El sistema abre un formulario de registro               <ol style="list-style-type: none"> <li>El usuario ingresa:                   <ol style="list-style-type: none"> <li>Correo electrónico</li> <li>Nombres</li> <li>Apellidos</li> <li>Contraseña</li> </ol> </li> </ol> </li> <li>El sistema valida la información ingresada</li> <li>El sistema actualiza la base de datos.</li> <li>El sistema notifica al usuario que el registro fue exitoso</li> <li>El sistema redirige a la pantalla principal o al proceso de escaneo.</li> </ol>
<p><b>Flujos Alternativos:</b></p> <ul style="list-style-type: none"> <li><b>A. Error de Validación:</b> <ul style="list-style-type: none"> <li>Si la información es inválida (por ejemplo, correo mal formado o contraseña débil), el sistema muestra un error y solicita al usuario corregir los datos. (Regresa a 2).</li> </ul> </li> <li><b>B. Cancelación:</b> <ul style="list-style-type: none"> <li>El usuario cancela el registro y continúa usando la aplicación en modo invitado (Regresa a 1).</li> </ul> </li> </ul>

Personificación del perfil
Actor: usuario registrado
Breve descripción: El usuario puede configurar y actualizar sus objetivos ambientales y preferencias personales, de modo que la aplicación ofrezca una experiencia personalizada y recomendaciones acordes a sus metas.
<p><b>Precondiciones:</b></p>
<ol style="list-style-type: none"> <li>El usuario ingresa a la sección de “Perfil” usando el botón desde el menú principal.</li> <li>La aplicación muestra la información actual del usuario, datos personales, objetivos ambientales y preferencias y un botón para editar .</li> <li>El usuario selecciona la opción “Personalizar Perfil”.</li> <li>El usuario puede modificar:               <ol style="list-style-type: none"> <li>Objetivos ambientales</li> <li>Preferencias de notificaciones</li> <li>Información Personal adicional</li> </ol> </li> <li>El usuario guarda los cambios mediante el botón de “Guardar”</li> </ol>

6. La aplicación valida la información ingresada
7. El sistema actualiza la interfaz con los cambios realizados

**Flujos Alternativos:**

- **A. Error en la Validación de Datos:**
  - Si alguno de los datos ingresados no cumple con los requisitos como campos obligatorios vacíos, el sistema muestra un mensaje de error indicando qué debe corregirse y se permite al usuario corregir la información.
- **B. Cancelación de la Edición:**
  - Si el usuario decide no realizar cambios, y selecciona el botón “Cancelar”, el sistema descarta las modificaciones y retorna a la vista del perfil sin alterar la configuración actual.

Historial de Escaneo

Actor: usuario registrado o no registrado

Breve descripción: Permite al usuario registrado acceder y revisar su historial de escaneos, filtrar por fecha o categoría, y obtener consejos personalizados basados en sus hábitos de reciclaje.

**Precondiciones:**

- El usuario ha iniciado sesión en la aplicación.
- El usuario ha realizado al menos un escaneo previo.
- La aplicación tiene acceso a una conexión a internet (para sincronización inicial).

**1.Inicio:**

El usuario selecciona la opción "Historial" desde la pantalla principal.

**2.Carga del historial:**

- El sistema recupera los datos del historial de escaneos asociados al usuario desde la base de datos.

Muestra una lista cronológica con:

- Miniaturas de las imágenes escaneadas.
- Fecha y hora del escaneo.
- Categoría del objeto (ej. "plástico", "vidrio").

**3.Filtrado (opcional):**

El usuario elige filtrar por:

- Fecha (ej. "últimos 7 días").
- Categoría (ej. "reciclaje", "orgánico").
- El sistema actualiza la lista según los filtros seleccionados.

#### **4.Detalle del escaneo:**

- El usuario selecciona un registro del historial.
- El sistema muestra detalles completos:
- Imagen original del objeto.
- Material identificado
- Estadísticas comparativas (ej. "Has reciclado un 20% más que el mes pasado").

#### **Cierre:**

- El usuario presiona "Volver" para regresar al historial o a la pantalla principal.

#### **Flujos Alternativos:**

##### **A1: Sin escaneos recientes**

- Si no hay registros en el historial, el sistema muestra un mensaje: "¡Comienza a escanear objetos para ver tu impacto aquí!" con un botón para redirigir al escáner.

##### **A2: Eliminar registro (opcional)**

- El usuario selecciona "Eliminar" en un registro.
- El sistema solicita confirmación.
- Tras confirmar, elimina el registro y actualiza la base de datos.

Registro de usuario

Actor: usuario no registrado

Breve descripción: Permitir que un usuario se registre para acceder a funcionalidades adicionales, como guardar el historial de escaneos y personalizar objetivos ambientales.

**Precondiciones:**

- El usuario no tiene una cuenta activa en la aplicación.

7. El usuario selecciona el botón “Registrarse” desde la pantalla de inicio o durante el primer escaneo.
8. El sistema abre un formulario de registro
  - a. El usuario ingresa:
    - i. Correo electrónico
    - ii. Nombres
    - iii. Apellidos
    - iv. Contraseña
9. El sistema valida la información ingresada
10. El sistema actualiza la base de datos.
11. El sistema notifica al usuario que el registro fue exitoso
12. El sistema redirige a la pantalla principal o al proceso de escaneo.

Recordatorios y Desafíos Ambientales

Actor: usuario no registrado

Breve descripción: Permite al usuario recibir recordatorios personalizados para mejorar sus hábitos de reciclaje y participar en desafíos ambientales con objetivos específicos, como "Recicla 5 botellas de plástico en 3 días".

**Precondiciones:**

- El usuario está registrado y ha iniciado sesión.
- El usuario ha habilitado las notificaciones push en la aplicación.
- La aplicación tiene acceso a internet (para sincronizar desafíos y enviar notificaciones).

**Configurar recordatorios:**

- El usuario navega a la sección "Desafíos y Recordatorios" desde el menú principal.
- Selecciona "Programar Recordatorio" y elige:
  - Frecuencia (ej. diario, semanal).
  - Hora preferida (ej. 18:00).
  - Tipo de recordatorio (ej. "Separar residuos", "Llevar bolsa reutilizable").
- El sistema guarda la configuración y confirma: "¡Recordatorio activado!".

**Unirse a un desafío:**

- En la misma sección, el usuario selecciona "Desafíos Disponibles".
- Elige un desafío (ej. "Recicla 10 latas esta semana") y presiona "Comenzar".

- El sistema inicia el contador del desafío y muestra un progreso (ej. "0/10 completado").

#### **Notificaciones automáticas:**

- En la hora programada, el sistema envía una notificación push con el recordatorio (ej. "¡Es hora de separar tus residuos orgánicos!").
- Para desafíos activos, el sistema envía actualizaciones periódicas (ej. "Te faltan 3 latas para completar el desafío").

#### **Completar un desafío:**

- El usuario cumple el objetivo (ej. escanea 10 latas).
- El sistema detecta el progreso y envía una notificación: "¡Desafío completado! Ganaste 50 puntos ecológicos".
- Actualiza el perfil del usuario con la recompensa.

#### **Flujos Alternativos:**

##### **A1: Editar o cancelar un recordatorio**

- El usuario selecciona "Mis Recordatorios" → "Editar".
- Modifica la frecuencia, hora o tipo, o elige "Cancelar Recordatorio".
- El sistema actualiza o elimina la configuración.

##### **A2: Abandonar un desafío**

- El usuario selecciona "Desafíos Activos" → "Abandonar".
- El sistema solicita confirmación y, al aceptar, elimina el desafío de su historial.

##### **A3: Notificación de inactividad**

Si el usuario no interactúa con la app por 3 días, el sistema envía una notificación motivacional: "¡Vuelve! Tu racha ecológica está en riesgo".

##### **A4: Desafíos sugeridos**

- El sistema propone desafíos basados en el historial del usuario (ej. "¿Reciclar 5

botellas de vidrio? ¡Gana 30 puntos!").

## Reconocimiento de Materiales

Actor: Usuario (registrado o no registrado).

Breve descripción: El sistema identifica el material principal de un objeto escaneado y ofrece consejos detallados para su preparación, limpieza o reciclaje, evitando la contaminación cruzada.

### Precondiciones:

- La aplicación está abierta y el usuario ha accedido al escáner.
- El dispositivo tiene acceso a la cámara y permisos activos (o funciona en modo offline con el módulo local cargado).
- El objeto escaneado está en el campo de visión de la cámara con suficiente claridad.

### Inicio del escaneo:

- El usuario selecciona "Escanear objeto" desde la pantalla principal.
- La cámara se activa y el usuario enfoca el objeto.

### Captura y procesamiento:

- El usuario toma la foto del objeto.
- El sistema procesa la imagen utilizando un modelo de IA entrenado para:
- Clasificar el objeto en una categoría (ej. "plástico PET", "vidrio verde").
- Identificar el material principal (ej. "aluminio", "cartón con recubrimiento de cera").

### Análisis de materiales:

- El sistema consulta una base de datos local o en la nube para:
- Determinar si el material es reciclable, compostable o debe desecharse como basura general.
- Detectar posibles contaminantes (ej. restos de comida en un envase de plástico).

### Mostrar resultados:

### La aplicación muestra en pantalla:

- Categoría principal: "Reciclaje (plástico PET)".
- Material detectado: "Botella de plástico PET grado 1".

**Consejos específicos:**

- "Lavar con agua para eliminar residuos líquidos antes de reciclar".
- "Retirar la etiqueta de papel y separarla del plástico".
- Advertencias (si aplica): "Este material no se recicla en tu localidad: consulta puntos de acopio especializados".

**Acciones adicionales (para usuarios registrados):**

Si el usuario está registrado, el sistema guarda automáticamente:

- Material identificado.
- Consejos aplicados.
- Fecha y hora del escaneo.

Se actualiza el "Historial de Escaneos" y las "Estadísticas Ambientales".

**Flujos Alternativos:**

**A1: Material no reconocido**

- Si el sistema no identifica el material, muestra: "No pudimos identificar este material. ¿Deseas intentarlo de nuevo o reportarlo para mejorar el servicio?".
- Opciones: "Reintentar" (vuelve a la cámara) o "Reportar" (envía la imagen al equipo técnico).

**A2: Material compuesto**

- Si el objeto tiene múltiples materiales (ej. una caja de pizza con grasa),
- **el sistema:** Destaca las capas o componentes (ej. "Cartón (contaminado con grasa)").
- Recomendación: "Separa la parte limpia (reciclable) y desecha la contaminada (basura general)".

**A3: Contaminación detectada**

Si el objeto está sucio o mezclado con residuos no compatibles:

- Muestra un mensaje destacado: "¡Este material contaminaría el reciclaje! Limpia



con [instrucciones] antes de desechar".

#### **A4: Modo offline**

##### **Si no hay conexión a internet:**

- El sistema usa un modelo de IA local reducido para identificar materiales básicos (ej. "vidrio", "plástico").
- Muestra consejos genéricos y sincroniza los datos con el servidor al recuperar la conexión.

#### Rachas de Usuarios

Actor: Usuario registrado

Breve descripción: Permite al usuario conocer y visualizar sus rachas de actividad, es decir, la cantidad de días consecutivos en los que ha realizado al menos un escaneo, incentivando el hábito de uso constante de la aplicación. Aunque se puede pasar máximo 3 días seguidos sin usar el escaner sin perder la racha.

##### **Precondiciones:**

- El usuario está registrado y ha iniciado sesión.
- La aplicación tiene acceso a internet (para sincronizar datos y actualizarlos).
- La fecha y hora del dispositivo deben estar sincronizadas con el sistema.

##### **Inicio de Sesión y Acceso a la Sección de Rachas:**

- El usuario inicia sesión en EcoScan.
- Desde el menú principal, selecciona la opción "Racha".

##### **Cálculo de la Racha Actual:**

- Al hacer un escaneo se incrementa un día de racha y se actualiza el dato.
- El sistema consulta el historial de escaneos del usuario.
- Se identifican las fechas consecutivas en las que se registró al menos un escaneo.
- El sistema determina si el usuario ha tenido inactividad. Se permite que el usuario pase hasta 3 días sin actividad sin reiniciar la racha.
- El sistema calcula la racha actual (por ejemplo, "3 días consecutivos").
- Si la inactividad es mayor a 3 días, la racha se reinicia (se considera que la racha anterior ha finalizado y se inicia una nueva).

##### **Visualización de la Racha:**

- Se muestra en pantalla la racha actual y, opcionalmente, la racha máxima alcanzada históricamente.
- Se despliega un mensaje motivador, por ejemplo: "¡Estás en una racha de 3 días! Sigue así para batir tu récord."

**Actualización en Tiempo Real:**

- Cada vez que el usuario realiza un nuevo escaneo en un día distinto, el sistema actualiza automáticamente la racha.
- Si el usuario tiene inactividad pero dentro del límite de 3 días, se mantiene la racha; de lo contrario, se reinicia y se muestra un aviso de reinicio.

**Flujos Alternativos:**

- **A1: Sin Actividad Reciente Dentro del Límite Permitido:**
  - Si el usuario no ha realizado escaneos en los últimos 1 a 3 días, el sistema muestra un mensaje: "¡Recuerda que puedes pasar hasta 3 días sin perder tu racha! Aprovecha para retomar tu actividad."
- **A2: Inactividad Excesiva:**
  - Si el usuario no realiza ningún escaneo por más de 3 días, el sistema reinicia la racha. Se muestra un mensaje: "Has superado el límite de 3 días sin actividad. Tu racha se ha reiniciado. ¡Vuelve a comenzar y establece una nueva racha!"
- **A3: Error en la Consulta del Historial:**
  - Si se produce un error al consultar el historial, se muestra un mensaje de error y se sugiere reintentar o contactar soporte.

Notificaciones Educativas

Actor :Sistema (acción automatizada) / Usuario registrado

Breve descripción: El sistema envía notificaciones push con consejos ambientales, datos curiosos y recordatorios de buenas prácticas de reciclaje. Estas notificaciones buscan educar y motivar al usuario de forma periódica.

**Precondiciones:**

- El usuario debe estar registrado.
- El dispositivo debe tener conectividad a internet o, en su defecto, la notificación se almacenará en el historial de la app.
- El sistema cuenta con una base de datos de contenidos educativos actualizada.

### **Configuración de Notificaciones:**

- De forma automática, el sistema selecciona un mensaje educativo o un tip ambiental desde la base de datos (por ejemplo, "¿Sabías que reciclar una tonelada de papel ahorra 17 árboles?").

### **Generación de Contenido**

- En la misma sección, el usuario selecciona "Desafíos Disponibles".
- Elige un desafío (ej. "Recicla 10 latas esta semana") y presiona "Comenzar".
- El sistema inicia el contador del desafío y muestra un progreso (ej. "0/10 completado").

### **Envío de Notificación:**

- A la hora programada o según la frecuencia configurada, el sistema envía una notificación push al dispositivo del usuario.
- Ejemplo de notificación: "Tip EcoScan: Recuerda separar los residuos orgánicos para mejorar el compostaje."

### **Interacción del Usuario:**

- Al recibir la notificación, el usuario puede hacer clic para ver más detalles o ignorarla.
- Si el usuario interactúa, se redirige a una sección de guías educativas o a un contenido relacionado.

### **Flujos Alternativos:**

- A1: Notificaciones Desactivadas:
  - Si el usuario ha desactivado las notificaciones, el sistema almacena los mensajes en el historial de notificaciones de la app, accesible desde el menú "Mis Notificaciones".
- A2: Error en el Envío:
  - Si la notificación falla al enviarse, el sistema reintenta el envío en un intervalo predefinido y, de persistir, registra el fallo para su monitoreo.

Detección de Contaminación Cruzada
Actor :Usuario (registrado o anónimo)
Breve descripción: El sistema analiza la imagen escaneada para detectar la presencia de contaminantes que puedan interferir con el proceso de reciclaje. Si se detecta contaminación, se ofrecen recomendaciones para limpiarlo o prepararlo adecuadamente
<p><b>Precondiciones:</b></p> <ul style="list-style-type: none"> <li>• La aplicación debe estar activa y el usuario debe haber realizado un escaneo utilizando la cámara.</li> <li>• El dispositivo tiene la cámara en funcionamiento.</li> <li>• La imagen capturada debe tener la calidad necesaria para permitir un análisis detallado.</li> </ul>
<p><b>Captura de Imagen:</b></p> <ul style="list-style-type: none"> <li>• El usuario activa el modo de escaneo y toma una foto del objeto a reciclar (similar al caso de uso "Escáner Inteligente").</li> </ul> <p><b>Procesamiento Inicial:</b></p> <ul style="list-style-type: none"> <li>• El sistema procesa la imagen con algoritmos de reconocimiento (TensorFlow) para clasificar el objeto.</li> <li>• Se realiza un análisis adicional específicamente para identificar signos de contaminación cruzada (por ejemplo, manchas, presencia de suciedad excesiva, mezcla con otros residuos).</li> </ul> <p><b>Detección de Contaminación:</b></p> <ul style="list-style-type: none"> <li>• Si el análisis detecta que el objeto presenta contaminación (por ejemplo, residuos sucios o mezclados), se activa un módulo de recomendaciones.</li> <li>• Se consulta una base de datos de recomendaciones para determinar las acciones a tomar (por ejemplo, "Lavar el objeto antes de reciclarlo" o "Separar el material contaminado").</li> </ul> <p><b>Visualización del Resultado:</b></p> <ul style="list-style-type: none"> <li>• El sistema muestra al usuario la clasificación del objeto junto con una advertencia de contaminación y recomendaciones específicas para su limpieza o preparación.</li> <li>• Ejemplo de mensaje: "Se ha detectado contaminación: Se recomienda lavar el objeto para mejorar su reciclaje."</li> </ul> <p><b>Confirmación y Registro:</b></p> <ul style="list-style-type: none"> <li>• El usuario puede confirmar la recomendación y optar por realizar otro escaneo tras la limpieza.</li> <li>• El resultado se registra en el historial, indicando que el objeto presentaba contaminación y la recomendación aplicada.</li> </ul>

**Flujos Alternativos:**

**A1: Imagen de Baja Calidad:**

- Si la imagen capturada no tiene la resolución suficiente para un análisis preciso, el sistema muestra un mensaje: "La imagen no es clara, por favor intente nuevamente."

**A2: No se Detecta Contaminación:**

- Si el análisis no detecta contaminación, el flujo continúa mostrando la clasificación estándar sin advertencias adicionales.

**A3: Error en el Procesamiento:**

- Si ocurre un error durante el análisis, el sistema informa al usuario y sugiere reintentar el escaneo.

Estadísticas Ambientales

Actor: Usuario registrado

Breve descripción: La funcionalidad de Estadísticas Ambientales permite al usuario visualizar de forma interactiva el impacto de sus acciones de reciclaje, así como ver datos agregados de toda la comunidad. Se muestran gráficos y cifras sobre reciclaje personal (por ejemplo, cantidad de residuos clasificados, puntos ganados, etc.) y estadísticas globales.

**Precondiciones:**

**Estadísticas personales**

- El usuario debe estar registrado e iniciar sesión.
- El usuario debe haber realizado acciones (escaneos, participación en desafíos, etc.) que permitan generar datos históricos.

**Estadísticas globales**

- El sistema debe haber recopilado y procesado datos de múltiples usuarios (al menos un conjunto mínimo de registros para que los gráficos sean significativos).
- La aplicación debe contar con acceso a internet para actualizar y sincronizar la información.

**Navegación a la sección:**

- El usuario, desde el menú principal, selecciona la opción estadísticas ambientales.

**Carga de datos:**

- El sistema solicita y recopila datos del usuario (por ejemplo, cantidad de escaneos, puntos ganados, desafíos completados) y datos globales (total de residuos reciclados, impacto acumulado, etc.) desde la base de datos.
- Se ejecutan procesos de análisis o se obtienen resúmenes preprocesados.

**Visualización de estadísticas:**

El sistema muestra una **vista interactiva** que incluye:

- **Gráficos personales:** Por ejemplo, un gráfico de barras o línea que muestre la evolución de las acciones de reciclaje del usuario a lo largo del tiempo, o un medidor de progreso hacia metas personales.
- **Estadísticas globales:** Indicadores de impacto comunitario (total de residuos reciclados, CO2 ahorrado, etc.) presentados en gráficos circulares, tablas, infografías, etc..

**Actualización dinámica:**

- El sistema actualiza la información de forma periódica para reflejar nuevos datos conforme el usuario realiza acciones.

**Confirmación de visualización:**

- El usuario interactúa con la vista, visualizando sus estadísticas personales y, si lo desea, explorando las métricas globales.
- Se muestra un mensaje o indicador de "datos actualizados" para confirmar la sincronización.

**Flujos Alternativos:**

#### **A1: Datos Insuficientes o Ausentes**

- El usuario no ha realizado suficientes acciones o aún no hay datos globales disponibles.
  1. El sistema detecta la falta de datos y muestra un mensaje informativo, por ejemplo:
    - Aún no has registrado suficientes acciones para generar estadísticas personales. ¡Empieza a reciclar y verás tus avances!.
    - Aún no se han acumulado suficientes datos a nivel global. Vuelve en unos días para ver el impacto de la comunidad.
  2. Se invita al usuario a interactuar con la aplicación para empezar a acumular datos.

#### **A2: Error en la Recuperación de Datos**

- Ocurre un error en la consulta o procesamiento de datos (problemas de conectividad, error en el backend, etc.).
  1. El sistema muestra un mensaje de error (por ejemplo, "No se han podido cargar las estadísticas. Inténtalo más tarde.").
  2. Se registra el error en los logs del sistema para posterior análisis.

#### **A3: Filtros y Personalización**

- El usuario decide aplicar un filtro (por ejemplo, visualizar estadísticas de reciclaje del último mes).
  1. El usuario selecciona el filtro deseado en la interfaz.
  2. El sistema vuelve a consultar y procesa la información de acuerdo al filtro y actualiza los gráficos y métricas en pantalla.

Reconocimiento sin conexión

Actor: usuario no registrado

Breve descripción: Esta funcionalidad permite que el usuario realice escaneos básicos de objetos sin necesidad de conexión a Internet. La aplicación utiliza un módulo de reconocimiento local para identificar el objeto y almacenar temporalmente los resultados.

#### **Precondiciones:**

- El dispositivo del usuario no cuenta con conexión a Internet (o se encuentra en modo avión/Offline).
- La aplicación debe tener previamente descargado y configurado un modelo de reconocimiento local.
- Existe un sistema de almacenamiento temporal local para guardar los registros de escaneo realizados en modo offline.
- El usuario puede haber iniciado sesión previamente (si se requiere asociar los datos a un

perfil) o, en modo invitado, se identifican de forma local hasta la sincronización.

#### **Detección del Modo Offline:**

- Al iniciar la aplicación o al intentar realizar un escaneo, el sistema detecta que no hay conexión a Internet.

#### **Activación del Módulo Offline:**

- La aplicación activa el módulo de reconocimiento local, el cual posee capacidades limitadas (por ejemplo, reconocimiento básico de materiales o categorías generales).

#### **Realización del Escaneo:**

- El usuario accede a la funcionalidad de escáner y toma una foto u obtiene una imagen del objeto a clasificar.
- El modelo local procesa la imagen y devuelve una clasificación preliminar (por ejemplo, "Plástico", "Vidrio", etc.) junto con posibles consejos básicos.

#### **Visualización Inmediata del Resultado:**

- El sistema muestra al usuario el resultado del escaneo e informa que se ha realizado en modo offline.
- Se puede incluir un mensaje como: "Escaneo completado. Los datos se sincronizan cuando se recupera la conexión."

#### **Almacenamiento Temporal:**

- El registro del escaneo (incluyendo la imagen, la clasificación y otros metadatos como fecha y hora) se almacena localmente en una base de datos temporal o en almacenamiento interno.

#### **Monitoreo de Conectividad y Sincronización:**

- La aplicación ejecuta un proceso en segundo plano o utiliza un servicio de sincronización que monitoriza el estado de la conexión a Internet.
- Una vez detectada la reconexión, el sistema inicia la sincronización de los datos almacenados localmente con el servidor central.

#### **Confirmación de Sincronización:**

- Tras la sincronización exitosa, el sistema actualiza el historial del usuario en el backend y notifica al usuario (por ejemplo, mediante un mensaje emergente) que sus escaneos offline han sido registrados.

#### **A1: Error en el Reconocimiento Local**

- El módulo de reconocimiento local falla al procesar la imagen o devuelve un resultado de baja confiabilidad.
  1. El sistema muestra un mensaje de error o advertencia, por ejemplo: "No se pudo reconocer el objeto. Por favor, inténtalo de nuevo o espera a tener conexión para un análisis más preciso."



2. Se ofrece la opción de reintentar el escaneo o almacenar el registro con la indicación de "resultado incierto", para que sea revisado posteriormente en modo online.

#### **A2: Fallo en la Sincronización**

- Al reconectar, el proceso de sincronización falla por problemas de servidor o errores de comunicación.
  1. El sistema reintentar la sincronización en intervalos regulares o notifica al usuario que algunos escaneos no pudieron sincronizarse.
  2. Se mantiene la información en el almacenamiento local hasta que la sincronización sea exitosa.

#### **A3: Cancelación o Edición de Escaneo Offline**

- El usuario decide eliminar o editar un registro antes de que se sincronice.
  1. Desde la sección "Historial Offline" o "Escaneos Pendientes", el usuario selecciona el registro a modificar.
  2. Se permite la edición (por ejemplo, corregir la clasificación) o la eliminación del registro.
  3. El sistema actualiza la base de datos local y, en el próximo ciclo de sincronización, los cambios se reflejarán en el servidor.

Historias de usuario

Escaneo de Objetos

#### **Descripción conceptual**

<b>Módulo Escaneo de Objetos</b>	<i>Ejemplo: Módulo de Gestión de Usuarios, Módulo de Reportes, etc.</i>
<b>Descripción de la(s) funcionalidad(es) requerida(s):</b>	<p><i>Este módulo se encarga de permitir a los usuarios capturar imágenes de objetos y enviarlas al servidor. Los usuarios pueden</i></p> <ul style="list-style-type: none"> <li>• <i>Tomar imágenes de los objetos.</i></li> <li>• <i>Recibir información procesada de la imagen por un modelo entrenado de ia.</i></li> <li>• <i>Determinar la categoría del objeto</i></li> <li>• <i>Recibir recomendaciones o consejos para los objetos</i></li> <li>• <i>Realizar un análisis básico del objeto</i></li> <li>• <i>Almacenar los datos del objeto en un historial.</i></li> </ul>

#### **Descripción técnica**

En este apartado existen dos partes, la del backend y la del frontend

Backend

Aquí se debe explicar de manera técnica lo esperado,

URL	Método	Código html
http://localhost:8080/scan	POST	200 OK 400 Bad Request
<p><b>Descripción</b></p> <p>Este endpoint recibe una imagen enviada y opcionalmente otros datos (como por ejemplo, un identificador de usuario en caso de usuario registrado). El servidor procesa la imagen utilizando módulos de visión por computadora y aprendizaje automático para clasificar el objeto y generar recomendaciones específicas.</p>		
<p><b>Datos de entrada</b></p> <p>POST /scan HTTP/1.1 Host: localhost:8080 Content-Type: multipart/form-data; boundary=----WebKitFormBoundary7MA4YWxkTrZu0gW -----WebKitFormBoundary7MA4YWxkTrZu0gW Content-Disposition: form-data; name="image"; filename="objeto.jpg" Content-Type: image/jpeg [contenido binario de la imagen] -----WebKitFormBoundary7MA4YWxkTrZu0gW Content-Disposition: form-data; name="userId" 123 -----WebKitFormBoundary7MA4YWxkTrZu0gW-</p>	<p><b>Datos de salida</b> <b>200 OK</b></p> <pre>{   "status": "success",   "data": {     "scanId": 45,     "processedAt": "2025-01-31, 14:30:06"   } }</pre>	
<p>POST /scan HTTP/1.1 Host: localhost:8080 Content-Type: multipart/form-data; boundary=----WebKitFormBoundary7MA4YWxkTrZu0gW -----WebKitFormBoundary7MA4YWxkTrZu0gW Content-Disposition: form-data; name="image"; filename="xxxxxxxxx" Content-Type: image/jpeg [contenido binario de la imagen] -----WebKitFormBoundary7MA4YWxkTrZu0gW Content-Disposition: form-data;</p>	<p><b>400 Bad Request</b></p> <pre>{   "status": "error",   "message": "Imagen no válida o   parámetros incompletos" }</pre>	

name="userId" 123 -----WebKitFormBoundary7MA4YWxkTrZu 0gW-	
---	--

<b>URL</b> http://localhost:8080/scan	<b>Método</b> GET	<b>Código html</b> 200 OK 400 Bad Request
<b>Descripción</b> Este endpoint permite consultar los detalles de un escaneo previamente realizado, devolviendo la información de clasificación y recomendaciones asociadas.		
<b>Datos de entrada</b>	<b>Datos de salida</b> 200 OK <pre>{   "status": "success",   "data": {     "scanId": 45,     "classification": "Plástico",     "recommendations": "Lavar el objeto y separarlo en la sección de plásticos",     "processedAt": "2025-01-31, 14:30:06",     "imageUrl":       "http://localhost:8080/images/scan_45.jpg"   } }</pre>	
	<b>400 Bad Request</b> <pre>{</pre>	

	<pre>"status": "error",    "message": "No se encontró un escaneo con el ID proporcionado"  }</pre>
--	--

### Frontend

#### Interacción esperada:

El usuario al entrar al escaneo de objeto se le abrirá un menú con los siguientes botones.

Botón “Escanear nuevo objeto” → Abre un formulario modal o activa la cámara para capturar una imagen del objeto.

Botón “Ver detalles” → Muestra un resumen detallado del escaneo, incluyendo la imagen, recomendaciones y fecha de procesamiento.

Botón “Reintentar análisis” (si el estado es “Error”) → Permite volver a enviar la imagen para procesamiento.

Botón “Subir imagen” → Permite al usuario subir una imagen de su galería.

#### Mockups/Prototipos:

1. Menú principal del scan
2. Cámara del scan
3. Detalles del scan. Cuando el usuario escanea correctamente y presiona el botón ver detalles
4. Mensaje de error de escaneo.

#### Flujo visual y eventos:



