**Documento de Requerimientos de Software**

Aplicación de Detección y Valorización de Residuos de aparatos electricos y electronicos (EcoElectro)

***Fecha: 14/09/2025***

**Tabla de contenido**

[**2. Alcance del Producto / Software 5**](#_heading=)

[**Propósito general del software: 5**](#_heading=)

[**Beneficios: 5**](#_heading=)

[**Objetivos y Metas: 5**](#_heading=)

[**3. Referencias 6**](#_heading=)

[**4. Funcionalidades del Producto 7**](#_heading=)

[**5. Clases y Características de Usuarios 8**](#_heading=)

[**6. Entorno operativo 9**](#_heading=)

[**7. Requerimientos funcionales 10**](#_heading=)

[**7.1. Autenticación y Gestión de Usuarios 10**](#_heading=)

[**7.2. Captura e Identificación por IA 11**](#_heading=)

[**7.3. Clasificación y Guía de Componentes 11**](#_heading=)

[**7.4. Mapa y Logística de Reciclaje 12**](#_heading=)

[**7.5. Valorización y Mercado Secundario 12**](#_heading=)

[**7.6. Gamificación 13**](#_heading=)

[**7.7. Administración (Backoffice) 14**](#_heading=)

[**8. Reglas de Negocio 15**](#_heading=)

[**9. Requerimientos de Interfaces Externas 16**](#_heading=)

[**9.1. Interfaces de Usuario 16**](#_heading=)

[**9.2. Interfaces de Hardware 16**](#_heading=)

[**9.3. Interfaces de Software 16**](#_heading=)

[**9.4. Interfaces de comunicación 17**](#_heading=)

[**10. Requerimientos No Funcionales 18**](#_heading=)

[**Usabilidad y Accesibilidad: 18**](#_heading=)

[**Rendimiento: 18**](#_heading=)

[**Disponibilidad y Compatibilidad: 18**](#_heading=)

[**Seguridad: 19**](#_heading=)

[**Mantenibilidad: 19**](#_heading=)

[**11. Otros Requerimientos 19**](#_heading=)

[**12. Glosario 20**](#_heading=)

Historial de Versiones

| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| --- | --- | --- | --- | --- |
| 13/09/2025 | 1.0 | Equipo de Desarrollo | DUOC UC | Versión inicial completa y corregida. Integración de requerimientos funcionales y no funcionales. Corrección de stack tecnológico a .NET MAUI. |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Información del Proyecto

| Empresa / Organización | DUOC UC |
| --- | --- |
| Proyecto | Aplicación de Detección y Valorización de Residuos de aparatos electricos y electronicos (RAEE) |
| Fecha de preparación | 11/09/2025 |
| Cliente | Arturo Vargas |
| Patrocinador principal | Ignacia Arancibia |
| Gerente / Líder de Proyecto | Gabriela Gomez |
| Gerente / Líder de Análisis de negocio y requerimientos | Vania Troncoso |

Aprobaciones

| **Nombre y Apellido** | **Cargo** | **Departamento u Organización** | **Fecha** | **Firma** |
| --- | --- | --- | --- | --- |
| Arturo Vargas | Profesor a cargo | Informática |  |  |
| Gerardo Galán | Auditor | Duoc UC |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

1. **Propósito**

El software para desarrollar se denomina EcoElectro – Aplicación de Detección y Valorización de Residuos de Aparatos Eléctricos y Electrónicos(RAEE), en su versión 1.0 (Release inicial – MVP)

El propósito de este documento es especificar los requerimientos funcionales y no funcionales que conforman el sistema, asegurando que el desarrollo del producto se realice de manera ordenada y consistente con los objetivos del proyecto.

El documento cubre la totalidad del software en su primera versión, que incluye los siguientes componentes:

* Módulo de autenticación y gestión de usuarios.
* Módulo de captura de imágenes y detección de dispositivos electrónicos mediante IA.
* Módulo de identificación y clasificación de componentes (reciclables, no reciclables, reutilizables, valorizables).
* Módulo de mapa de puntos de reciclaje y opciones de retiro a domicilio.
* Módulo de valorización con enlaces a plataformas de compraventa.
* Módulo de gamificación con sistema de puntos y recompensas.
* Panel de administración para gestión de usuarios, puntos de reciclaje, empresas de retiro y contenidos educativos.

Este documento establece los lineamientos para el desarrollo, pruebas, validación y mantenimiento inicial del software, garantizando trazabilidad entre requerimientos y funcionalidades entregadas.

# 2. Alcance del Producto / Software

## Propósito general del software:

EcoElectro tiene como objetivo ofrecer a los consumidores una aplicación móvil intuitiva que permita detectar dispositivos electrónicos en desuso, identificar sus componentes, orientar al usuario sobre el reciclaje adecuado y facilitar la valorización de piezas reutilizables, fomentando así la economía circular.

## Beneficios:

* Ambientales: disminución del impacto negativo de los RAEE en vertederos.
* Sociales: concientización de los usuarios sobre el reciclaje responsable.
* Económicos: valorización de componentes reutilizables que aún tienen demanda.
* Normativos: apoyo al cumplimiento de la Ley Responsabilidad Extendida del Productor (REP) (20.920), en Chile.
* Tecnológicos: integración de inteligencia artificial en una solución de uso cotidiano.

## Objetivos y Metas:

* Desarrollar una aplicación móvil multiplataforma (Android como prioridad, usando .NET MAUI).
* Implementar un modelo de visión computacional y redes neuronales capaz de identificar dispositivos electrónicos comunes.
* Clasificar automáticamente los componentes detectados en categorías de reciclaje, reutilización o venta.
* Proporcionar un mapa interactivo de puntos de reciclaje
* Proporcionar listado de empresas de retiro a domicilio.
* Establecer un sistema de recompensas para incentivar la participación de los usuarios.
* Lograr un MVP en 10 semanas, validado mediante pruebas de usabilidad y QA.

Este alcance se encuentra alineado con los objetivos de sostenibilidad, innovación tecnológica y responsabilidad social, promovidos tanto por la industria tecnológica como por las políticas medioambientales en Chile.

# 3. Referencias

A continuación, se listan los documentos y fuentes de información utilizadas como referencia para la elaboración de este documento de requerimientos:

1. Ley REP (20.920) – Responsabilidad Extendida del Productor. Gobierno de Chile, Ministerio del Medio Ambiente. Santiago, 2016. Disponible en: https://mma.gob.cl/ley-rep
2. Manual de Especificación de Requerimientos de Software. IEEE Std 830-1998, IEEE Computer Society, 1998.
3. Documentación de .NET MAUI. Microsoft. Última versión consultada en 2025.
4. Documentación de TensorFlow Lite. Google Developers. Última versión consultada en 2025.
5. Documentación de Google Maps Platform y OpenStreetMap. Consultado en 2025.
6. Documentación interna del Proyecto APT – EcoElectro. Versión 1.0. Equipo de Desarrollo DUOC UC, 2025.

# 4. Funcionalidades del Producto

Las principales funcionalidades del software son:

1. **Gestión de Usuarios:** Registro, Inicio de Sesión, Recuperación de Contraseña y Perfil Básico (Nombre, Comuna).
2. **Captura e Identificación de Dispositivos:** Uso de la cámara para tomar fotografías y Aplicar el Modelo de IA para detectar el dispositivo.
3. **Clasificación de Componentes:** Listado automático de piezas del dispositivo, indicando si son reciclables, no reciclables, reutilizables o valorizables.
4. **Mensajes de Error:** Notificación al usuario en caso de que no se logre identificar el dispositivo (“Por favor tomar foto otra vez” o “Dispositivo no encontrado”).
5. **Mapa de Puntos de Reciclaje:** Visualización de centros de reciclaje cercanos con filtros por tipo de componente y comuna.
6. **Servicio de Recolección:** Información de empresas que ofrecen retiro a domicilio, con detalles de cobertura y contacto.
7. **Valorización Económica:** Enlace directo a plataformas de compraventa de componentes reutilizables (ej. Mercadolibre, Yapo).
8. **Gamificación:** Sistema de puntos y niveles (Bronce, Plata, Oro, Platino) para incentivar el reciclaje y mostrar historial de logros.
9. **Educación al Usuario:** Información sobre qué se hace con los componentes reciclados y consejos de reutilización.
10. **Administración (BackOffice):** Gestión de usuarios, puntos de reciclaje, empresas de retiro, componentes y reglas de gamificación.
11. **Analítica Básica:** Métricas sobre detecciones, uso de la app, puntos registrados y componentes más comunes.

# 5. Clases y Características de Usuarios

Se definen dos tipos principales de usuarios:

1. **Usuario consumidor (Final):**

* Frecuencia de Uso: Ocasional a frecuente (cuando recicla o detecta dispositivos).
* Funcionalidades Relevantes: Registro, Inicio de Sesión, Captura de Dispositivo, Clasificación de Componentes, Mapa de Reciclaje, Servicios de Retiro, Valorización, Recompensas y Contenido Educativo.
* Características: Usuario común, sin conocimientos técnicos, requiere interfaz intuitiva y mensajes claros.

1. **Administrador del Sistema:**

* Frecuencia de Uso: Frecuente, ya que gestiona y actualiza información.
* Funcionalidades Relevantes: Gestión de usuarios, puntos de reciclaje, empresas de retiro, catálogo de componentes, reglas de gamificación y visualización de métricas.
* Características: Usuario técnico con permisos especiales, nivel de experiencia medio-alto, responsable de mantener la base de datos actualizada.

# 6. Entorno operativo

El sistema se ejecutará en el siguiente entorno:

1. **Hardware requerido:**

* Dispositivos móviles (smartphones con cámara y GPS).
* Resolución mínima de pantalla: 720x1280 px.

1. **Software y Plataformas:**

* Aplicación móvil: Desarrollada en .NET MAUI usando Visual Studio 2022 o superior.
* Lenguaje de programación: C# y XAML.
* Sistema operativo: Android 10 o superior (MVP). iOS se considera para fases futuras.
* Backend: ASP.NET Core Web API o Firebase (servicios en la nube).
* **Base de datos:** Se utilizará **SQL Server en Azure** como sistema de gestión de base de datos en la nube, lo que permitirá una estructura relacional robusta, mayor trazabilidad, control de acceso y compatibilidad con herramientas de auditoría y documentación técnica. Para el almacenamiento local en la aplicación móvil, se mantendrá **SQLite** como solución ligera y eficiente para operaciones offline, con mecanismos de sincronización hacia la nube cuando sea necesario.
* Modelo de IA: TensorFlow Lite integrado en .NET MAUI para detección de dispositivos.

1. **Integraciones Externas:**

* Google Maps API / OpenStreetMap para geolocalización.
* Enlaces externos a plataformas de compraventa (Mercadolibre, Yapo).

1. **Entorno de red:**

* Conexión a internet 4G o WiFi.
* HTTPS/TLS 1.2+ para seguridad de datos.

1. **Otros componentes:**

* Compatibilidad con librerías de accesibilidad (ej. lectores de pantalla).
* Logs de auditoría para administrador.

# 7. Requerimientos funcionales

## 

### 7.1 Autenticación y Gestión de Usuarios

### Descripción: Permite a los usuarios registrarse, iniciar sesión, recuperar contraseña y gestionar su perfil básico. Prioridad: Alta

## Requerimientos:

## RF-01: El sistema debe permitir el registro de un nuevo usuario utilizando correo electrónico y contraseña, validando formato y unicidad. *(CU2)*

## RF-02: El sistema debe permitir el inicio de sesión con correo electrónico y contraseña, manteniendo una sesión persistente y segura mediante token JWT. *(CU1)*

## RF-03: El sistema debe proporcionar recuperación de contraseña mediante correo electrónico registrado, con validación de identidad. *(CU3)*

## RF-04: El sistema debe permitir al usuario crear y editar su perfil básico, incluyendo nombre y comuna de residencia, con validación de campos obligatorios. *(CU4)*

### 7.2 Captura e Identificación por IA

### Descripción: Usa la cámara del dispositivo para capturar imágenes de RAEE y los identifica mediante un modelo de IA. Prioridad: Alta

## Requerimientos:

## RF-05: La aplicación debe incluir un botón dedicado para acceder a la cámara y tomar una fotografía del dispositivo. *(CU5)*

## RF-06: El sistema debe procesar la imagen mediante una red neuronal entrenada para detectar el tipo de dispositivo electrónico (ej. teléfono, notebook, monitor). *(CU6)*

## RF-07: Si el dispositivo no es reconocido, el sistema debe mostrar el mensaje: “Dispositivo no encontrado. Por favor, intente con otra foto”, visible durante al menos 3 segundos. *(CU6)*

## RF-08: Tras una detección exitosa, el sistema debe listar los componentes principales identificados (pantalla, batería, placa madre, etc.). *(CU6)*

### *7.3 Clasificación y Guía de Componentes*

## *Descripción: Clasifica cada componente detectado por la IA y proporciona al usuario información clara sobre su disposición final. Prioridad: Alta*

## *Requerimientos:*

## *RF-09: Para cada componente listado, el sistema debe indicar su estado: Reciclable, No Reciclable, Reutilizable o Vendible, utilizando íconos o colores diferenciadores para facilitar la comprensión. (CU7)*

## *RF-10: El sistema debe mostrar una explicación textual por componente, sugiriendo acciones concretas (ej. “Esta pantalla puede reciclarse en puntos especializados”). (CU7)*

## *RF-11: El sistema debe incluir fichas educativas que expliquen el destino de los componentes reciclados y no aprovechables, accesibles desde la interfaz del usuario. (CU8)*

### 7.4 Mapa y Logística de Reciclaje

## Descripción: Proporciona herramientas basadas en ubicación para encontrar puntos de reciclaje y servicios de retiro. Prioridad: Alta

## Requerimientos:

## RF-12: El sistema debe mostrar un mapa interactivo con puntos de reciclaje cercanos a la comuna registrada en el perfil del usuario. *(CU9)*

## RF-13: Cada punto en el mapa debe mostrar dirección, horario, tipos de RAEE aceptados e información de contacto. *(CU9)*

## RF-14: El sistema debe listar empresas que ofrecen retiro a domicilio, mostrando cobertura geográfica, costos (si aplica) y canal de contacto. *(CU10, CU12)*

### 7.5 Valorización y Mercado Secundario

## Descripción: Facilita la venta de componentes reutilizables mediante enlaces a plataformas de compraventa. Prioridad: Media

## Requerimientos:

## RF-15: Para componentes marcados como “Reutilizable” o “Vendible”, el sistema debe generar un enlace directo a una búsqueda preconfigurada en plataformas como Mercado Libre o Yapo, usando nombre y modelo. *(CU13, CU17)*

### 7.6 Gamificación

## Descripción: Incentiva la participación mediante puntos, niveles y recompensas. Prioridad: Baja (para MVP, pero esencial para engagement)

## Requerimientos:

## RF-16: El sistema debe asignar puntos al usuario por acciones como: subir diagnóstico correcto, registrar entrega en punto de reciclaje, solicitar retiro o marcar venta como realizada. *(CU12)*

## RF-17: El sistema debe organizar a los usuarios en niveles (Bronce, Plata, Oro, Platino) según puntos acumulados, mostrando requisitos y beneficios. *(CU12)*

## RF-18: El sistema debe mostrar al usuario un historial de puntos acumulados y acciones realizadas. *(CU11)*

### 7.7 Administración (Backoffice)

## Descripción: Herramientas para que el administrador gestione la información del sistema. Prioridad: Media-Alta

## Requerimientos:

## RF-19: El sistema debe proporcionar una interfaz segura para gestionar (CRUD) cuentas de usuario. *(CU14)*

## RF-20: El sistema debe permitir la gestión (CRUD) de puntos de reciclaje, incluyendo nombre, dirección, geolocalización, tipos de RAEE y horarios. *(CU15)*

## RF-21: El sistema debe permitir la gestión (CRUD) de empresas de retiro, incluyendo cobertura geográfica y condiciones de servicio. *(CU16)*

## RF-22: El sistema debe permitir la gestión (CRUD) de un catálogo de componentes, con nombres, imágenes y reglas de clasificación. *(CU17)*

## RF-23: El sistema debe permitir la gestión (CRUD) de contenidos educativos y textos explicativos. *(CU15)*

## RF-24: El sistema debe permitir configurar reglas de gamificación (puntos por acción, umbrales de niveles). *(CU12)*

# 8. Reglas de Negocio

1. Solo los usuarios registrados y con sesión activa pueden acumular puntos en el sistema de gamificación.
2. Los enlaces a plataformas de compraventa deben abrirse en el navegador web predeterminado del dispositivo, fuera de la aplicación.
3. La visualización de puntos de reciclaje en el mapa debe priorizar aquellos ubicados en la comuna del usuario, si esta está registrada.
4. Un componente sólo puede ser clasificado en una de las siguientes categorías: Reciclable, No Reciclable, Reutilizable o Vendible.
5. El acceso al panel de administración está restringido exclusivamente a usuarios con el rol "Administrador".
6. Las imágenes capturadas por la cámara no deben almacenarse permanentemente en los servidores sin el consentimiento explícito del usuario.
7. Los dispositivos que la aplicación detecta están acotados a solo 6 categorías, de las cuales se excluye la detección de baterías.

# 9. Requerimientos de Interfaces Externas

## 9.1. Interfaces de Usuario

1. La aplicación debe presentar una interfaz de Usuario (UI) nativa para Android, desarrollada con .NET MAUI.
2. El diseño debe seguir los principios de Material Design.
3. Debe existir una navegación principal basada en pestañas (tabs) o un menú lateral (drawer) para acceder a las secciones: Cámara, Mapa, Perfil, Recompensas.
4. Todas las pantallas deben mantener una coherencia visual en cuanto a colores, tipografía y tamaño de botones (mínimo 44px para áreas táctiles).

## 9.2. Interfaces de Hardware

1. La aplicación debe ser capaz de interactuar con la cámara posterior de los dispositivos Android.
2. La aplicación debe acceder al GPS del dispositivo para obtener la ubicación del usuario y mostrar puntos de reciclaje cercanos.

## 9.3. Interfaces de Software

1. **Modelo de IA:** La aplicación integrará un modelo de TensorFlow Lite empaquetado localmente dentro de la aplicación para la detección de dispositivos.
2. **Mapas:** La aplicación utilizará la API de Google Maps para .NET MAUI o una integración similar para OpenStreetMap para renderizar el mapa y los puntos de reciclaje.
3. **Backend:** La aplicación se comunicará con un backend basado en una API REST, desarrollado en **ASP.NET Core** y desplegado en **Microsoft Azure**, para gestionar operaciones que requieran persistencia de datos como autenticación, sistema de puntos y métricas analíticas. Esta arquitectura permitirá una integración directa con **Azure SQL Database**, servicios de identidad y herramientas de monitoreo para asegurar escalabilidad, seguridad y trazabilidad.

## 9.4. Interfaces de comunicación

1. Todas las comunicaciones con servicios externos (backend, APIs de mapas) deben realizarse mediante el protocolo HTTPS utilizando TLS 1.2 o superior.
2. El formato de intercambio de datos para las APIs REST será JSON.
3. Para la autenticación, se utilizarán tokens JWT (JSON Web Tokens) que se enviarán en el header `Authorization` de las peticiones HTTP.

# 10. Requerimientos No Funcionales

### Usabilidad y Accesibilidad

* **RNF-01:** El flujo principal de la aplicación (Capturar → Ver resultados → Ver opciones de reciclaje o venta) debe completarse en un máximo de tres pantallas consecutivas, sin navegación adicional ni pantallas emergentes innecesarias.
* **RNF-02:** Todos los textos, mensajes y botones deben estar redactados en español de Chile, utilizando lenguaje claro, inclusivo y sin tecnicismos innecesarios.
* **RNF-03:** La aplicación debe cumplir con pautas básicas de accesibilidad, incluyendo contraste de color adecuado, tamaño de fuente legible y compatibilidad con lectores de pantalla como TalkBack.

### Rendimiento

* **RNF-04:** El tiempo de inferencia del modelo de IA (desde la captura de imagen hasta la entrega del resultado) debe ser igual o inferior a 3 segundos en dispositivos Android de gama media con conexión 4G estable.
* **RNF-05:** El mapa interactivo debe cargar y mostrar los pines de puntos de reciclaje en menos de 2 segundos después de obtener la ubicación del usuario.
* **RNF-06:** La aplicación debe almacenar en caché local la lista de puntos de reciclaje, permitiendo funcionamiento ágil y resiliente ante conexiones intermitentes, con actualización periódica en segundo plano.

### Disponibilidad y Compatibilidad

* **RNF-07:** Los servicios backend críticos (autenticación, detección de dispositivos) deben garantizar una disponibilidad mensual mínima del 99%, con monitoreo activo mediante herramientas como Azure Application Insights.
* **RNF-08:** La aplicación móvil debe ser compatible con dispositivos Android versión 10.0 (API nivel 29) y superiores, asegurando funcionamiento estable en resoluciones mínimas de 720p.

### Seguridad

* **RNF-09:** Las credenciales de usuario y los tokens de autenticación deben almacenarse de forma segura utilizando las APIs de Secure Storage provistas por .NET MAUI, sin acceso por otras aplicaciones ni almacenamiento en texto plano.
* **RNF-10:** Todos los datos personales deben cifrarse en tránsito mediante protocolos HTTPS con soporte para TLS 1.2 o superior, validando el certificado del servidor.
* **RNF-11:** El acceso a los endpoints del API de administración debe estar restringido por roles, permitiendo únicamente el ingreso de usuarios con perfil de administrador validado.

### Mantenibilidad

* **RNF-12:** La aplicación debe implementar un sistema de logging que registre eventos críticos del backend, incluyendo fallos de autenticación, errores en el proceso de inferencia y actividad administrativa, con timestamp, tipo de evento y usuario afectado.
* **RNF-13:** El modelo de IA debe poder ser actualizado y reemplazado en el servidor sin requerir una nueva versión de la aplicación móvil, mediante descarga desde una CDN o configuración remota controlada por el backend.

# 11. Otros Requerimientos

* **Base de datos:** El sistema de base de datos debe ser diseñado para soportar de forma estructurada las funcionalidades de **gamificación**, **gestión de puntos de reciclaje** y **analítica de uso**, garantizando trazabilidad, rendimiento y escalabilidad. Para el almacenamiento local en el dispositivo móvil se utilizará **SQLite**, mientras que para la persistencia en la nube se empleará **Azure SQL Database**, permitiendo integración con servicios backend desarrollados en ASP.NET Core y compatibilidad con herramientas de monitoreo y auditoría.

1. **Internacionalización:**

* Para el MVP, la aplicación solo estará disponible en español de Chile.

1. **Legales:**

* La aplicación debe incluir una Política de Privacidad y Términos de Uso accesibles dentro de la aplicación, los cuales deben detallar el manejo de datos personales y de imágenes.

1. Reutilización de componentes:

* Se priorizará el uso de librerías y componentes estables de la comunidad .NET MAUI para agilizar el desarrollo y garantizar la mantenibilidad.

# 12. Glosario

1. **RAEE:** Residuos de Aparatos Eléctricos y Electrónicos. Desechos provenientes de dispositivos tecnológicos.
2. **Ley REP (20.920):** Ley de Responsabilidad Extendida del Productor. Establece un marco para la gestión de residuos en Chile, impulsando la economía circular.
3. **MVP (Minimum Viable Product):** Producto Mínimo Viable. Versión inicial de una aplicación con las funcionalidades básicas necesarias para ser validada por los usuarios.
4. **IA (Inteligencia Artificial):** Simulación de procesos de inteligencia humana por máquinas, en este caso, para el reconocimiento de imágenes.
5. **.NET MAUI:** .NET Multi-platform App UI. Framework de Microsoft para crear aplicaciones nativas para dispositivos móviles y de escritorio con C#.
6. **TensorFlow Lite:** Versión ligera de la librería de machine learning TensorFlow, diseñada para ejecutarse en dispositivos móviles y embebidos.
7. **Backend:** Conjunto de servicios y lógica de servidor que dan soporte a las funcionalidades de la aplicación móvil.
8. **CRUD:** Acrónimo de Create (Crear), Read (Leer), Update (Modificar), Delete (Eliminar). Operaciones básicas de persistencia de datos.
9. **JWT (JSON Web Token):** Estándar abierto para la creación de tokens de acceso que permiten transmitir información de forma segura entre partes.
10. **API (Application Programming Interface):** Interfaz que permite la comunicación e integración entre diferentes componentes de software.