

# PokeStorage - Documento de Requisitos de Software (SRD)

---

## 1. Información General

- **Proyecto:** PokeStorage
  - **Versión:** 1.0
  - **Fecha:** 2025-11-04
  - **Autores:** Equipo PokeStorage
  - **Metodología:** Modelo Tradicional Basado en Prototipos
  - **Herramienta de Gestión:** Jira Software
  - **Repositorio:** [PokeStorage-Agiles](#)
- 

## 2. Introducción

### 2.1 Propósito

El propósito de este documento es especificar, detallar y formalizar todos los requerimientos funcionales y no funcionales del sistema PokeStorage, un software educativo y recreativo que permite gestionar Pokémon capturados, visualizarlos, organizarlos y almacenarlos usando estructuras de datos (listas, pilas y árboles).

Este documento sirve como base para el diseño, desarrollo, evaluación y documentación final del prototipo bajo la metodología tradicional basada en prototipos, gestionada en Jira Software.

### 2.2 Alcance

El sistema PokeStorage permitirá:

- Registrar usuarios con autenticación básica.
- Iniciar sesión y gestionar una colección personal de Pokémon.
- Agregar, visualizar, eliminar y recorrer Pokémon mediante una lista circular.
- Guardar Pokémon en estructuras tipo Pila y Árbol AVL.
- Visualizar un historial de capturas.
- Mostrar los Pokémon con su imagen, nombre, tipo y poder.

El prototipo actual se desarrollará con Java (Swing) y persistencia local en archivos .txt o SQLite, simulando el entorno de un entrenador Pokémon que gestiona su almacenamiento.

### 2.3 Definiciones

- **Prototipo:** versión preliminar funcional que evoluciona según la retroalimentación del usuario.
  - **Jira:** herramienta de gestión de tareas y documentación utilizada para seguimiento de fases.
  - **Actor:** entidad (usuario o sistema) que interactúa con el software.
- 

## 3. Descripción general

### 3.1 Perspectiva del sistema

El sistema es una aplicación de escritorio desarrollada en Java, compuesta por:

- **Interfaz gráfica (GUI):** ventanas y botones para registrar, visualizar y recorrer Pokémon.
- **Módulo lógico:** estructuras de datos (lista circular, pila, árbol AVL).
- **Persistencia:** archivos locales o base de datos SQLite.

### 3.2 Usuarios

- **Entrenador Pokémon (usuario principal):** gestiona su colección personal.

### 3.3 Suposiciones y dependencias

- Java instalado (versión 11 o superior).
- Librerías Swing disponibles.
- Los datos se almacenan de forma local.

## 4. Requisitos funcionales

ID	Nombre	Descripción	Prioridad	Jira Issue
RF01	Registro de usuario	El sistema permitirá crear nuevos usuarios con nombre, edad y contraseña, generando un archivo local.	Alta	GM-3
RF02	Inicio de sesión	El sistema validará usuarios registrados mediante lectura de archivo o BD.	Alta	GM-13
RF03	Registro de Pokémon	Permite añadir Pokémon (nombre, tipo, poder, imagen).	Alta	GM-18
RF04	Visualización de Pokémon	Muestra los Pokémon registrados en una lista circular, con botones Siguiente y Anterior.	Alta	GM-37
RF05	Historial de capturas	Muestra un historial de Pokémon capturados.	Media	GM-42
RF06	Inserción y recorrido	Permite recorrer o insertar Pokémon en la lista circular según su posición.	Media	GM-39
RF07	Visualización con árbol AVL	Permite clasificar Pokémon por poder utilizando un árbol AVL.	Media	GM-45
RF08	Interfaz visual con imágenes	La interfaz mostrará imágenes, botones y mensajes de error o confirmación.	Media	GM-53

## 5. Requisitos no funcionales

ID	Categoría	Descripción	Prioridad
----	-----------	-------------	-----------

ID	Categoría	Descripción	Prioridad
RNF01	Usabilidad	La interfaz debe ser intuitiva y visualmente atractiva (colores, botones, tipografía).	Alta
RNF02	Rendimiento	El sistema debe responder en menos de 2 segundos por acción.	Media
RNF03	Compatibilidad	El software debe ejecutarse en cualquier equipo con Java 11+.	Alta
RNF04	Persistencia	Los datos deben guardarse automáticamente tras cada cambio.	Alta

## 6. Actores y casos de uso

Actor	Descripción	Casos de uso asociados
<b>Usuario (Entrenador)</b>	Persona que usa la aplicación para almacenar y ver sus Pokémon.	CU01–CU06
<b>Sistema</b>	Software que responde a acciones del usuario.	CU01–CU07

### 6.1 Casos de uso principales

ID	Nombre	Descripción breve	RF asociado
CU01	Registrar usuario	Permite crear un nuevo perfil.	RF01
CU02	Iniciar sesión	Valida las credenciales.	RF02
CU03	Registrar Pokémon	Añade un Pokémon a la lista circular.	RF03
CU04	Ver lista circular	Muestra el Pokémon actual.	RF04
CU05	Eliminar Pokémon	Remueve el Pokémon actual de la lista.	RF04
CU06	Visualizar historial	Muestra Pokémon eliminados o registrados.	RF05
CU07	Visualizar árbol AVL	Muestra jerarquía por poder.	RF07