

Córdoba, Argentina

Universidad Blas Pascal

Cursado: 2020

Entrega: 21 de Febrero de 2022

# Trabajo Final

## PokePedia

### **Objetivo:**

Recolectar información sobre pokemones principalmente utilizando una API y realizar un seguimiento de las batallas que ocurren entre los mismos.

### **Integrantes**

- Franco Geremia
- Facundo Facello

### **Materia**

- Ingeniería informática: Laboratorio II

### **Profesor**

- Gustavo Funes

# Proyecto:

# PokePedia

## **Descripción:**

Se trata de un sistema basado en Django que le permite al usuario buscar información sobre pokemones (la cual es guardada en una base de datos) a través de un sitio web que implementa una API, utilizando el Admin de Django se pueden gestionar todos los datos obtenidos además de darle la posibilidad al usuario de que registre un pokemon propio.

Nuestro proyecto también permite registrar y gestionar batallas que ocurren entre dichos pokemones.

El sistema administra los siguientes tipos de datos:

- **Pokemon:** Ser similar a un animal o bestia mitológica que participara de batallas.
- **Batalla:** Registro de una pelea que ocurrió entre 2 o más pokemones.
- **Lugar:** Localización en donde se llevarán a cabo batallas pokemon.
- **Tipo:** Característica natural que define en gran medida a un pokemon.

## **Requerimientos Funcionales:**

El sistema deberá permitirle al usuario realizar lo siguiente:

- Búsqueda de un pokemon en base a su nombre o número asociado.
- Registro automático de un pokemon cuando este es buscado a través de la API.
- Gestión general (alta, baja, modificación y búsqueda) de todos los datos manejados.
- Seguimiento de batallas que ocurren entre pokemones.

## **Supuestos:**

- Toda la información será almacenada de forma local.
- No se traducirá el idioma de la información obtenida a través de la API.

## **Restricciones:**

- Las batallas no tendrán más de un ganador.
- Las batallas pueden llegar a ser eliminadas si se borra un pokemon del registro.

## **Infraestructura:**

- **Servidor:** Django cuenta con un servidor que funciona de manera local que nos permite visualizar nuestro proyecto, decidimos utilizar este mismo.

- **Base de datos:** Django crea automáticamente una base de datos SQLite en el momento que el proyecto lo requiere, por lo que nos manejamos con esta misma, dentro del proyecto definimos las tablas que va a tener esta base de datos utilizando código Python, su contenido lo podemos gestionar a través del admin de Django y externo al sistema también usamos DBeaver, que es una herramienta especializada para administrar bases de datos.
- **Lenguajes:** Para el **Front-end** empleamos HTML, CSS y JavaScript, además de las herramientas que nos proporcionaron las bibliotecas Bootstrap y JQuery.  
En cuanto al **Back-end** se trata únicamente de Python.
- **FrameWork:** Como fue mencionado previamente, la base de nuestro proyecto es **Django**, un framework de desarrollo web que nos facilita en gran parte el trabajo sobre nuestro sistema.

## **Repositorios:**

- <https://github.com/Masterjkk7/Trabajos-Practicos-LAB-III.git>
- <https://github.com/Le0-h/TPS.git>