

UNIVERSIDAD EUROPEA DEL ATLÁNTICO

GRADO EN Ingeniería Informática



SISTEMAS DISTRIBUIDOS Y PROGRAMACIÓN EN PARALELO

PROYECTO FINAL POKECATCHER

Trabajo realizado por

MANUEL ORTEGA PUYOL

ANDRÉS EMILIO REQUENA GONZÁLEZ

Profesor

Daniel Iglesias Santamaria

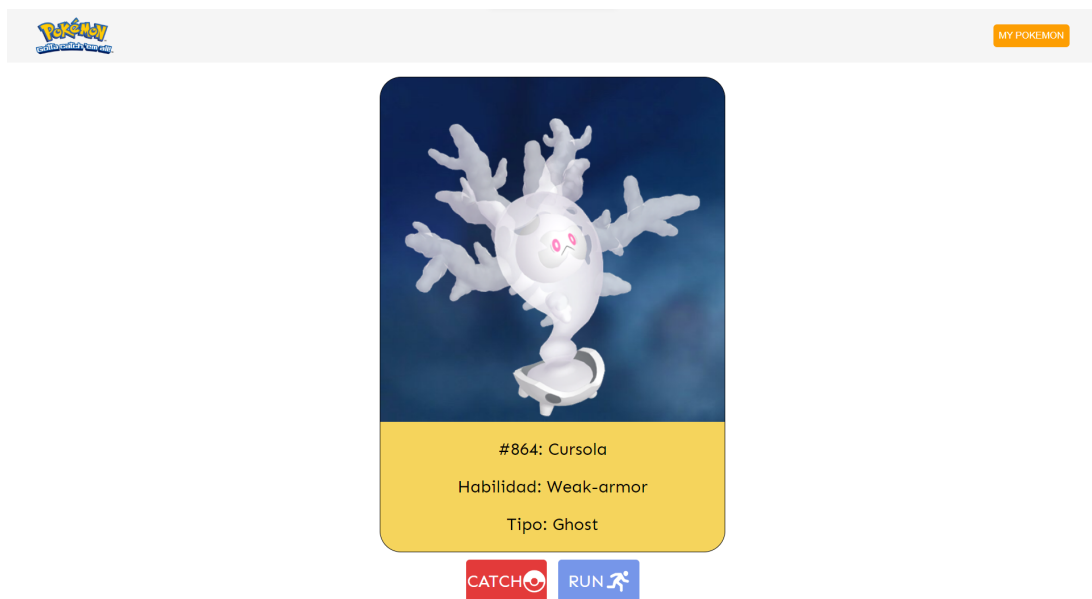
SANTANDER – 13, 12, 2022

El programa consistirá de tres elementos:

1. Frontal Web:

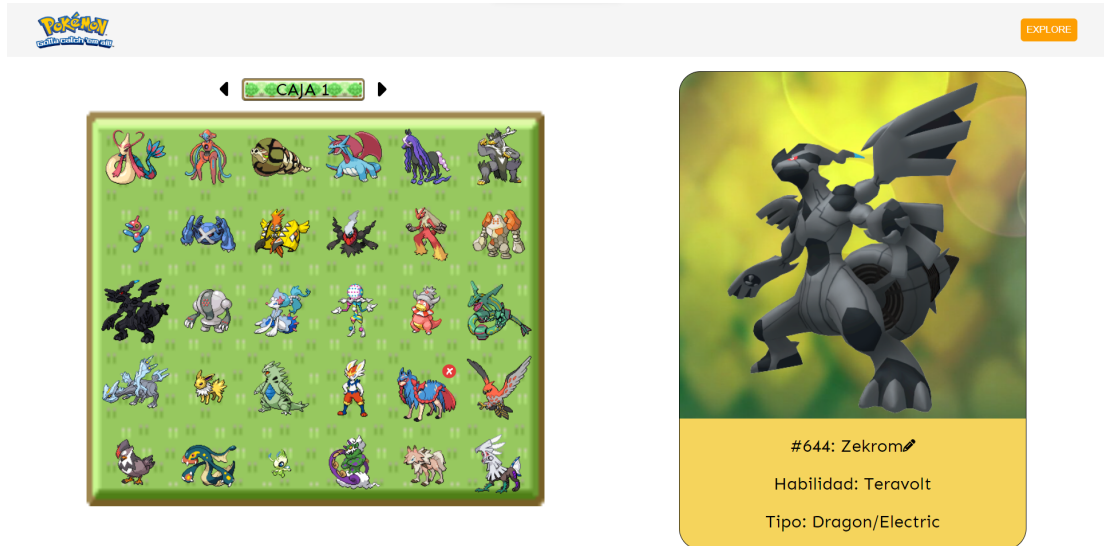
Consiste de 2 vistas, en la primera vista se hace una llamada GET a nuestra Web API, ésta devolverá un response que contendrá los datos de un pokemon generado aleatoriamente. El usuario puede entonces decidir si desea “capturar” a ese pokemon o si desea “huir”.

En caso que el usuario decida capturar al pokemon, la información de este será enviada a nuestra Web API mediante una llamada POST. En caso contrario, se realizará otra llamada GET para conseguir a un nuevo pokémon generado aleatoriamente.



La segunda vista consiste en una “caja” en la que se pueden ver pokémon que hayan sido atrapados (hasta 30 por página), esta vista contiene también una carta donde -al clicar uno de los pokémon atrapados- se podrá ver la información del pokémon.

En esta vista también se pueden "liberar" pokemon dándole doble click al botón X que aparece al posar el ratón sobre alguno de los pokémon. Los pokémon liberados serán eliminados de la base de datos mediante una llamada DELETE que es enviada a la Web API.



También se pueden poner "motes" a los pokémon dando click a al lápiz junto a su nombre, al hacer click el botón cambiará a tener un icono de "guardar" que usaremos al terminar de escribir el nuevo nombre de nuestro pokémon



2. Web API:

Consiste de un proyecto Web API con un controlador que se encarga de recibir las peticiones enviadas desde el frontal y realizar acciones con los datos recibidos.

Las peticiones que admite nuestra Web API son:

- **Get()**: Devuelve un pokémon random.
- **Get(int page)**: Devuelve un arreglo de pokémon que han sido capturados paginados de 30 en 30.
- **Post([FromBody] Pokemon pokemon)**: Recibe un pokémon que ha sido capturado y lo guarda en la base de datos.
- **Put(string "nickname", string "id")**: Recibe el id de un pokémon y un mote que le será asignado y actualiza el documento en la base de datos.
- **Delete(string "id")**: Recibe el id de un pokémon almacenado en la base de datos y lo elimina de esta.

El controlador se apoya también en un proyecto que contiene toda la lógica de negocio. Este proyecto cuenta con toda la logística necesaria para realizar las operaciones anteriormente descritas, además de realizar la conexión con la base de datos.

3. Base de datos:

Consiste en una base de datos no relacional MongoDB Atlas. Esta base de datos, llamada "pokecatcher" cuenta con una colección de documentos llamada "box" donde se almacenan los datos de los pokémon atrapados.

