

# Standard Bank Case Study

Name: Muhammad Omar

Date: 05/10/2025

Case study title: PokeAPI coding project 5.

Language used: C#, html and css

Framework: .NET Blazor webassembly

Document purpose:

This document outlines how to install and view the project as well as the purpose of the project and how to use the project.

# Pokédex Web Application Documentation

## Overview

The Pokédex Web app is a modern web application that was developed using the .NET blazor webassembly framework. The purpose of the application is to pull API data from the PokeAPI and display it in a readable manner to the end user. This application functions similar to the fictional pokedex found inside the world of Pokemon and inspiration was heavily drawn from it.

## How To Use The Features

**Pokémon List:** Displays a list of all Pokémon fetched from the PokéAPI. Users can click on the names of the Pokémon to gain more information and data about the Pokémon. This is displayed on a card on the left of the list.

**Search Bar:** Quickly search Pokémon by name.

**Filter Menu:** Filter Pokémon by type, ability, colour, shape, habitat and generation.

**Pokémon Details Card:** Displays full Pokémon information such as stats, abilities, height, and weight after the user clicks on a Pokémon name in the list.

**Fight Button/Comparison Mode:** Allows users to compare Pokémon under the pretence of fighting against each other and the better Pokémon, based on the comparison criteria. is displayed as the winner. To access the comparison mode the user clicks on the fight button and is then prompted to select a Pokémon. After the first selection a second prompt asks the user to select another Pokémon to fight against. The base stats of each Pokémon and their types are displayed. At the bottom of the respective Pokémon's card, it states which Pokémon won the "battle". If it is a tie both Pokémon cards display a tie.

**View Button:** To return to viewing Pokémon details while in comparison mode the user can click on the View button, and it will then work the same as when the application is first accessed.

**Responsive Design:** Works seamlessly on both desktop and mobile screens.

## Project Setup

Requirements:

- .Net SDK 8+
- Visual Studio 2022
- Internet access
- Internet browser

## Running Locally

1. Clone the repository:

git clone <https://github.com/ZeMushroomMan/Pokedex>

2. Open the project in Visual Studio or VS Code.

3. Run the program

4. Open your browser at <https://localhost:7176/> to access the webpage.

## Accessing The Deployed and Hosted version

Alternatively, the entire program is deployed and hosted on Vercel and can be accessed from any modern internet browser at the link <https://pokedex-hosted.vercel.app/>

## Technical Details

**Comparison Criteria:** Pokémon are compared on a total of their base stats and their types. Certain types deal bonus damage to another type, so this is considered. Each Pokémon's attack is modified by a value calculated based on the Pokémon's type versus its opponent's type. The base stats and type modifier values are added as a score and the Pokémon with the higher score is determined to be the winner. If the scores are the same, then it is a tie.

All API requests are cached in the local storage of the browser to reduce the amount of API requests to the PokeAPI.

The program employs sufficient error handling and edge case testing to prevent any runtime errors and any abnormal behaviour. If errors are encountered, they are handled gracefully and provide sufficient detail to the user. More detail can be found in the developer console of the browser.

## Developed By

Muhammad Omar

Built using Blazor WebAssembly, C#, Css, HTML and PokéAPI. Hosted on Vercel .