

# UMSL CS 4220 2021 Summer Semester Project 2

## Overview

For this project, students are tasked with performing networking requests to aggregate and display additional data about a selected Pokémon.

## Repo

Students may choose one of the following;

- Continue developing in the project file downloaded during **Project 1**, adding to the overall functionality of the application.
- Re-clone or download the project file again [here](#), renaming the provided implementation files appropriately.

## Objectives

- Integrate with a server-side API to provide new enhancements.

## Tasks

- Create a [PokédexMenuItemAction](#) and pass it to the [PokédexViewController](#) in `MainViewController.swift` on app launch.
- Create a `UIViewController` subclass that performs GET requests to [PokéAPI](#) using URLs provided by that Pokémon object to collect and display additional data.

## Requirements

- Students may choose one of the following Pokémon attributes to pass to their `UIViewController` subclass;
  - [Pokémon.abilities](#)
  - [Pokémon.stats](#)
  - [Pokémon.types](#)
- The `UIViewController` subclass **MUST** use a `UITableView` to display the elements of the chosen list.

## Requirements cont'd

- UITableViewCells should display **at least two values**.
  - The name property from the element's NamedAPIResource attribute should be used as the primary text.
  - // NOTE: – Each one of the above options only has **1** NamedAPIResource.
- When the user selects a cell;
  - Use the url property from the tapped element's NamedAPIResource attribute to perform a GET request that fetches additional data.
  - Map **at least five attributes (4 single-object, 1 list)** from the response to a Decodable object(s).
  - Display the object using an additional UIViewController subclass.
  - Use a UITableView to display the object's list.
- Visibly block the UI while a network response is pending.  
// HINT: – UIActivityIndicatorView
- Errors **MUST** be safely handled.
- Optionals **MUST** be safely **unwrapped and filtered**.
- Auto Layout is **required**.

## Restrictions

- The [Combine](#) & [Concurrency](#) libraries are prohibited for networking code. They may be used for view models.