Bike results app

Document of how to use the app

This is a coding assessment to showcase how to read a JSON file and populate a grid on a UI

2022

Contents

[Setup to run project 2](#_Toc97587129)

[Screen overviews 2](#_Toc97587130)

[Service 2](#_Toc97587131)

[The front end 3](#_Toc97587132)

[Java script screen 4](#_Toc97587133)

**Table of figures**

[Figure 1: Startup projects to be ran 2](#_Toc97587134)

[Figure 2: Rest service using swagger UI 2](#_Toc97587135)

[Figure 3: Blazor UI 3](#_Toc97587136)

[Figure 4: Grid that shows sorted data 3](#_Toc97587137)

[Figure 5: Button style and placeholder demo 3](#_Toc97587138)

[Figure 6: Java script demo 4](#_Toc97587139)

[Figure 7: Java script result 4](#_Toc97587140)

# Setup to run project

You need to have .Net core 5 installed and then you need to specify the projects to be ran as shown below:

* My Bike Results is the REST service
* Bike Results Web is the web application where the grid sits

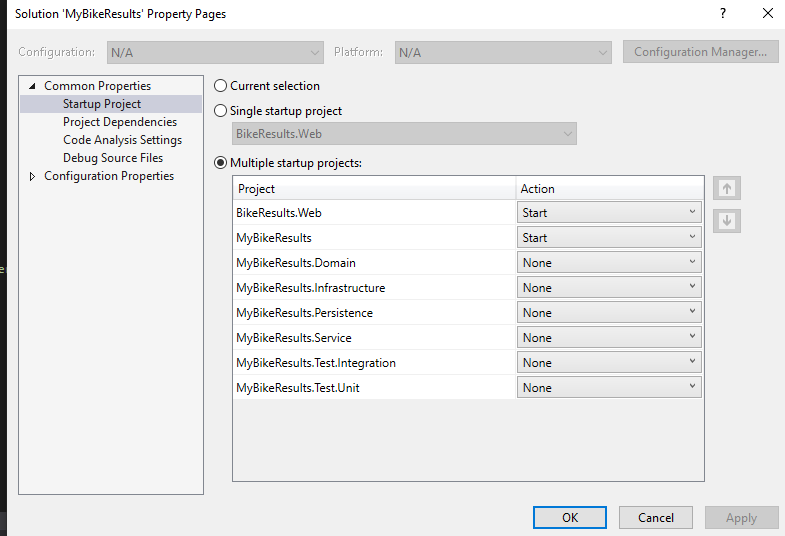


Figure : Startup projects to be ran

# Screen overviews

## Service

There is a Restful service that has a single API that reads the JSON file to get the data to the application.

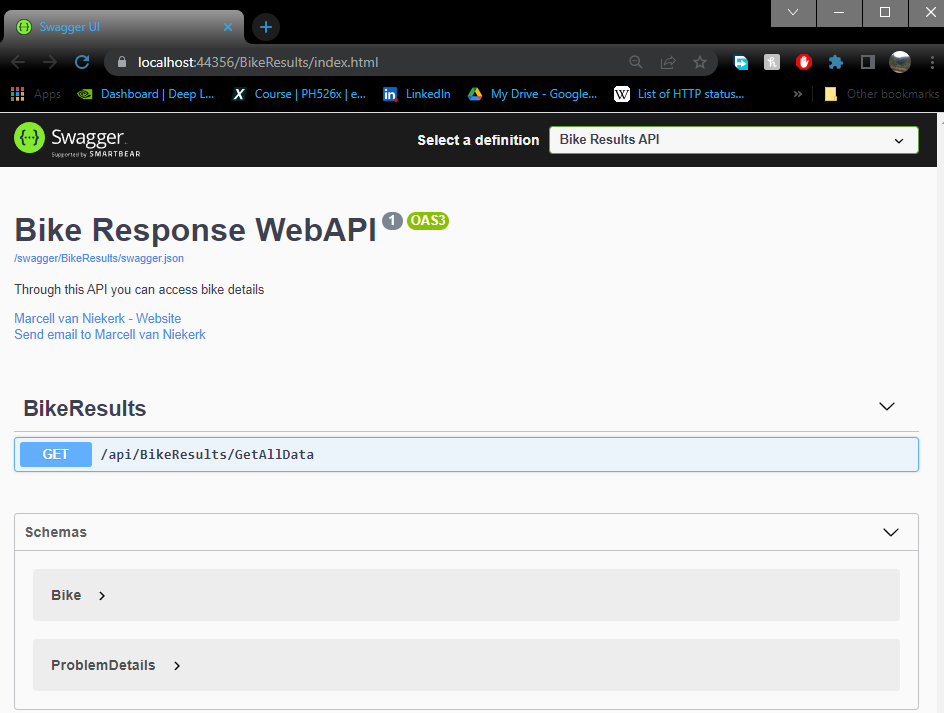


Figure : Rest service using swagger UI

## The front end

The frontend is created using Blazor and the grid is created using MudBlazor. The startup web page will look as follows:

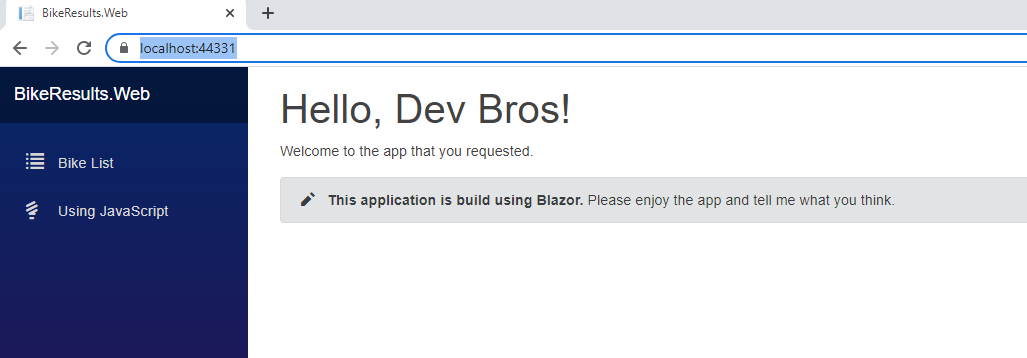


Figure : Blazor UI

When you click on the Bike list button it will show you the following screen where the data gets loaded and sorted using the make and then the model. The Grid does have it’s own search function where you can search by either typing the model or description.

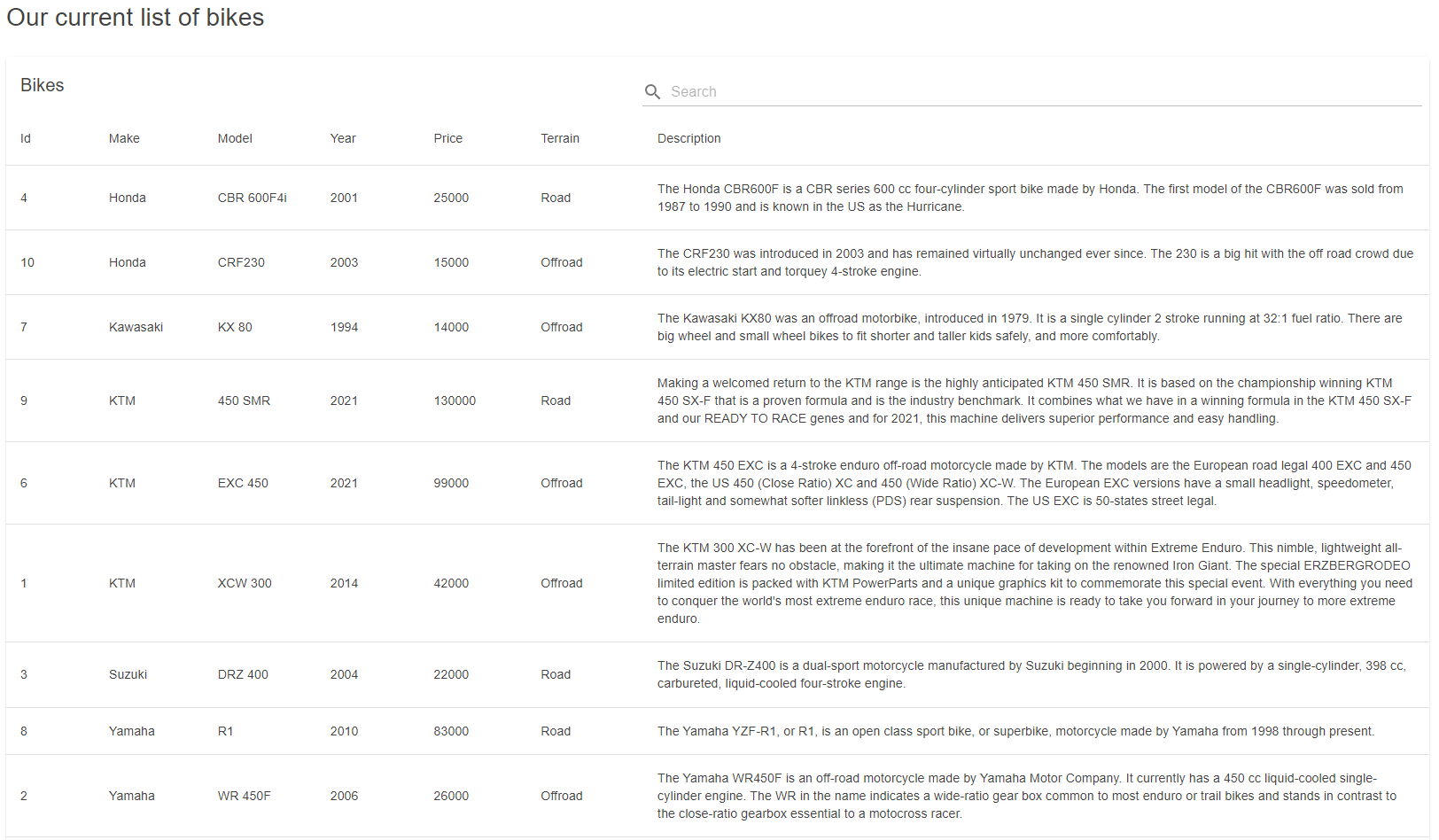


Figure : Grid that shows sorted data

Under the grid there is a small section that shows how the button’s text can be changed by typing in the placeholder input field.

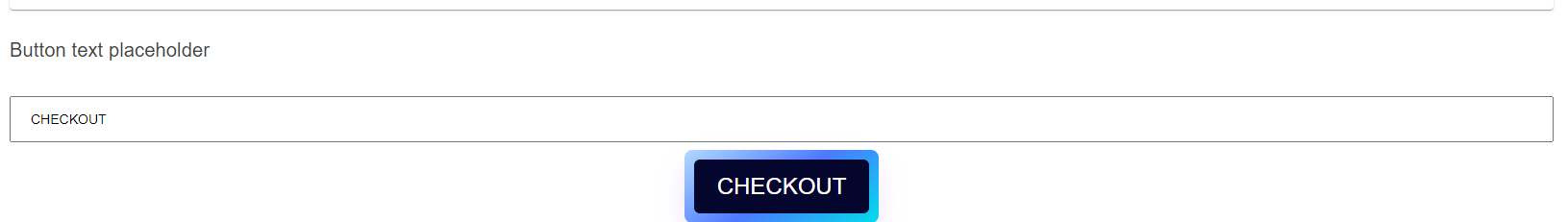


Figure : Button style and placeholder demo

## Java script screen

I attempted to have a Java script function to return the data to the Blazor side so that I can populate a grid, but wasn’t able to return the data into an object. When the button is clicked it will read the JSON file and will display the results on the left side.

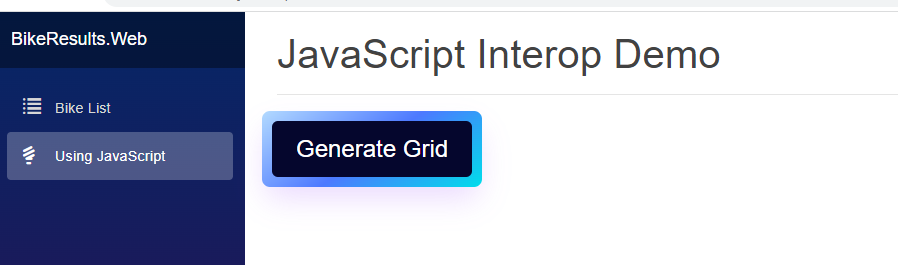


Figure : Java script demo

The picture below shows how the data was read using the java script function:

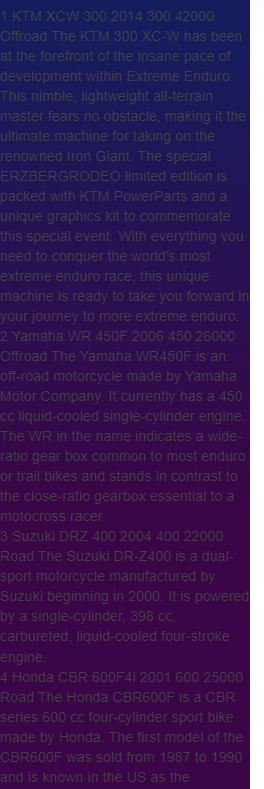


Figure : Java script result