TU Dublin, Tallaght Campus,

Department of Computing

B.Sc. (hons.) in Computing

2022/2023

EAD1- CA3

Out: 14/11, In: 14/12, Value: 30%

Type: elapsed, individual

Lecturer: Gary Clynch

**ASP.Net Core Blazor Project**

An ASP.Net Core Client Side Blazor application is required which provides simple interactive functionality to display data from a public API and provide other simple features.

A Single-Page Application is required implemented using Blazor (client side). 1/2 pages will suffice. The application should interact with a public API of your choice (GET only). The API should be RESTful (JSON), a public open API (with no authentication) is fine.

Pick a topic that you are interested in like entertainment, sports, music, weather, finance, fitness, gaming, health, social media, etc. See here for a list of public APIs <https://www.programmableweb.com/category/all/apis>

Maybe the app show allow a search capability (e.g. input a city to see the weather forecast for the city), or sorting, or maybe a calculation of some sort (e.g. real-time currency conversion), depending on the topic chosen. Include validation on inputs.

Keep the UI clean and simple. Pay attention to code quality and metrics. Use Github to version control your project. Have regular commits to Github, no branching is required.

Host the application on Azure or any public cloud or hosting service of your choice.

Once you have an idea for your project send me an email and I can help scope it for you.

Note: for consuming a public API it needs to support CORS (i.e. issue an access-control-allow-origin header). You can check if this is the case by issuing a GET to the API in Postman and looking at the return headers. You may need to sign up as a user of the API in order to get an API key which is required in order to call the API. Usually the key is specified as a parameter in the query string or using a HTTP basic authentication header, see the documentation for the API on how to exactly do this.

**Deliverables**:

* source code (ZIP)
* Short report (2/3 pages) containing a short description of system, screen shots, URI for deployed app, description of testing performed, code quality/metrics, Github repo URI
* 5 – 10 minute video demo where you walk through the running application