

## Project Overview

This project is part of the recruitment process so that we can better assess the skills and capabilities of the job applicants. The objective is to evaluate their proficiency in various aspects and determine their suitability for the job position. The questions have escalating difficulty. The applicant is expected to complete as many questions as possible.

## Scope

The key areas are the following:

- **C# language syntax:** Applicants should have at least a basic understanding of the C# language.
- **Architecture:** Applicants will demonstrate their ability to design and develop maintainable and clean software.
- **Database:** The applicants are expected to have experience with SQL and database integration with a C# project.
- **Debugging:** Based on the project we could better understand the applicant's ability to debug and provide efficient solutions to problems/tasks.

## Project requirements

The only project requirements are the use of the following:

- **Language:** C# (Choose any framework of your choice).
- **IDE:** Visual Studio.
- **Database:** Microsoft SQL Server.

## Evaluation Criteria

- **Correctness**
- **Code Quality**
- **Efficiency**
- **Documentation**
- **Unit Testing (optional)**

## Submission Process

Applicants should submit their project **within 5 days** along with any additional documentation or instructions in a public GitHub repository. After the completion of the project, please reply to this email with the GitHub URL.

## Project Description

The applicant is expected to implement a Web API and solve the following questions.

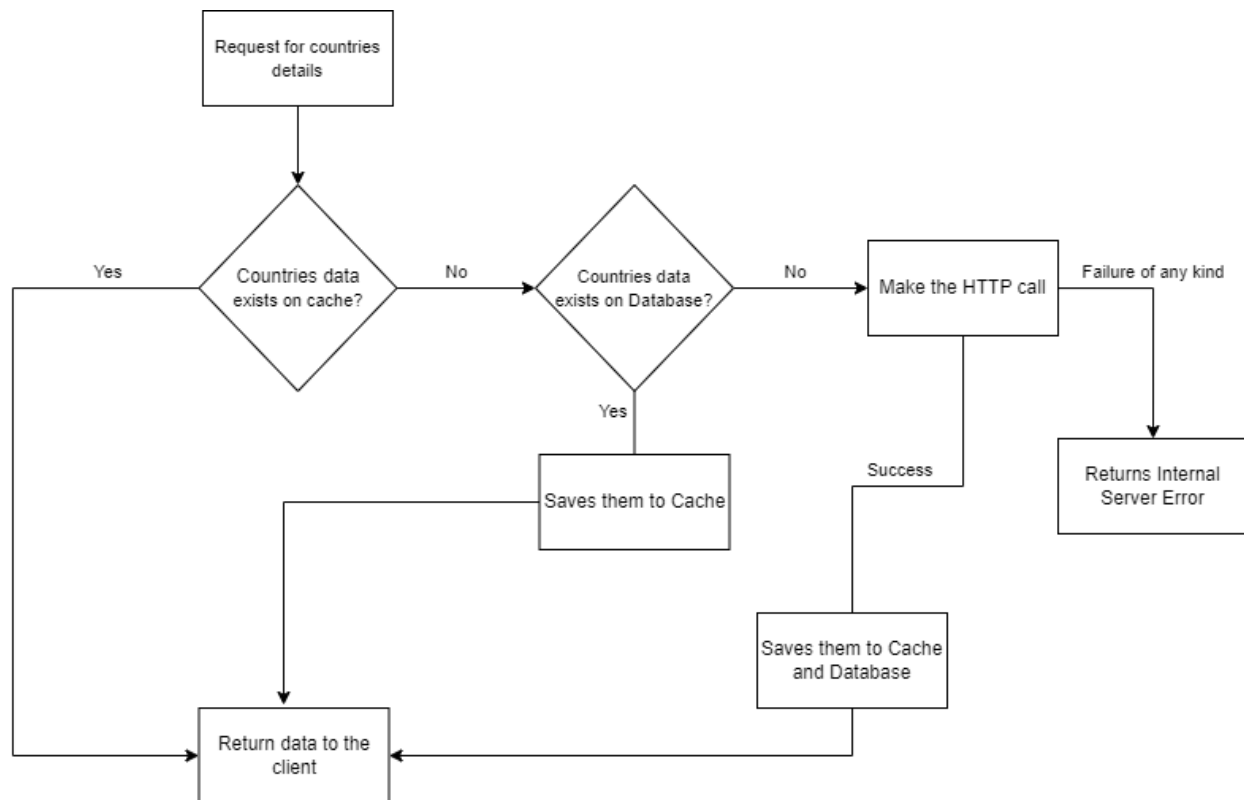
### Questions

- 1) Implement a HTTP Post endpoint that will receive a JSON body of the following class. The second largest integer of the array should be returned.

```
public class RequestObj
{
    public IEnumerable<int> RequestArrayObj { get; set; }
}
```

- 2) In this question, the applicant should implement an HTTP Get endpoint that will be calling a 3<sup>rd</sup> Party API. The API is the restcountries (<https://restcountries.com/>), it is Open Source and Free to use. The endpoint should retrieve all the countries. The response object should return the following properties:
  - a) The common name of the country
  - b) The capital of the country
  - c) The borders of the country
- 3) This question is based on the previous question. Before returning the countries save them in a Microsoft SQL Server.
- 4) This question is based on the 2 previous questions.
  - Except for saving the countries to the database, save them also to a local cache (e.g., MemoryCache).
  - After that, change the endpoint so that the first step is to check if there are saved countries in the cache and then return them, if not check the database and return them (after saving to the cache)
  - If there are no data in cache and database then make the HTTP call (after saving them both to cache and database).

The following chart is given for a better understanding of Question 4.



**Good luck and thank you for your time!**