

CSU33012 - Assignment 2 – Microservices and External Services

Version 1. This assignment is worth 15% of your grade.

Conventions

These are the same as for Assignment 1.

All project **must**

- be version controlled in git
 - on gitlab.scss.tcd.ie
 - in **private** repositories
 - commit incremental changes, do not just commit the final version
- Use the given repository name
- Use the given project name
 - as artifactId in Spring Initializr/pom.xml
 - as project name (root folder of the project structure).
Note that this is different from the repository name.
- Use the namespace [ie.tcd.scss](https://gitlab.scss.tcd.ie)
 - Use this as group ID in pom.xml and base package in Java
- Use Spring Boot 3.1.4 or newer
 - Stable version of Spring 3. No snapshots or milestones.
- Use Maven as a build system
- Use the Maven standard file structure
- Use Java 17
- Include at least the following dependencies in pom.xml, you might have to add more dependencies depending on the given task.

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-test</artifactId>
  <scope>test</scope>
</dependency>
```

Submission

- Use a repository on gitlab.scss.tcd.ie with the specified name.
- Commit regular progress, do not just commit the whole project in the end.
- The content of your project should be in the default branch, that is usually “main” or “master”.

1 Task 2.1 Country Info

(Weight 15%)

Implements a Spring Boot application that provides information about countries.

- Start from the project at <https://gitlab.scss.tcd.ie/botterweck/countryinfo-startercode>
 - After cloning the repository, delete the .git subdirectory and rename the project directory to **countryinfo**
 - You then need to initialise your own git repository in that directory (e.g., via “git init” or in the VCS menu in IntelliJ)
- Your project must conform to
 - Gitlab repository **countryinfo**
 - Project name **countryinfo**
 - Root Package **ie.tcd.scss.countryinfo**
- Implement a Spring Boot application which provides
 - **ie.tcd.scss.countryinfo.controller.CountryController**
 - provides a REST API, i.e., responds to incoming HTTP requests
 - **ie.tcd.scss.countryinfo.service.CountryService**
 - provides methods to retrieve necessary information, used by **CountryController**
 - is using external services that also offer REST APIs

Many methods are already implemented in the provided project. All code locations where you need to fill in information are marked with TODO: in comments.

- CountryController.java
 - Line 69 // TODO: Implement this method
 - Line 81 // TODO: Implement this method
 - Line 113 // TODO: Implement this method
- CountryService.java
 - Line 45 // TODO: Implement this method
 - Line 71 // TODO: Implement this method
 - Line 137 // TODO: Implement the lookup of the translation for the given language

In IntelliJ also see the TODO window (View > Tool Windows > TODO).

1.1 Tests

The Class **CountryControllerTest** already contains a large number of tests.

1.2 REST API Endpoints

The controller is implementing a REST API to retrieve information about countries. The controller should provide the REST API endpoints specified in the sections below. All are using HTTP GET as a method. There are no endpoints with POST, PUT, DELETE etc.

Any lists returned in a response, should be given as elements separated by “, ” that is a comma with a space after it. There should be no comma or whitespace after the last element.

1.2.1 Get Country Information (already implemented)

- URL: **/countries/{countryname}**
- Path Variable: **countryname** - Name of the country to retrieve information for.

- Success Response:
 - HTTP Status: 200 OK
 - Content: Country object with country information
- Error Response:
 - HTTP Status: 400 Bad Request - When the country name is empty
 - HTTP Status: 404 Not Found - When the country information is not found

1.2.2 Get Country Flag URL (already implemented)

- URL: `/countries/{countryname}/flag`
- Path Variable: `countryname` - Name of the country to retrieve the flag URL for.
- Success Response:
 - HTTP Status: 200 OK
 - Content: String URL to the country's flag image
- Error Response:
 - HTTP Status: 404 Not Found - When the flag URL is not found

1.2.3 Get Country Map URL

- URL: `/countries/{countryname}/map`
- Path Variable: `countryname` - Name of the country to retrieve the map URL for.
- Success Response:
 - HTTP Status: 200 OK
 - Content: String URL to the country's map on Google Maps
- Error Response:
 - HTTP Status: 404 Not Found - When the map URL is not found

1.2.4 Get A Country's Continents

- URL: `/countries/{countryname}/continents`
- Path Variable: `countryname` - Name of the country to retrieve continent information for.
- Success Response:
 - HTTP Status: 200 OK
 - Content: List of continents the country is in
- Error Response:
 - HTTP Status: 404 Not Found - When no continents are found for the country

1.2.5 Get Most Populous Countries Containing Substring (*already implemented*)

- URL: `/countries/{substring}/mostPopulous`
- Path Variable: `substring` - Substring to search for within country names.
- Success Response:
 - HTTP Status: 200 OK
 - Content: List of country names containing `substring`, ordered by descending population size
For instance, if the substring is "Bu", then the response might be something like Burkina Faso, Burundi, Bulgaria, Antigua and Barbuda
- Error Response:
 - HTTP Status: 404 Not Found - When no countries are found containing the

substring

1.2.6 Get Most Populous Countries With Population Containing Substring

- URL: `/countries/{substring}/mostPopulousWithPopulation`
- Path Variable: `substring` - Substring to search for within country names.
- Success Response:
 - HTTP Status: 200 OK
 - Content: List of country names containing `substring` plus their population in parentheses, ordered by descending population size
For instance, if the substring is "Bu", then the response might be something like Burkina Faso (20903278), Burundi (11890781), Bulgaria (6927288), Antigua and Barbuda (97928)
- Error Response:
 - HTTP Status: 404 Not Found - When no countries are found containing the substring

1.2.7 Get Country Name Translation (partially implemented)

- URL: `/countries/{countryname}/translation/{language}`
- Path Variables:
 - `countryname` - Name of the country to get the translation for.
 - `language` - Language code for the translation
- Success Response:
 - HTTP Status: 200 OK
 - Content: String of the translated "common" country name (not the longer "official" one)
- Error Response:
 - HTTP Status: 404 Not Found - When a translation is not found

See translation element in <https://restcountries.com/v3.1/name/France> for some examples