

RVM2 — RVM2 TASK 1: DEVELOP AN INTERACTIVE WORLD MAP

JAVASCRIPT PROGRAMMING — D280

PRFA — RVM2

Preparation

Task Overview

Submissions

Evaluation Report

COMPETENCIES

4077.1.1 : Writes Basic Scripts

The learner writes basic scripts to accomplish tasks with JavaScript.

4077.1.2 : Uses Existing Frameworks

The learner uses existing frameworks, assets, and web content that address stakeholder preferences and enhance website functionality.

4077.1.3 : Adds APIs and Web Services

The learner adds data capabilities to web applications with application programming interfaces (APIs) and web services.

INTRODUCTION

Throughout your career in software engineering, you will be asked to enhance website functionality using JavaScript programming in conjunction with existing frameworks, assets, and web content. For this assessment, you will enhance a website using the Angular JavaScript framework and an application programming interface (API) by creating a visual interface for a scalable vector graphics (SVG) map.

The skills you demonstrate in this task will be useful in responding to technical interview questions for future employment. This website may also be added to your portfolio.

Your submission should include a link to the project that contains a copy of the repository branch history file and the URL to the GitLab repository for evaluation. The submission must keep the project file and folder structure intact for the integrated development environment (IDE).

SCENARIO

You are a developer for a state government office. On a previous project, your supervisor asked you to build a website with several pages related to a geographic area in the United States. Now your supervisor wants you to expand on this site and provide a website that provides information on each country in the world.

Your task is to present a map of the world in an SVG format so that each country is highlighted upon a mouse event. Then you will convert the SVG map into an interactive Angular component and connect the application to an API service to provide the country information.

REQUIREMENTS

Your submission must be your original work. No more than a combined total of 30% of the submission and no more than a 10% match to any one individual source can be directly quoted or closely paraphrased from sources, even if cited correctly. The similarity report that is provided when you submit your task can be used as a guide.

You must use the rubric to direct the creation of your submission because it provides detailed criteria that will be used to evaluate your work. Each requirement below may be evaluated by more than one rubric aspect. The rubric aspect titles may contain hyperlinks to relevant portions of the course.

*Tasks may **not** be submitted as cloud links, such as links to Google Docs, Google Slides, OneDrive, etc., unless specified in the task requirements. All other submissions must be file types that are uploaded and submitted as attachments (e.g., .docx, .pdf, .ppt).*

Note: You may use either Microsoft Visual Studio Code or JetBrains IntelliJ as your IDE. No external libraries (other than Angular), third-party sources, or packages may be used in this project's creation.

- A. Create your subgroup and project in GitLab using the "GitLab" web link and the "GitLab How-To" web link by doing the following:
- Clone the project to your selected IDE.
 - Commit with a message and push to the Working branch when you complete *each* requirement listed in parts C, D, E, and F.

Note: You may commit and push whenever you want to back up your changes, even if a requirement is not yet complete.

- Submit a copy of the GitLab repository URL in the "Comments to Evaluator" section when you submit this assessment.
 - Once the project is completed and all commits are finalized, submit a screenshot of the repository branch history retrieved from your repository, which must include the commit messages and dates.
1. Create a README file that contains the configuration details of the project, including **both** of the following items:
- your student ID
 - Angular version

Note: Open the Angular project from your terminal and type the following command:

```
ng --version  
or:  
ng -v
```

This command will output a similar version for your readme file; it is acceptable if your version numbers change:

```
Angular CLI: 14.2.3  
Node: 16.15.0  
Package Manager: npm 8.11.0  
OS: win32 x64  
Angular: 14.2.2
```

2. Push the project files to your Working branch. In addition to your src folder, the branch must include *all* of the following files:
 - tsconfig.app.json
 - tsconfig.json
 - tsconfig.spec.json
 - angular.json
 - package.json
 - package-lock.json
 - README
 - SVG

Note: Any submissions that do not contain the files listed in part A2 and commit messages for each requirement listed in parts C, D, E, and F will not be evaluated.

- B. Using the "SVG Map" web link, provide the world map visual interface for this project in your application.
- C. Using the "World Bank API" web link, identify *each* of the following **six** properties for *each* country:
 - country name (e.g., Chad)
 - country capital (e.g., N'Djamena)
 - country region (e.g., Sub-Saharan Africa)
 - income level (e.g., low income)
 - **two** additional country properties of your choice
- D. Using Angular routing, configure the root component to redirect automatically from the default URL to a URL of your choice (i.e., /map, /home, /index, /main).

Note: Your default route may not be left blank.

- E. Create an HTML layout with **two** columns: **one** column for the map itself and **one** column to show the required country information from the API in part C.
- F. Using event binding, convert the SVG map into an interactive Angular component.
 1. Connect the SVG file to a mouse event handler to transmit data to the component for processing.
- G. Generate an API service using HttpClient to make HTTP calls and include the following methods:
 - **one** method within the service that accepts a two-letter country code as an input parameter that returns additional information gathered from the API for the selected country
 - **one** method within the component that will trigger the service method when a country is selected and set a local variable that will receive the information about the country for display in the appropriate column of the HTML page

- H. Demonstrate professional communication in the content and presentation of your submission.

File Restrictions

File name may contain only letters, numbers, spaces, and these symbols: ! - _ . * ' ()

File size limit: 200 MB

File types allowed: doc, docx, rtf, xls, xlsx, ppt, pptx, odt, pdf, csv, txt, qt, mov, mpg, avi, mp3, wav, mp4, wma, flv, asf, mpeg, wmv, m4v, svg, tif, tiff, jpeg, jpg, gif, png, zip, rar, tar, 7z

RUBRIC

A:GITLAB REPOSITORY

NOT EVIDENT

A GitLab repository is not provided.

APPROACHING COMPETENCE

The subgroup and project are created in GitLab, but 1 or more of the given actions are not completed, or they are completed incorrectly.

COMPETENT

The subgroup and project are created in GitLab, and *all* of the given actions are completed correctly.

A1:README FILE

NOT EVIDENT

A README file is not created.

APPROACHING COMPETENCE

A README file is created, but the configuration details are not accurate, or the file does not contain *both* of the given requirements.

COMPETENT

A README file is created, the configuration details are accurate, and the file contains *both* of the given requirements.

A2:PROJECT FILES

NOT EVIDENT

Project files are not pushed to the Working branch.

APPROACHING COMPETENCE

Project files are pushed to the Working branch, but the branch does not include *all* of the required files.

COMPETENT

Project files are pushed to the Working branch, and the branch includes *all* of the required files.

B:SVG MAP

NOT EVIDENT

The required SVG map is not used to provide the application's visual interface.

APPROACHING COMPETENCE

Not applicable.

COMPETENT

The required SVG map is used to provide the application's visual interface.

C:API WEB LINK

NOT EVIDENT

The required API is not used.

APPROACHING COMPETENCE

COMPETENT

The required API is used, but *all* 6 of the given properties for *each* country are not identified.

The required API is used, and *all* 6 properties for each country are identified.

D:ANGULAR ROUTING

NOT EVIDENT

The root component does not redirect automatically using Angular routing.

APPROACHING COMPETENCE

Not applicable.

COMPETENT

The root component redirects automatically using Angular routing.

E:HTML LAYOUT

NOT EVIDENT

An HTML layout is not created.

APPROACHING COMPETENCE

The HTML layout is created but does not provide *both* required columns.

COMPETENT

The HTML layout is created and includes *both* required columns.

F:CONVERSION TO ANGULAR COMPONENT

NOT EVIDENT

The SVG map is not converted into an interactive Angular component.

APPROACHING COMPETENCE

The SVG map is converted into an interactive Angular component but does not use event binding.

COMPETENT

The SVG map is converted into an interactive Angular component using event binding.

F1:MOUSE EVENT HANDLER

NOT EVIDENT

The SVG file is not connected to a mouse event handler.

APPROACHING COMPETENCE

The SVG file is connected to a mouse event handler, but the map does not function correctly.

COMPETENT

The SVG file is connected to a mouse event handler, and the map functions correctly.

G:API SERVICE

NOT EVIDENT

An API service is not generated.

**APPROACHING
COMPETENCE**

The API service does not use HTTPClient to make HTTP calls, or the API service does not include *both* required methods. Or 1 or *both* of the required methods do not function correctly.

COMPETENT

The API service uses HTTPClient to make HTTP calls, includes *both* required methods, and *both* methods function correctly.

H:PROFESSIONAL COMMUNICATION**NOT EVIDENT**

Content is unstructured, is disjointed, or contains pervasive errors in mechanics, usage, or grammar. Vocabulary or tone is unprofessional or distracts from the topic.

**APPROACHING
COMPETENCE**

Content is poorly organized, is difficult to follow, or contains errors in mechanics, usage, or grammar that cause confusion. Terminology is misused or ineffective.

COMPETENT

Content reflects attention to detail, is organized, and focuses on the main ideas as prescribed in the task or chosen by the candidate. Terminology is pertinent, is used correctly, and effectively conveys the intended meaning. Mechanics, usage, and grammar promote accurate interpretation and understanding.

WEB LINKS

[GitLab](#)

[GitLab How-To](#)

[IntelliJ IDEA \(Ultimate Edition\) IDE – Student License](#)

[MS Visual Studio Code](#)

[SVG Map](#)

[World Bank API](#)

SUPPORTING DOCUMENTS

[IntelliJ Ultimate Edition Instructions.docx](#)