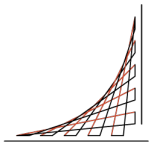


Laboratory 7

Amazon Gateway and Lambda

Author: Angi Paola Jimenez Pira
Teacher: Luis Daniel Benavides Navarro

Date: Monday 5th April, 2021
Bogota, Colombia

1 Introduction

The purpose of this workshop is to learn about Amazon Gateway and Lambda components, developing a basic application using tools as Java, Spark and Docker..

In this document you will find the explanation of challenge purposed in the laboratory 7, the architecture implemented, the tools used during development of this project and other sections that are important to understand how the project works (how install it, how run it, etc.) and if you want extend the code and implement new functionalities.

2 Objectives

- To develop a simple application to conver temperature
- To learn about use of S3 services of AWS.
- To learn about use of API Gateway services of AWS.

3 Glossary

- **Spark Framework:** A micro framework for creating web applications in Kotlin and Java 8 with minimal effort.
- **AWS:** Amazon web service is an online platform that provides scalable and cost-effective cloud computing solutions.
- **EC2:** Amazon Elastic Compute Cloud (Amazon EC2) is a service that provides scalable computing capacity in the Amazon Web Services (AWS) Cloud.
- **Circleci:** CircleCI is a modern continuous integration and continuous delivery (CI/CD) platform.
- **Git:** Git is an open-source version control system that was started by Linus Torvalds—the same person who created Linux.
- **Github:** GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.
- **S3:** Amazon Simple Storage Service (Amazon S3) is storage for the Internet. It is designed to make web-scale computing easier for developers.
- **API Gateway:** An API gateway is an API management tool that sits between a client and a collection of backend services. An API gateway acts as a reverse proxy to accept all application programming interface (API) calls, aggregate the various services required to fulfill them, and return the appropriate result.

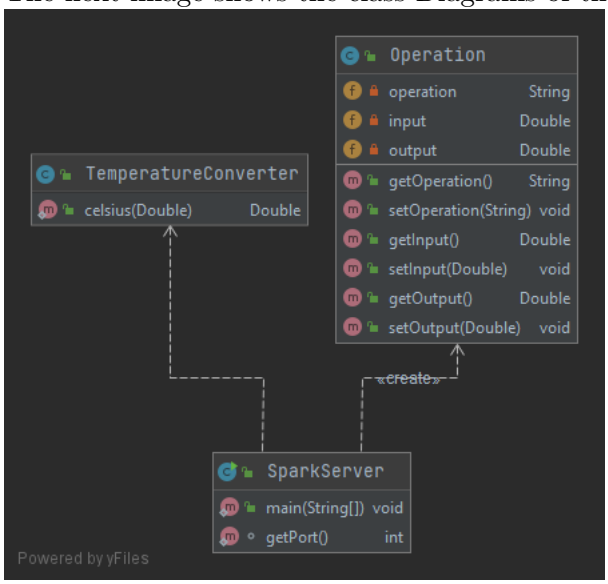
4 Challenge description

- Create a web service in Spark that converts from degrees Fahrenheit to degrees Celcius. The service must respond to a JSON.
- Deploy the service on an AWS EC2 machine and publish it.
- Create a route on the API gateway to access the service. Be careful, the integration is not with lambda function.
- Create a JAVA application to use the service. Deploy the application on S3. Make sure it is available over the internet.
- Test the web application
- Submit the code developed on Github, a test report, and a video with the experiment running.
- Try that the service in EC2 is not accessible over the internet, it should only be accessible through the API gateway.

5 Architecture of program

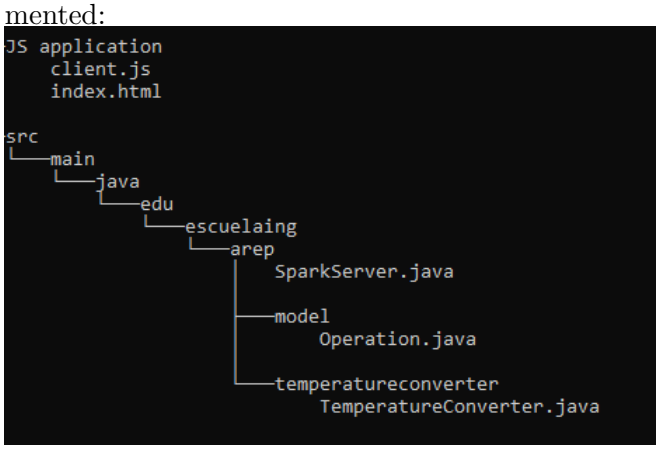
5.1 Class Diagram

The next image shows the class Diagrams of the solutions implemented:



5.2 Project Tree

The next image shows the structure of the projects generated by the Maven dependency manager, with the objective of giving more information for the reader about how the solutions were implemented:



5.3 Technology Stack

- **Development environment:** Java Development Kit 8, IntelliJ IDEA, Spark Framework, Keytool
- **Dependency Manager:** Maven
- **Version Control:** Git, Github
- **Deploy and build:** EC2, S3, API Gateway, Circleci

5.4 Documentation

The documentation and instructions for use are in a git repository: https://github.com/angipaola10/AREP-LAB7.

6 Conclusions

- The use of tools like Circleci to build a web app is very important because this tools help us to check the code and test developed in the program. Also, with this tools we can found bugs, duplicated code and other factors that make our project less efficient.

References

[1] Oracle. https://docs.oracle.com/javase/8/docs/technotes/tools/unix/keytool.html

[2] SimpliLearn. https://www.simplilearn.com/tutorials/aws-tutorial/what-is-aws

[3] Circleci. https://circleci.com/docs/enterprise/overview/

[4] Spark. https://sparkjava.com/

[5] OpenSource. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html