1. **Sprint 1**
   1. **Objectives**

The objective of this sprint was to implement a lambda function which can monitor the availability and latency of multiple websites.

* 1. **Technologies Used**

Following technologies were used in this project:

* + 1. **Cloud9**

Cloud9 was used to create the virtual environment to develop the project.

**1.2.2 Lambda**

Lambda was used to deploy the project to AWS console.

* + 1. **S3 Bucket**

S3 bucket was used to store multiple URL of the websites on which availability and latency was calculated.

* 1. **Sprint breakdown**

The sprint was divided into 4 tasks.

1. **Hello world lambda function**

The 1st task was to create a simple hello world lambda function to understand the working of AWS console and different features included in it.

1. **Web-Health monitoring function**

The 2nd task was to monitor the availability and latency of a website. For this I used [www.skipq.org](http://www.skipq.org) website and wrote a lambda function to monitor the availability and latency of the website.

1. **Periodic invocation of lambda function**

The 3rd task was to invoke the lambda function after every minute and a scheduler was set to invoke the function after every minute.

1. **Create S3 bucket and monitor the health of two websites**

A s3 bucket was created and a json file with two links of the websites was uploaded in the bucket. Then by using boto3 SDK the json data was read from the s3 bucket and availability and latency was calculated for two websites.

* 1. **Errors and Solutions**

During the implementation of the project, I faced multiple errors. The details of the error and their solutions are given below:

1. **Storage Error**

There was an error that the device is running of low storage so I increased the storage of the virtual environment from 10Gb to 20Gb.

1. **No module named aws-cdk**

This error was due to no packages installed on the environment so I installed the packages required for the project and the error was resolved.

1. **You have added another git repository inside your current repository**

This error came when I pasted my project inside cloned Triangulum directory. I solved this error by following commands:

git add Awais\_WebHealth\_Project/

git commit -m “project added”

git push

* 1. **Function documentation**

**def create\_lambda(self, newid, asset, handler):**

**Description:**

This function is used to setup the lambda function.

**Parameters:**

The function takes three parameters id is the name of the function, asset and handler is the name of the function lambda will execute.

**Return value:**

The function returns a function.

**def get\_availability(url):**

**Description:**

This function calculates the availability of given URL.

**Parameters:**

The function take the url of the website as parameter to calculate the availability.

**Return value:**

The function return 1.0 if the website is up and 0.0 if the website is down.

**def get\_latency(url):**

**Description:**

This function calculates the latency of given URL.

**Parameters:**

The function take the url of the website as parameter to calculate the latency.

**Return value:**

The function returns the time in second the website took to respond.

**client.get\_object(Bucket='awaiswebhealthbucket',Key='links.json'):**

**Description:**

This function reads the data from file stored in s3 bucket.

**Parameters:**

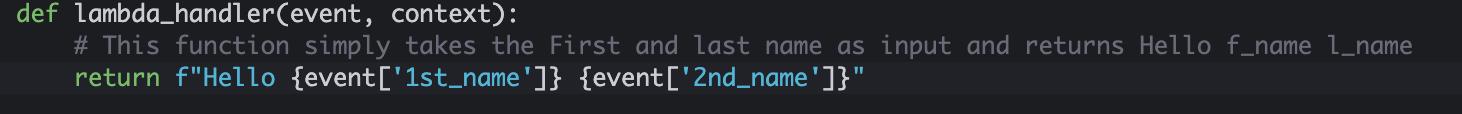
The function takes two parameters. 1st one is the name of s3 bucket and 2nd one is the name of file from which to read the data.

**Return value:**

The function returns an object with all the data present in the file.

* 1. **Implementation**
     1. **Hello World lambda function**

The following function simply print the Hello 1st name and 2nd name



* + 1. **Get Availability function**

The following function get the availability of given URL. It returns 1.0 which means the website is available while it returns 0.0 when the site is down.

Text

Description automatically generated

**1.2.3 Get Latency function**

The following function get the latency of given URL. This function get the current time and then request a response from given url and then return the time difference of the response.

Text

Description automatically generated

* + 1. **Alarm for the Availability**

The following code set the alarm and threshold values for availability.

**Text

Description automatically generated**

**1.2.5 Alarm for the Latency**

The following code set the alarm and threshold values for latency.

**Text

Description automatically generated**

**1.2.6 Parsing the value from SNS notification**

The following function parse the value from sns notification and pass the values to dynamoDb table.

Text

Description automatically generated

* 1. **Results**

**1.3.1 Availability and latency**

The following screenshot shows the results for availability and latency of [www.skipq.org](http://www.skipq.org) website.

Graphical user interface, text, application

Description automatically generated

**1.3.2 Availability cloud watch graph**

The following screenshot shows the graph for availability and latency of [www.skipq.org](http://www.skipq.org) website.

Graphical user interface, application

Description automatically generated

**1.3.3 Latency cloud watch graph**

The following screenshot shows the graph for latency and latency of [www.skipq.org](http://www.skipq.org) website.

Timeline

Description automatically generated