

SCRUM

**Document**

**Written by**

Umair Habib

Table of Figures

[Figure 1: Hello World Lambda Function 4](#_Toc103351152)

[Figure 2: Flow diagram for sprint2 project 5](#_Toc103351153)

[Figure 3: Urls availability & latency values 6](#_Toc103351154)

[Figure 4: Availability and Latency Metrices 6](#_Toc103351155)

[Figure 5: Latency alarm matrix 7](#_Toc103351156)

[Figure 6: Availability alarm matrix 7](#_Toc103351157)

[Figure 7: DynamoDB table selected parameters value 8](#_Toc103351158)

Table of Contents

[1. Sprint 1 2](#_Toc103351100)

[1.1. Orientation 2](#_Toc103351101)

[1.1.1. What is cloud computing? 2](#_Toc103351102)

[1.1.2. Benefits of cloud computing 2](#_Toc103351103)

[1.2. Amazon Web Services(AWS) 2](#_Toc103351104)

[1.2.1. What is AWS 2](#_Toc103351105)

[1.2.2. Benefits 2](#_Toc103351106)

[1.3. DevOps 3](#_Toc103351107)

[1.3.1. What is DevOps 3](#_Toc103351108)

[1.3.2. Benefits 3](#_Toc103351109)

[1.4. Task 1: Hello Lambda 3](#_Toc103351110)

[1.4.1. Implementation 3](#_Toc103351111)

[1.4.2. Result 3](#_Toc103351112)

[2. Sprint 2 4](#_Toc103351113)

[2.2. Task1: Web Health Lambda 4](#_Toc103351114)

[2.2.1. Implementation 4](#_Toc103351115)

[2.2.2. Result 4](#_Toc103351116)

[2.3. Task 2: Web Matrix Lambda 5](#_Toc103351117)

[2.3.1. Implementation 5](#_Toc103351118)

[2.3.2. Result 5](#_Toc103351119)

[2.4. Task 3: Web Alarm Lambda 5](#_Toc103351120)

[2.4.1. Implementation 5](#_Toc103351121)

[2.4.2. Results 5](#_Toc103351122)

[2.5. Task 5: Dynamo DB & Web Lambda 6](#_Toc103351123)

[2.5.1. Implementation 6](#_Toc103351124)

[2.5.2. Result 6](#_Toc103351125)

[2.6. Errors & It’s Solution 7](#_Toc103351126)

# Sprint 1

## 1.1. Orientation

### What is cloud computing?

The delivery of various services over the Internet is known as cloud computing. These resources include data storage, servers, databases, networking, and software, among other tools and applications. As long as an electronic gadget has internet access, it has access to data and the software programmers needed to execute it.

### Benefits of cloud computing

#### Cost saving

If you are worried about the price tag that would come with making the switch to cloud computing, you aren't alone [20% of organizations are concerned about the initial cost of implementing a cloud-based server](https://powermore.dell.com/2015-global-technology-adoption-index/). But those who are attempting to weigh the advantages and disadvantages of using the cloud need to consider more factors than just initial price they need to consider ROI.

#### Security

Many organizations have security concerns when it comes to adopting a cloud-computing solution. After all, when files, programs, and other data aren't kept securely onsite, how can you know that they are being protected? If you can remotely access your data, then what's stopping a cyber-criminal from doing the same thing? Well, quite a bit, actually.

#### Flexibility

Your business has only a finite amount of focus to divide between all of its responsibilities. If your current IT solutions are forcing you to commit too much of your attention to computer and data-storage issues, then you aren't going to be able to concentrate on reaching business goals and satisfying customers. On the other hand, by relying on an outside organization to take care of all IT hosting and infrastructure, you'll have more time to devote toward the aspects of your business that directly affect your bottom line.

## Amazon Web Services(AWS)

### What is AWS

AWS (Amazon Web Services) is a comprehensive, evolving cloud computing platform provided by Amazon that includes a mixture of infrastructure as a service (IaaS), platform as a service (PaaS) and packaged software as a service (SaaS) offerings.

### Benefits

#### Easy to Use

AWS is designed to allow application providers, ISVs, and vendors to quickly and securely host your applications – whether an existing application or a new SaaS-based application. You can use the AWS Management Console or well-documented web services APIs to access AWS’s application hosting platform.

#### Reliable

With AWS, you take advantage of a scalable, reliable, and secure global computing infrastructure, the virtual backbone of Amazon.com’s multi-billion-dollar online business that has been honed for over a decade.

#### Scale able

Using AWS tools, Auto Scaling, and Elastic Load Balancing, your application can scale up or down based on demand. Backed by Amazon’s massive infrastructure, you have access to compute and storage resources when you need them.

## DevOps

### What is DevOps

DevOps is the combination of cultural philosophies, practices, and tools that increases an organization's ability to deliver applications and services at high velocity: evolving and improving products at a faster pace than organizations using traditional software development and infrastructure management processes.

### Benefits

#### Ensure Fast Deployment

Faster and more frequent delivery of updates and features will not only satisfy the customers but will also help your company take a firm stand in a competitive market.

#### Stabilize Work Environment

Do you know that the tension involved in the release of new features and fixes or updates can topple the stability of your workspace and decreases the overall productivity? Improve your work environment with a steady and well-balanced approach of operation with DevOps practice.

#### Improvement in Product Quality

Collaboration between development and operation teams and frequent capturing of user feedback leads to a significant improvement in the quality of the product.

## Task 1: Hello Lambda

### Implementation

In our first task on AWS, we implemented a hello world lambda function, as a beginner everybody do in any new programming field. Created the directory for project imported the important libraries to our stack file and created a Lambda file in resources folder to start the process. First of all, defined a handler function for our Lambda file in stack. And in that file defined all the parameters required for a lambda function invoke in our stack file. In lambda file defined function for event and context and then given two strings to it for printing in console. User name concatenated with the hi user name have a nice day to print my first cloud computing message.

### Result

Results of first hello lambda function are given in figure 1.

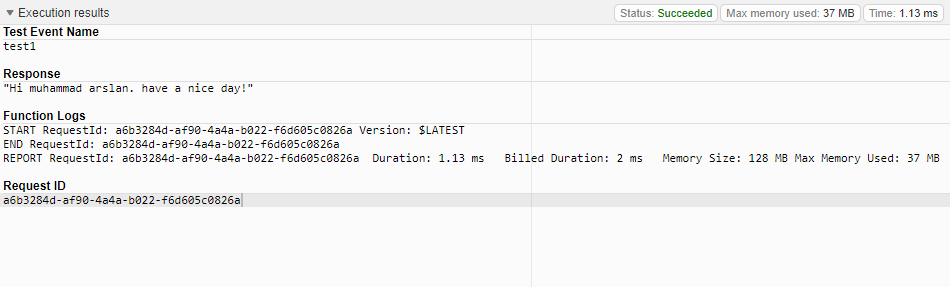


Figure 1: Hello World Lambda Function

# Sprint 2

* 1. What is the sprint2 Project?

As a Skipq DevOps trainee, I want to create a CDK system for web pages to monitor the latency and availability. In any case, if defined parameters (latency and availability) breaches the certain threshold, alarm will be generating and notify the web admin through SNS.

The flow diagram of sprint2 project.

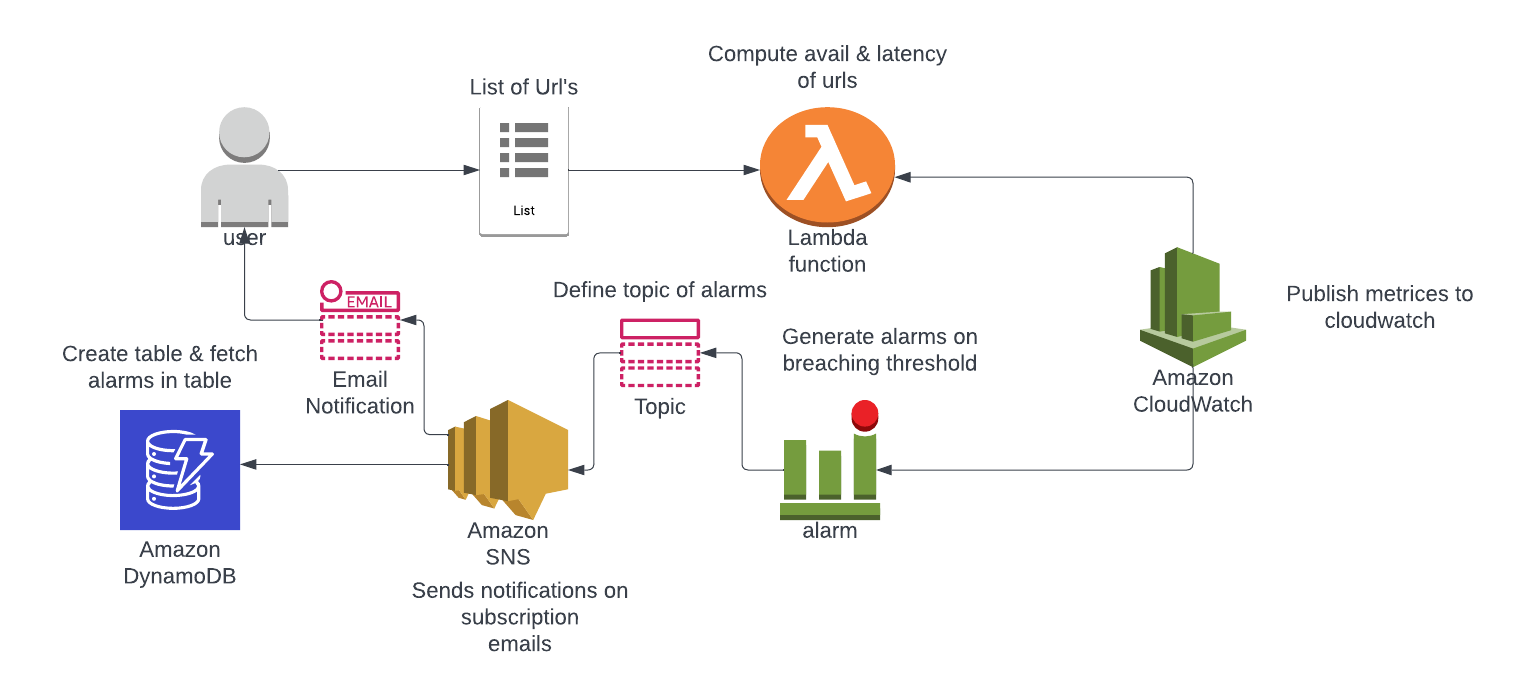


Figure 2: Flow diagram for sprint2 project

## Task1: Web Health Lambda

### Implementation

In this task we tried to implement a web health lambda function. This function returns the availability and latency of a given URL’s in Numeric values. As the same way started the lambda function and stack file for this task. Our stack files remain same because we don’t need any extra libraries for this task. We made changes in lambda file and created two functions. First one is for calculating the availability of given URL’s in terms of ones and zeros. Used built in function for that which returns Boolean output, so using if conditions converted them in ones and zeros. If a Web is not down it returns one else zero. And then created a function for latency of that web. It returns float values between zeros and ones. For latency requested a response from the given URL’s and saved the time before and after the response. Taken their difference and converted it to seconds as latency value.

### Result

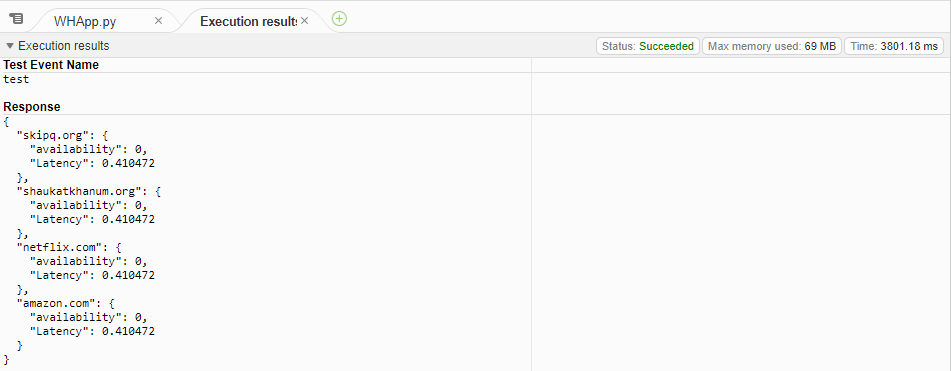


Figure 3: Urls availability & latency values

## Task 2: Web Matrix Lambda

### Implementation

In this task created a periodic lambda function that invokes after a certain duration to show the availability and latency of URL’s in a matrix graph to observer the trends related to that two parameters. For this task again used the functions from previous file that returns the values of latency and availability in numeric form. Created a matrix and given the values to that matrix. To be plotted in a graph and shown in cloud watch. In stack file created schedule for the lambda function to invoke after certain amount of time intervals. Imported cloud watch and related libraries in the lambda file. Initialized lambda rule function in the stack file to get access to cloud watch services.

### Result

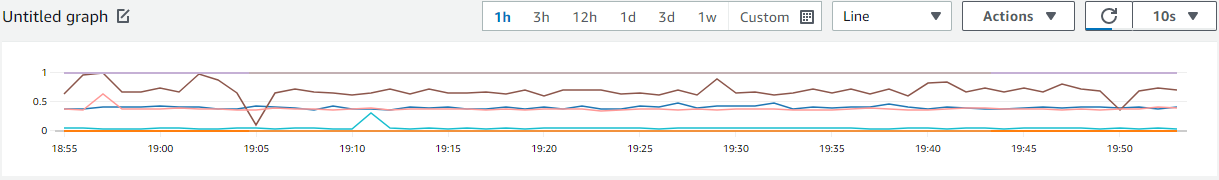


Figure 4: Availability and Latency Matrices

## Task 3: Web Alarm Lambda

### Implementation

This function not only shows the values of latency and availability of URL’s but creates an alarm if the values go beyond a defined threshold. For this function we used the functions from previous task. To implement it created another matrix in stack file and given it the same parameters as the matrix in lambda function. So that both of them get merged. Generated a threshold for each parameter and defined an alarm function from AWS libraries to create an alarm, given this function all the parameters required to generate an alarm including the matrix and threshold. After that generated an action for each alarm by sending an email to the related person. For this imported the actions and subscriptions libraries. Subscribed to the email of given user and created an action for that email. So it sends data of the alarm to all subscribed emails.

### Results

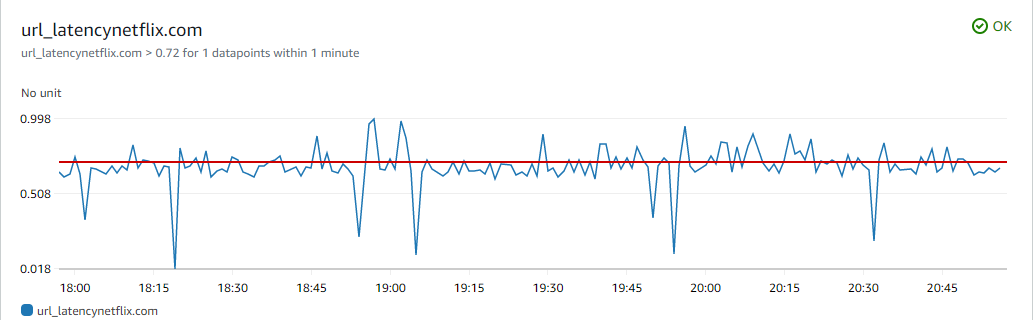


Figure 5: Latency alarm matrix

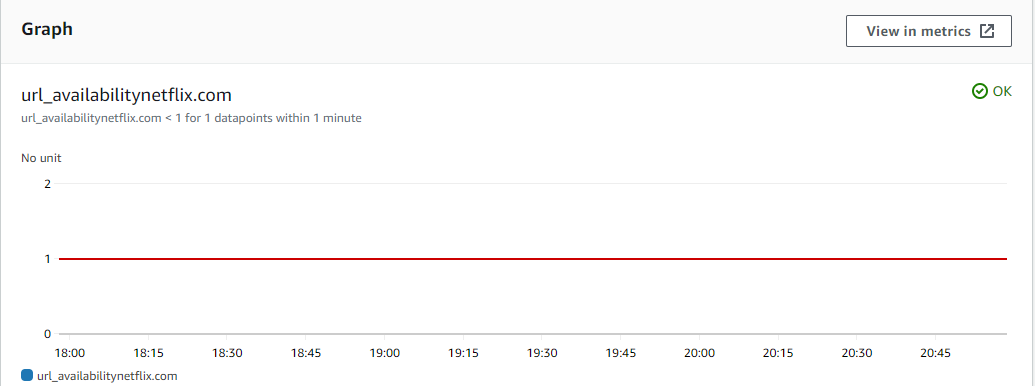


Figure 6: Availability alarm matrix

## Task 5: Dynamo DB & Web Lambda

### Implementation

During this task created a table for dynamo DB database to update the alarm values in it. First of all, created the table for dynamo DB and then given lambda function all the access to read and write in that function. Created a separate lambda file for adding sns alarm values to it created events for message of sns alarm and provided the message values to our table to be stored in dynamo DB database.

### Result

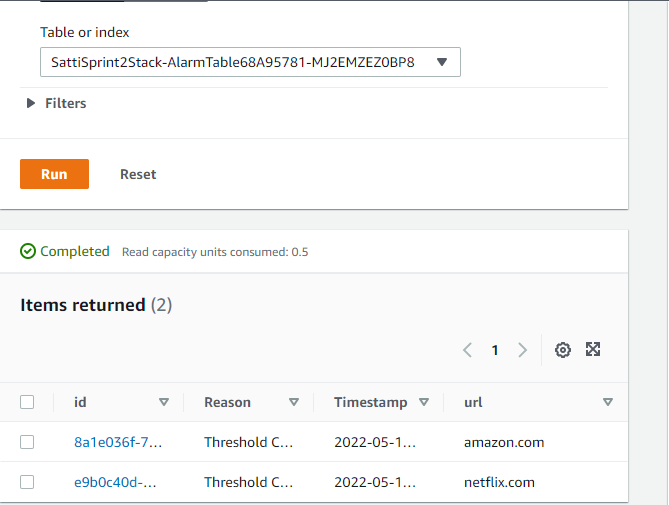


Figure 7: Dynamo DB table selected parameters value

## Errors & It’s Solution

"errorMessage": "2022-05-13T19:16:36.584Z 735165a4-2f66-48d5-b2a2-a6c95b7fc3ba Task timed out after 3.04 seconds"

##### Solution:

I faced this error. I just the increase the time in the lambda function by exploring the documentation. And my error got resolved.

timeout: cdk.Duration.minutes(5).

Type Error: lambda\_Stack.create\_lambda() takes 4 positions arguments but 5 were given.

##### Solution:

I just add 5th argument (role) in my lambda function and my error got resolved.

"errorMessage": insufficient data in alarm.

##### Solution:

Then check the metric name in stack and in web health lambda where you are publishing data to metric. My error was I was using different metric name. May be this is issue you are facing too.

## Technologies Used in Sprint2

### Lambda

Lambda is a compute service that lets you run code without provisioning or managing servers. Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, code monitoring and logging. With Lambda, you can run code for virtually any type of application or backend service. All you need to do is supply your code in one of the [languages that Lambda supports](https://docs.aws.amazon.com/lambda/latest/dg/lambda-runtimes.html).

### Cloud Watch

Amazon Cloud Watch is a monitoring and observability service built for DevOps engineers, developers, site reliability engineers (SREs), IT managers, and product owners. Cloud Watch provides you with data and actionable insights to monitor your applications, respond to system-wide performance changes, and optimize resource utilization. Cloud Watch collects monitoring and operational data in the form of logs, metrics, and events. You get a unified view of operational health and gain complete visibility of your AWS resources, applications, and services running on AWS and on-premises. You can use Cloud Watch to detect anomalous behavior in your environments, set alarms, visualize logs and metrics side by side, take automated actions, troubleshoot issues, and discover insights to keep your applications running smoothly.

### Dynamo BD

Amazon Dynamo DB is a fully managed Non SQL database service that provides fast and predictable performance with seamless scalability. Dynamo DB lets you offload the administrative burdens of operating and scaling a distributed database so that you don't have to worry about hardware provisioning, setup and configuration, replication, software patching, or cluster scaling. Dynamo DB also offers encryption at rest, which eliminates the operational burden and complexity involved in protecting sensitive data. For more information, see [Dynamo DB](https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/EncryptionAtRest.html) Encryption at Rest.