National College of Ireland

Configuration Manual

MSc Research Project
Cloud Computing

Murugalakshmi
Student ID: 21207232

School of Computing
National College of Ireland

Supervisor: Yasantha Samarawickrama

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Murugalakshmi
Student ID:	21207232
Programme:	MSc in Cloud Computing
Year:	2022-2023
Module:	MSc Research Project
Supervisor:	Yasantha Samarawickrama
Submission Due Date:	14/08/2023
Project Title:	Configuration Manual
Word Count:	780
Page Count:	8

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Murugalakshmi
Date:	13th August 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Murugalakshmi
21207232

1 Introduction

This Configuration Manual applies to the NCI Research project's requirements. In the following section, we will discuss the software tools and configurations needed to set up the application to use the novel algorithm proposed which is based on the data size of the task.

2 Objective

The intention is to maximize the performance of mobile computing by using a Data Size-Based Algorithm (DSBA) for offloading the high computational tasks, especially in handheld devices. This requires performing a dynamic evaluation of the complexity of operation as well as the size of data in order to minimize latency, decrease energy consumption, and maintain a balance between local execution and offloading.

3 Application Requirement

In order to perform this algorithm we had to create an android application to test how this data size benefits to offloads any task to cloud. The decision engine has to determine the data size and decide to offload it locally or into a cloud platform. This same can be also done by using fog devices but having an external fog device comes with another subject of challenges which are covered under MCC Challenge article where Author gives an overview of the limitations under Mobile Cloud computing (MCC)

3.1 Following are the tools and technologies used

- **Android Studio**
  - For the development of the application,Android studio Flamingo 2022.2.1 Patch 2 is used. The virtual machine is OpenJDK 64-Bit Server of Memory: 1280M Cores: 12
  - The IDE is integrated with out GITHub account for version control. The repository is linked to this Android studio.
  - The developed app can be installed in any Android device which has the latest 2 version of android, we have used an emulator from the Studio for demonstration

Figure 1: Android Studio

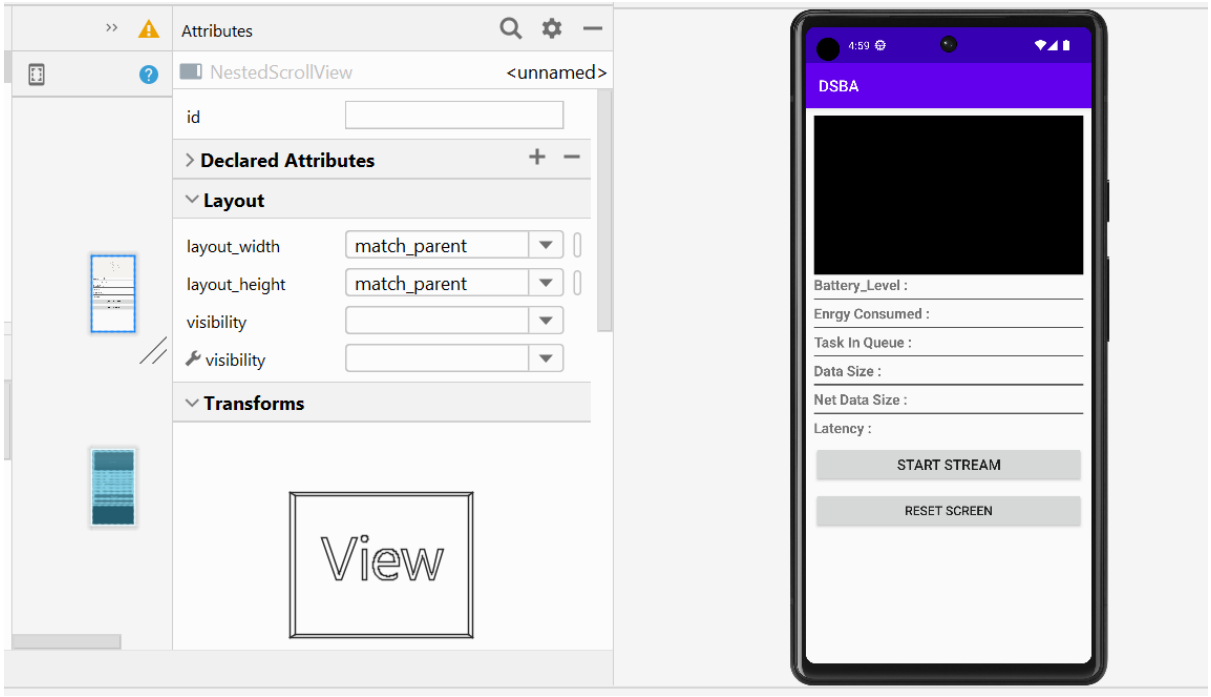


Figure 2: Application Development

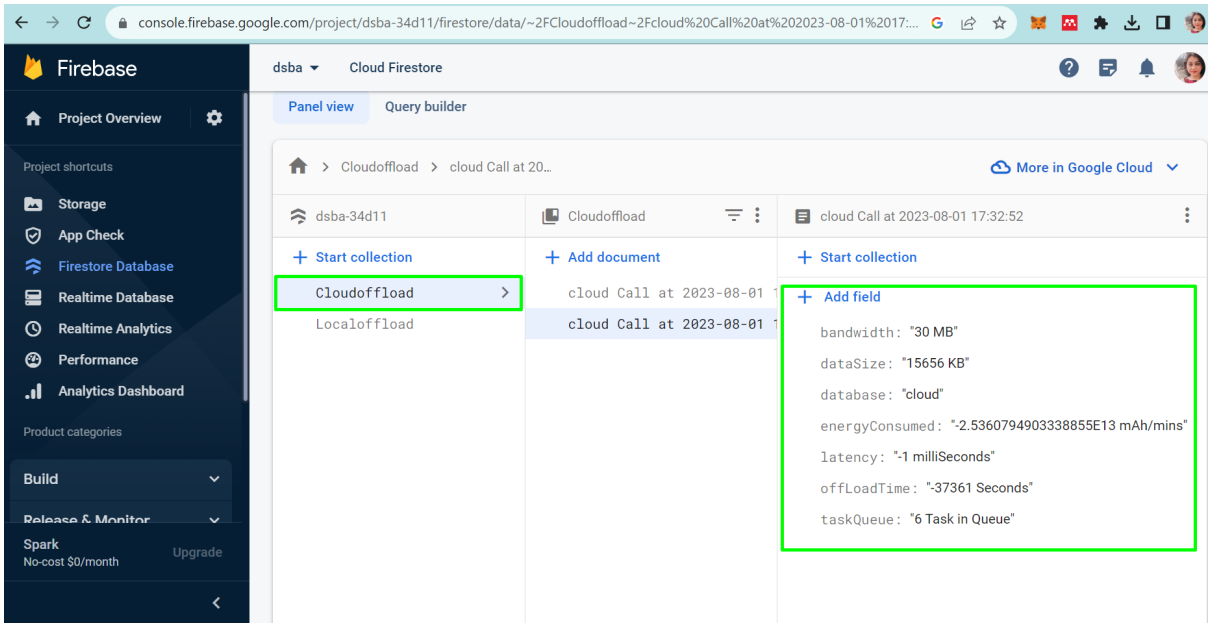


Figure 3: FireBase Database

- The results are calculated on Google Pixel Android device emulator.

- **Firebase Data Storage**
  - Create a Firebase project on the Google Firebase console
  - Link the Android studio project to Firebase project
  - Define the database structure and setup security rules to control access data
  - Testing this experiment we have uploaded video using the storage available under Firebase
  - Firebase is also used to monitor the performance
  - Firebase Analytics is used to determine the results

- **GitHub**
  - The code is pushed to GitHub for version control and seamless integration with the cloud services.
  - Also this centralized repository helps collaborate and track changes to keep the code updated
  - The main purpose of using GitHub is for their quick integration with CI/CD pipeline in order to test, deploy and scale up the services.

4 AWS Cloud Configuration

We use AWS Cloud Platform to offload the task and below are the amazon web services used as following

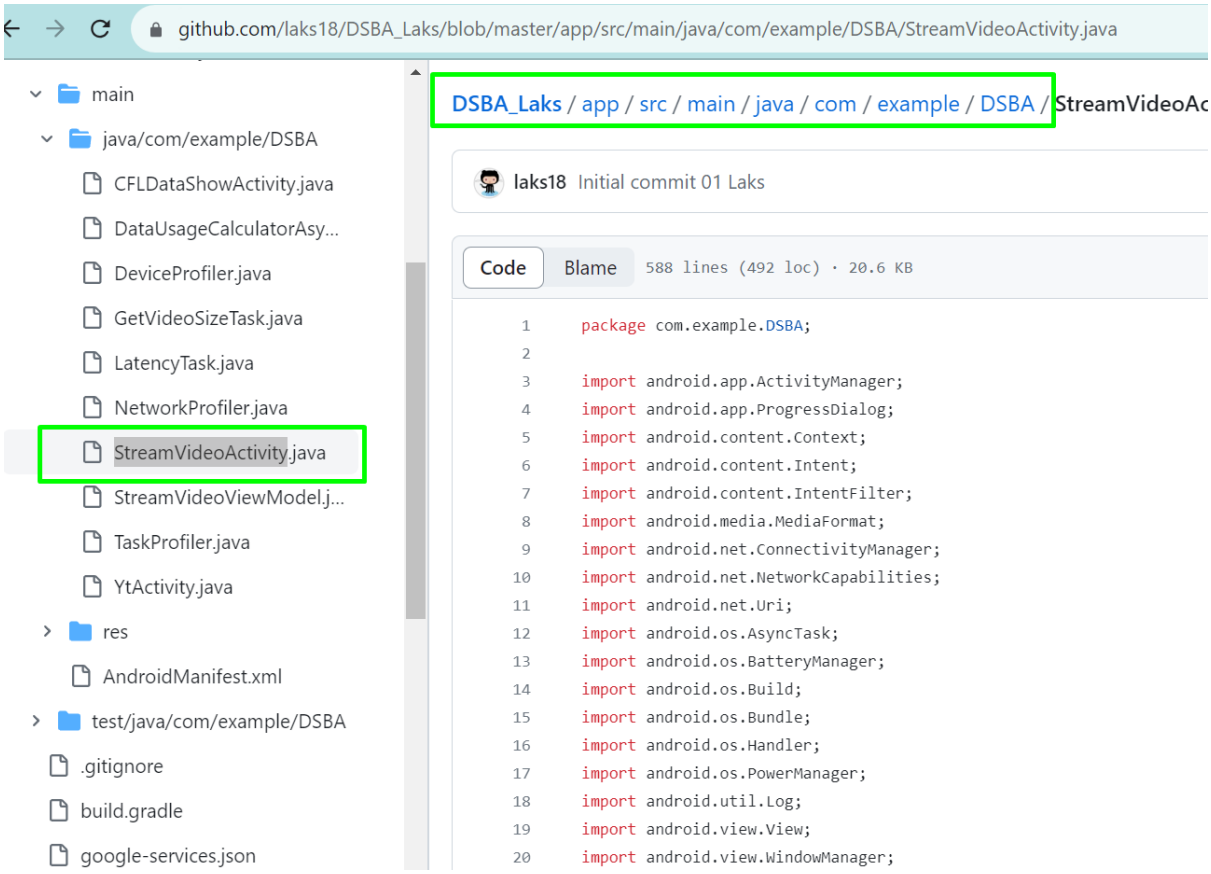


Figure 4: GitHub

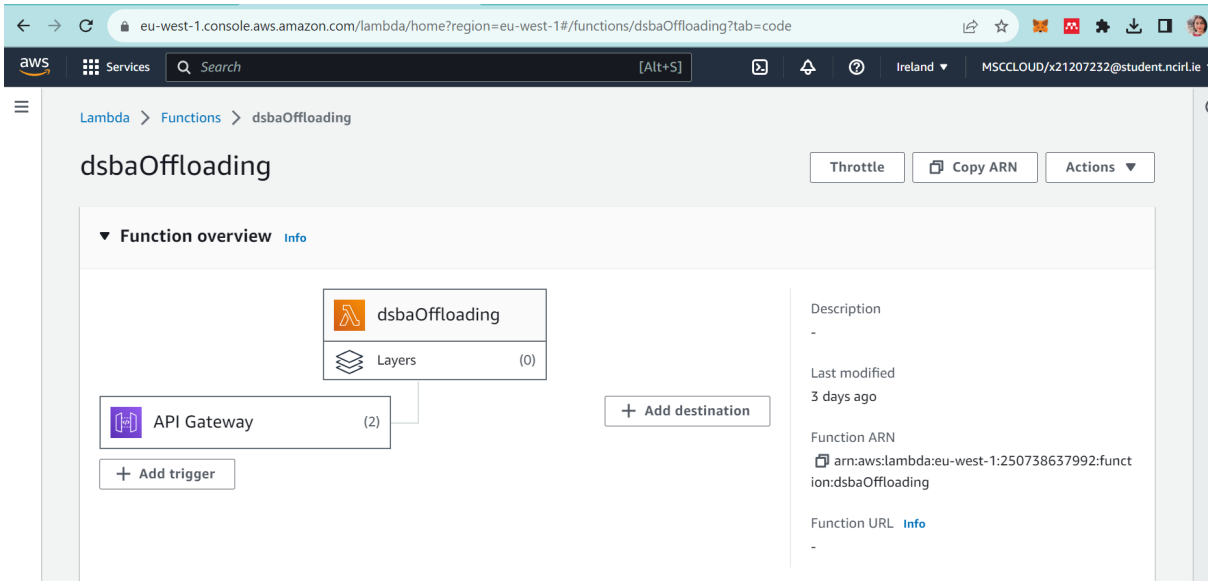


Figure 5: AWS Lambda Function Configuration

- **AWS Lambda**
  - Sign-in to AWS account and search for AWS Lambda
  - Create a Lambda function
  - Ensure you have IAM role available to interact with AWS service (It is provided in the ncirl cloud services)
  - You can use the basic information and name your function, runtime language and required permissions and proceed.
  - Add the network configuration under VPC section
  - Add the trigger source in our case it is the API Gateway
- **AWS API**
  - To begin with search AWS API in the management console of AWS homepage
  - Create an API , Select the type to be REST as it is suitable with Lambda, HTTP and other AWS services
  - Build API and create resources
  - Actions will be POST action as shown in the figure 5 for HTTP operations
  - Configure the request and response mappings
    Choose a deployment name and description , once deployed we get an URL to access the API
- **AWS API Gateway**
  - To create API Gateway search the service in the management console as done above
  - Click on API button and choose RESTAPI

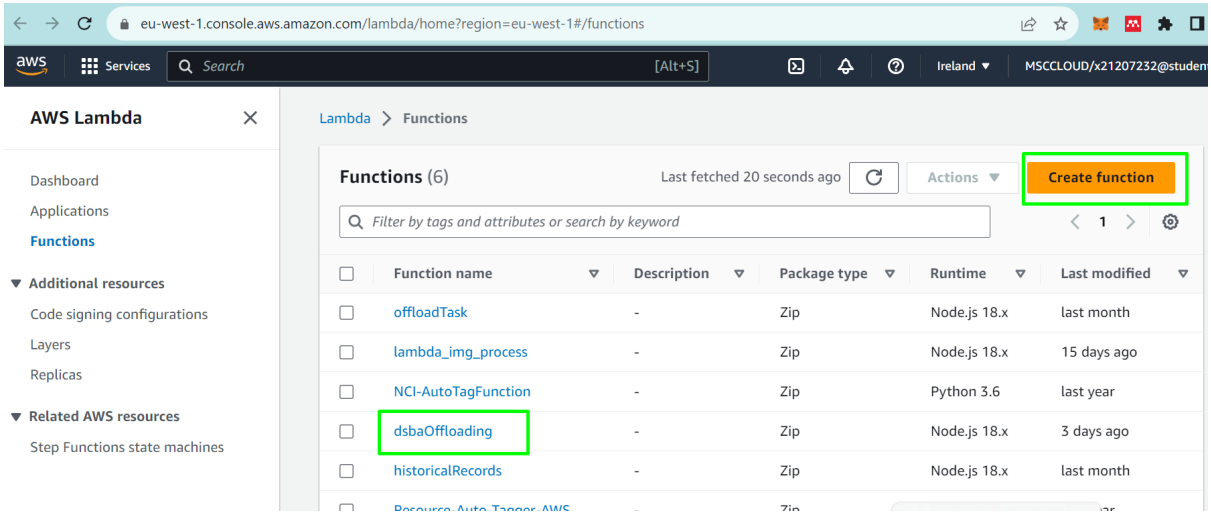


Figure 6: AWS Lambda Function

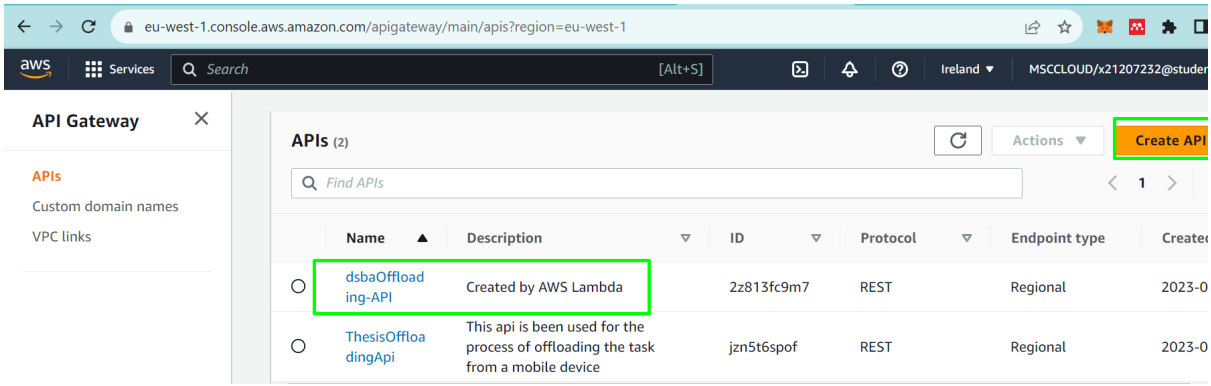


Figure 7: AWS API

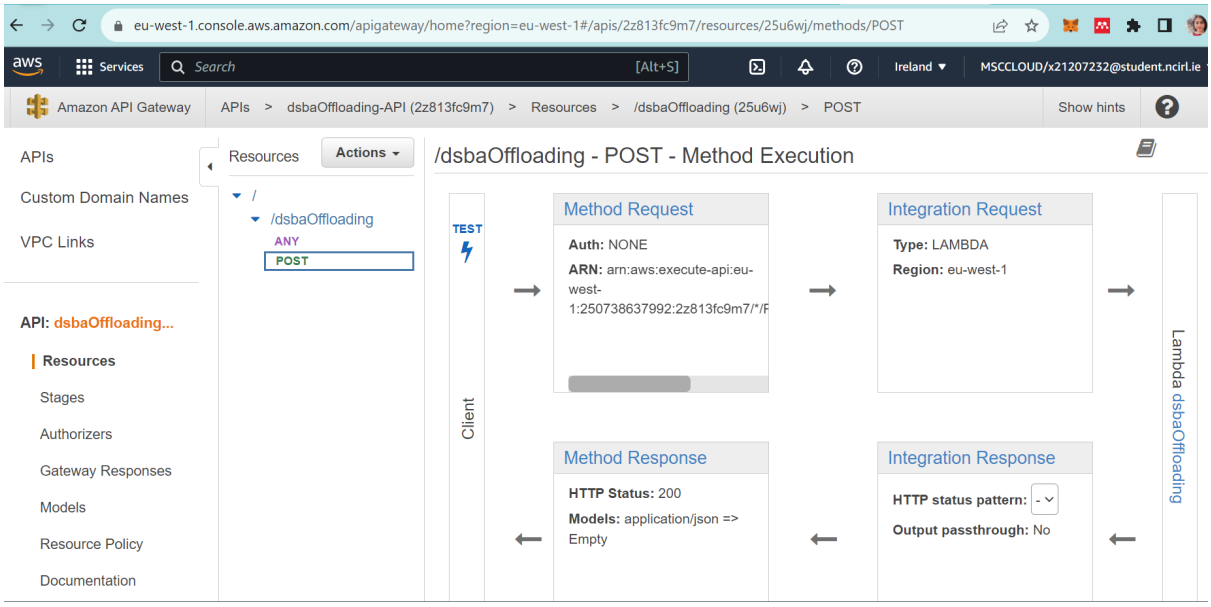


Figure 8: AWS API Gateway

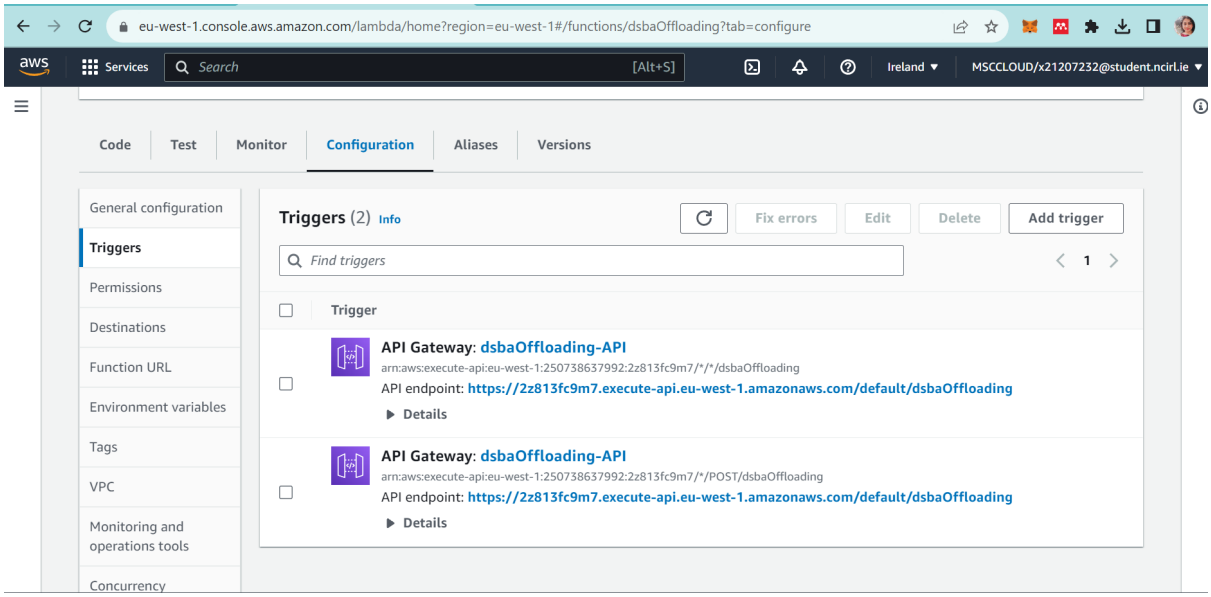
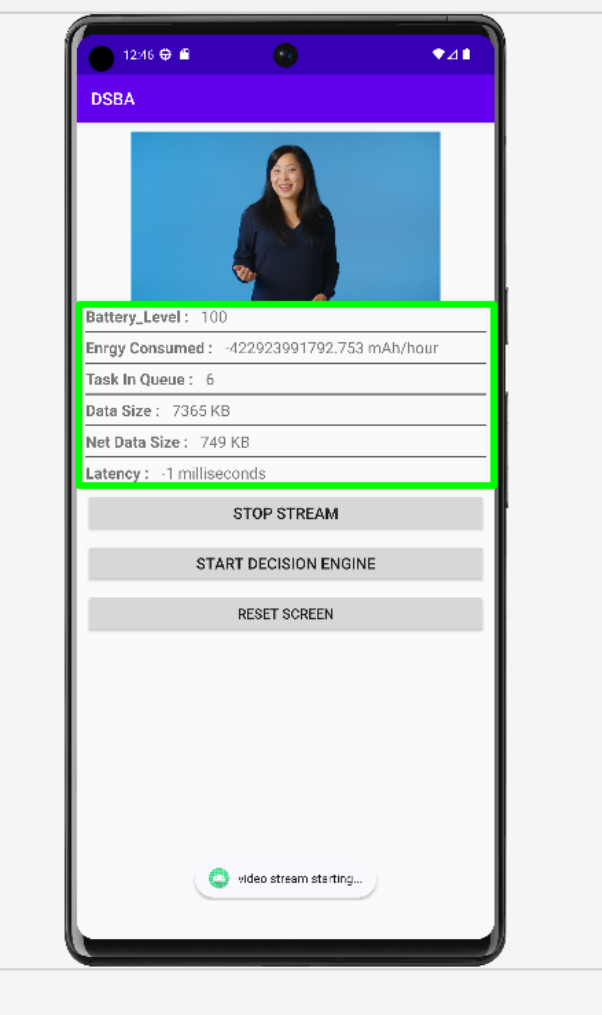


Figure 9: AWS Lambda Configuration

  - Choose the regional endpoint type
  - Link this API with the lambda function shown in figure 7
  - The link obtained under this needs to be updated in the android java class under the StreamVideoActivity to integrate

- On clicking Start stream under the app will play the HD video and tester can click on the decision engine to let the Algorithm determine the offloading as shown in the below figure.



References

[1] https://github.com/laks18/DSBA_Laks/tree/master

[2] https://docs.aws.amazon.com/lambda/latest/dg/welcome.html

[3] https://docs.aws.amazon.com/lambda/latest/dg/API_Reference.htm

[4] https://console.firebase.google.com/project/dsba-34d11/firestore/data/~2FCloudoffload~2Fcloud%20Call%20at%202023-08-01%2017:32:52

[5] https://docs.aws.amazon.com/lambda/latest/dg/services-apigateway.html

[6] https://developer.android.com/codelabs/basic-android-kotlin-compose-first-app#0

[7] DME:TechniqueforcomputationoffloadinginMobileCloudComputing.

[8] https://youtu.be/kCZSJgdq28Y