A Report
On

Development of a complete IoT/SaaS (Internet of Things/Software-as-a-Service) Solution - from the Sensor to the Cloud for Glocol Networks.

Submitted By:

P.T.S.Subhash Reddy 2017A7PS0228H

AT

Glocol Networks,
California.

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

(December 2020)

A Report
On

Development of a complete IoT/SaaS (Internet of Things/Software-as-a-Service) Solution - from the Sensor to the Cloud for Glocol Networks.

Submitted By:

P.T.S.Subhash Reddy 2017A7PS0228H

Prepared in fulfilment of the

Practice School-II

AT

Glocol Networks,
California.

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

(December 2020)

ACKNOWLEDGEMENT

My first experience with Project has been successful. I wish to acknowledge all of the people at Glocol Networks. However, I wish to make special mention of the following.

First of all, I am thankful to our project guide **Dr. Harsh Verma** under whose guideline I am able to work on our project. I wholeheartedly thank other colleagues for giving me their valuable time and attention, also providing me a systematic way to proceed forward in my project.

I am very thankful to our PS faculty coordinator **Prof. Viswanathan Hariharan** for his constant guidance, encouragement and support throughout the course of PS-II.

I am also very thankful to Glocol Networks for giving us the opportunity to work on this project.

Abstract Sheet

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI
(Hyderabad)

Practice School Division

Station: Glocol Networks **Centre**: California

Duration: 5 months **Date of Start:** 12/08/2020

Date of Submission: 08/12/2020

Title of the Project: Building a Complete IOT/SaaS solution

P.T.S.SUBHASH REDDY 2017A7PS0228H B.E COMPUTER SCIENCE

Mentor in PS Station: Dr Harsh Verma, Co-Founder

PS Faculty: Prof. Viswanathan Hariharan

Project Areas: Software Development

Abstract: The project is on building a web application and a complete IOT/SaaS solution using raspbian devices for Glocol Networks which fulfills their customer(a US based train company) needs using ReactJS/AngularJS in the front end, and DynamoDB as the database. The project involves setting up greengrass on AWS, raspbian devices and lambda functions in raspbian devices for non cloud based operations and offline situations.

There are 2 web applications, one is developed which is used for inputting data to DynamoDB for supervised learning phase and other for analysis of data.

Signature(s) of Student(s)
P.T.S.Subhash Reddy
Date: 08/12/2020

Signature of PS Faculty

Date

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
PILANI (RAJASTHAN)

PRACTICE SCHOOL DIVISION

RESPONSE OPTION SHEET

Station: Glocol Networks Center: California

ID No. & Name: 2017A7PS0228H & P.T.S.Subhash Reddy

Title of the Project: Building a Complete IOT/SaaS solution

Usefulness of the project to the on-campus courses of study in various disciplines. Project should be scrutinized keeping in view the following response options. Write Course No. and Course Name against the option under which the project comes.

Refer Bulletin for Course No. and Course Name.

Code No.	Response Option	Course No.(s) & Name
1.	A new course can be designed out of this project.	No
2.	The project can help modification of the course content of some of the existing Courses	No
3.	The project can be used directly in some of the existing Compulsory Discipline Courses (CDC)/ Discipline Courses Other than Compulsory (DCOC)/ Emerging Area (EA), etc. Courses	No
4.	The project can be used in preparatory courses like Analysis and Application Oriented Courses (AAOC)/ Engineering Science (ES)/ Technical Art (TA) and Core Courses.	No
5.	This project cannot come under any of the above mentioned options as it relates to the professional work of the host organization.	Yes

Signature of Student
Date:08/12/2020

Signature of Faculty
Date:08/12/2020

TABLE OF CONTENTS

Background of the Company

California based Glocol Networks is a startup company that is reinventing Seamless Mobility, IOT(Internet of things) and Smart City infrastructure components, exploring how emerging Transportation Data, Technologies and Applications can be integrated with existing systems for a smarter and better-connected future.

With the fast deployment of mobile technologies, the world is gearing up for IOT which will use Sensors, DSRC and RFID technologies to connect objects that can sense and communicate to us on our mobile devices. These "things" of the real world shall seamlessly integrate into the virtual world, enabling anytime, anywhere connectivity. In 2010, the number of everyday physical objects and devices connected to the Internet was around 12.5 billion. Cisco forecasts that this figure is expected to reach a number of 50 billion by 2020. Hence the company is trying to create a smart modular city platform which provides a modular solution set for smart transportation, smart infrastructure and a vision for transformation, including technologies and applications that consolidate infrastructures, support current and future systems, and improve service delivery.

Glocol Inc, has Intel, Toyota, Inrix, Leidos, Siemens, UC DAVIS, Berkerley (California universities)

Services Provided by the Company:

- **Product-Framework-Services:**

  Their services are designed and developed around their Product-framework-Services Commercialization concept which was spun off with a futuristic approach with seed funding from Cisco Systems – Cisco Applied R&D and early institutional partnership with UC Berkeley and current collaboration with UC Davis.

- **Rich Experiences drive Services:**

  With the rich experience of founders and partners, technological competence, vision and operational excellence and continuously evolving new partner relationships we are developing a complete suite of services for IOT, Mobility and Smart City vision elements.

Company Products (Glocol Smart City Fabric):

- Connected Vehicles
- Crosswalk Networks
- Intelligent Data Fusion
- Urban Analytics

During our initial days we were told to get familiar with AWS, Greengrass, Raspbian OS, ubuntu and Python programming for ML.

Objective of the project

The company Glocol Networks is still in its early stage, our work is to help along with other members of the company to create a complete IOT/SaaS solution (first using AWS, next with Azure) for the needs of its client and provide all the products which we have planned earlier(which includes a web interface for monitoring the train) to our clients and customers. The current main goal is to make an interface and a backend for clients to know the position of their trains and no.of connected devices to their network and verify it manually and improvise the learning model accordingly in order to predict the number of people boarding or present in train at a given point in time. The obtained interface should also be giving an insight on anomalies in the data so that swift decisions could be taken by the client for their business model.

Technologies Used:

- Usage of front-end Framework like **AngularJS** and **Reactjs** which makes the components reusable.
- For the database, the company uses **DynamoDB** to store all the data which is required to run and analyse the business.
- **OpenSSH** for connecting to Raspbian devices remotely and setting them up for their respective jobs through greengrass core and lambda functions.
- **AWS Cli** for testing correct extraction data from DynamoDB.
- Many python libraries such as **boto3, pandas, numpy, matplotlib, pyyaml, csv, os, datetime, geopy, pprint, scipy, unittest**.
- For testing of API's **Postman** is used.
- For analytics **QuickSight** is used.

Brief Intro on ReactJS:

React is an open-source, front-end, JavaScript library for building user interfaces or UI components. React can be used as a base in the development of single page or mobile applications. React applications are usually built around a single HTML element. We use components to tell React what we want to see on the screen. A react application is made of multiple components, each responsible for rendering a reusable piece of HTML. React uses a declarative paradigm which makes it easier to reason about our application and aims to be both efficient and flexible.

Brief Intro on AngularJS:

AngularJS is a JavaScript-based open-source front-end web framework mainly maintained by Google to produce static web pages with ease. AngularJS lets you extend HTML vocabulary for your application. The resulting environment is extraordinarily expressive, readable, and quick to develop. It is fully extensible and works well with other libraries. Every feature can be modified or replaced to suit our unique development workflow and feature needs. Angular is designed to be testable so It encourages behavior-view separation, comes pre-bundled with mocks, and takes full advantage of dependency injection.

Brief Intro on DynamoDB:

DynamoDB is a scalable, hosted NoSQL database service provided by Amazon with the facility to store the data in Amazon's cloud.DynamoDB uses tables, the core components – items and attributes. In DynamoDB, a table consists of a collection of items, and that each item is a collection of attributes. Primary keys are used to identify uniquely that is each item in a table and also secondary indexes in DynamoDB to provide more flexibility on querying the same.Security in DynamoDB is more secure and is generally provided by the available AWS security measure. To get access to DynamoDB, access should be granted with a special feature that is IAM through access/secret key pair and also it can be achieved by roles from the machine the code is running on.

Brief Intro on AWS Cli:

The AWS Command Line Interface (CLI) is a unified tool to manage your AWS services. We can control multiple AWS services from the command line and automate them through scripts.

Brief Intro on OpenSSH:

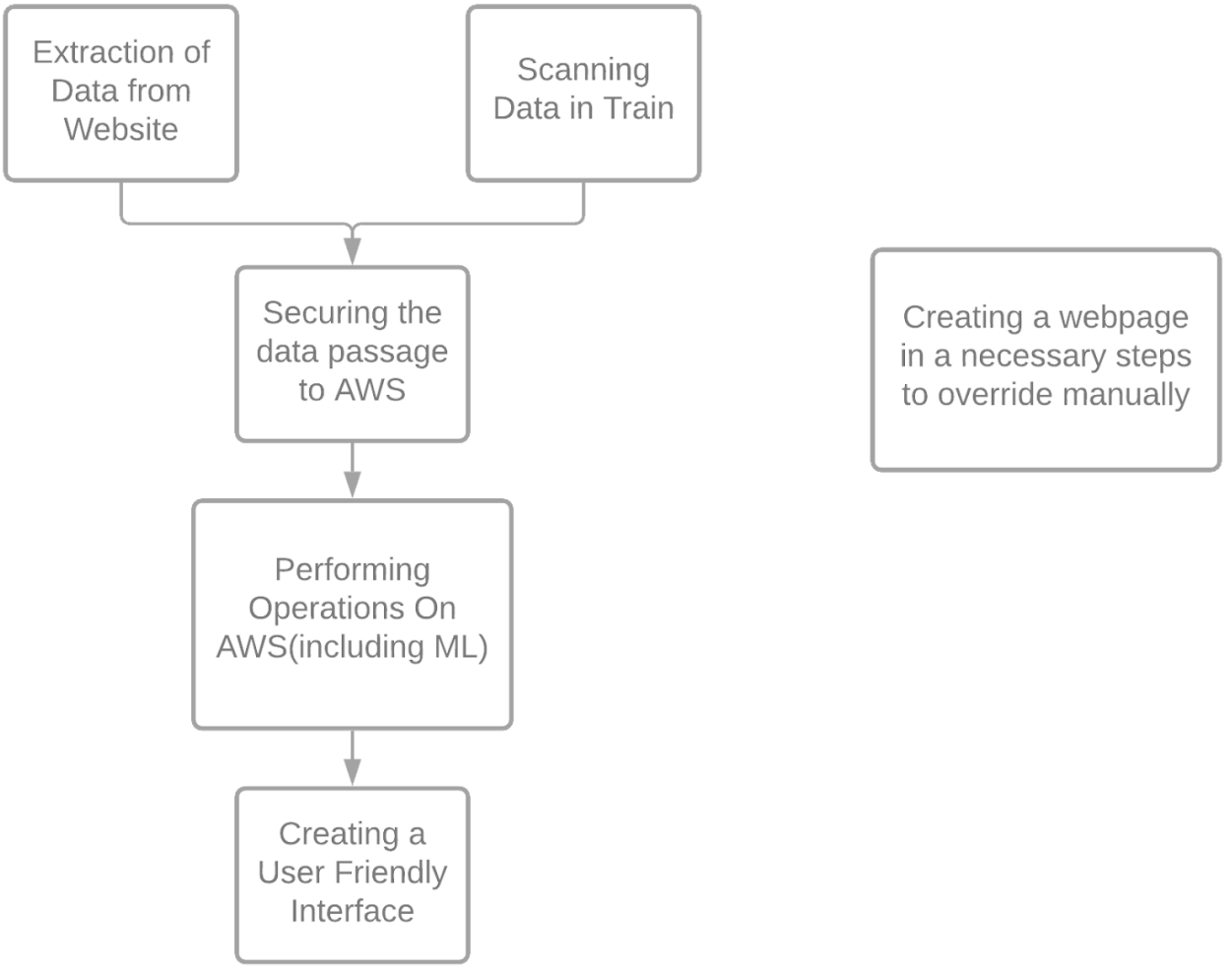
OpenSSH is the open-source version of the Secure Shell (SSH) tools used by administrators of Linux and other non-Windows for cross-platform management of remote systems. It is baked in Windows 10 1904 update and can be initialised from cmd to ssh to remote devices.

Brief Intro on Postman:

Postman is an Automated API testing tool used by many companies worldwide. For validation of API, on receiving a response, Postman validates the response as described in the test scripts.. Most interesting part is a JSON response can be parsed to an array and then the elements can be accessed by index and value or even be iterated.

The main benefit of using postman is that the user does not need to create a full JSON request programmatically unlike other automation API frameworks to put assert on it.

WorkFlow of Project



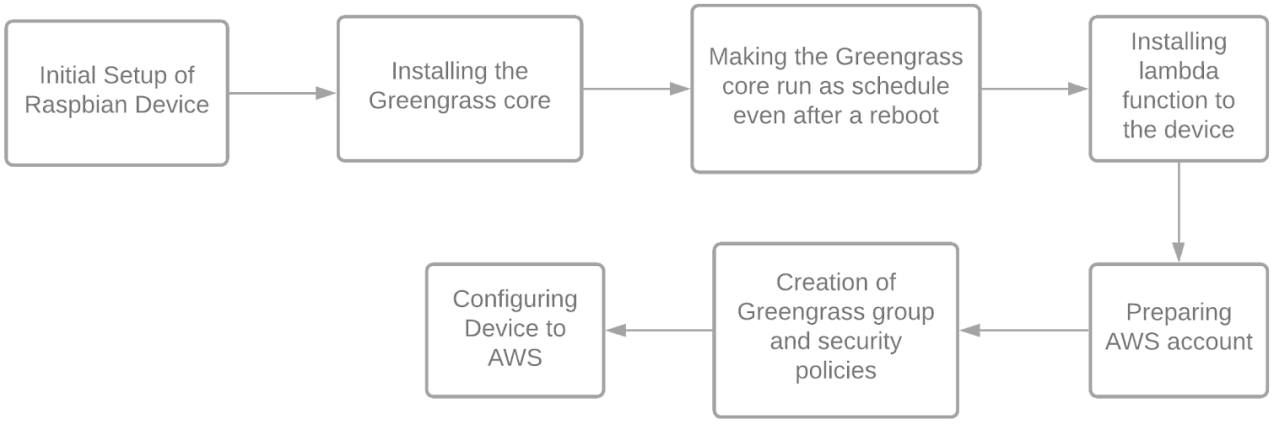
Each part of the above flowchart is done by a particular member of the project team. And every essential step had to be made automated hence a webpage for overriding data is created for all essential steps. And for continual projects after our departure necessary documentation was made so as to help the future members of the project. And most of the testing done till date is done internally and by each individual component wise.

Methodology Used for different Components

Obtaining the train schedule :

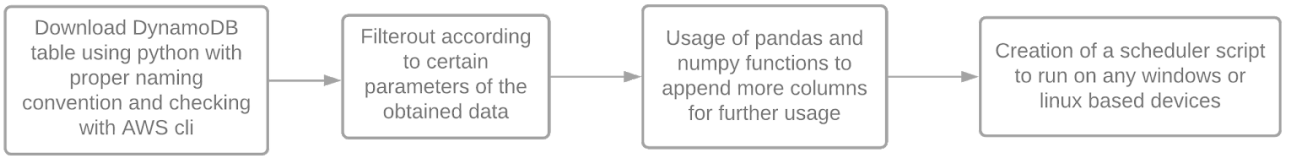
- The primary task given to me was to make a program for downloading the Train schedule data from their website and create a user friendly interface to the data.
- This was achieved by using a python script which when ran downloaded the data from their website and stored locally.
- Data was reviewed through a local server on a React program and it was thoroughly tested with Postman API.
- For different formats of data such as JSON, XML different testing was used in Postman so as to verify the data's accuracy through the process.

Configuring the raspbian devices with AWS Greengrass:



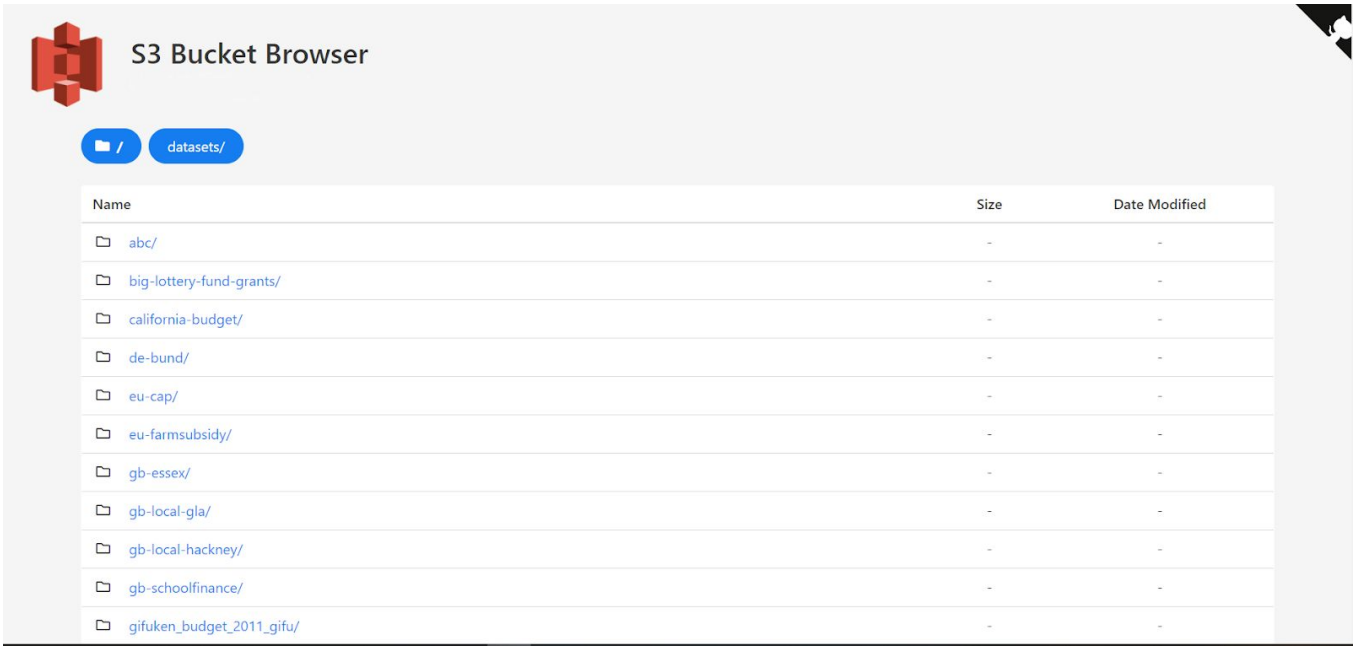
- This was a setting up of PeopleSense Devices so they send data from the train to AWS servers.
- This included installing greengrass core software in peoplesense devices.

Getting Data to Filtered state and making backup:



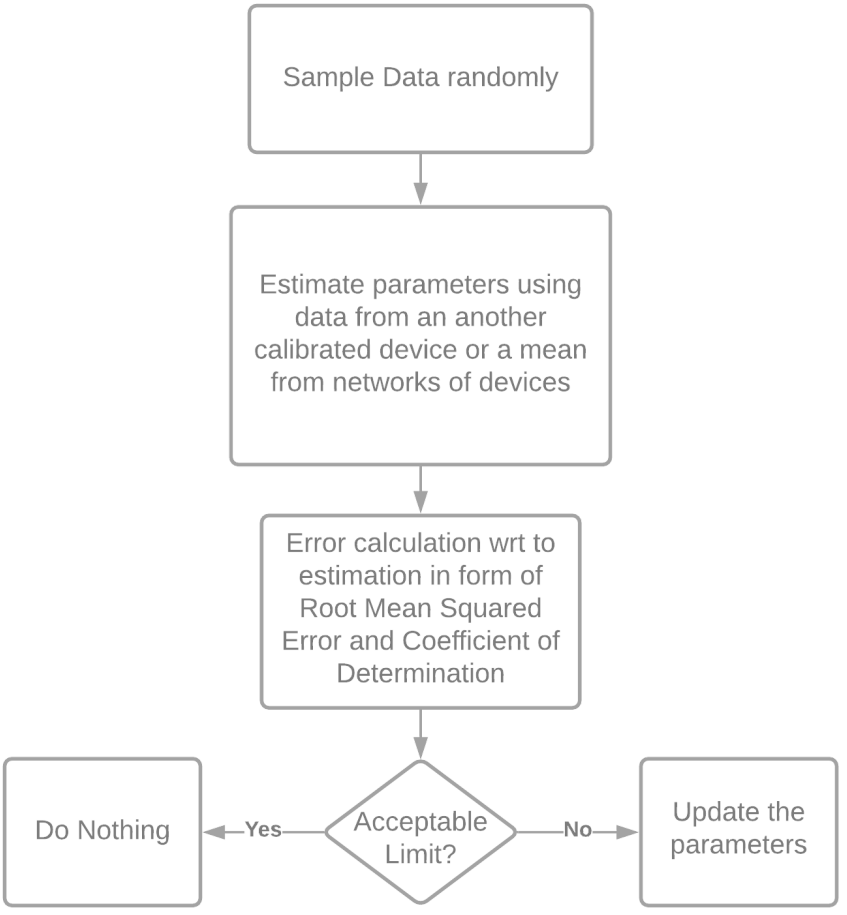
- The process involves getting data from dynamodb and testing the data using AWS Cli table downloads.
- Next certain operations are made to filter the data as required.
- And the data is uploaded to S3 bucket through a scheduling script which is in the process of being automated to lambda function.

Creation of a S3 bucket website:



- Above screenshot is a S3 bucket hosted webpage (with a non project related dataset) written in Angular so that the data can be verified from anywhere securely.
- The webpage also includes alphabetical filter and regex search on the bottom of the page.
- A color selection for light and dark theme will be added in future and the groundwork is on the code in s3 bucket.
- This webpage is thoroughly tested using Postman's automated GET request.

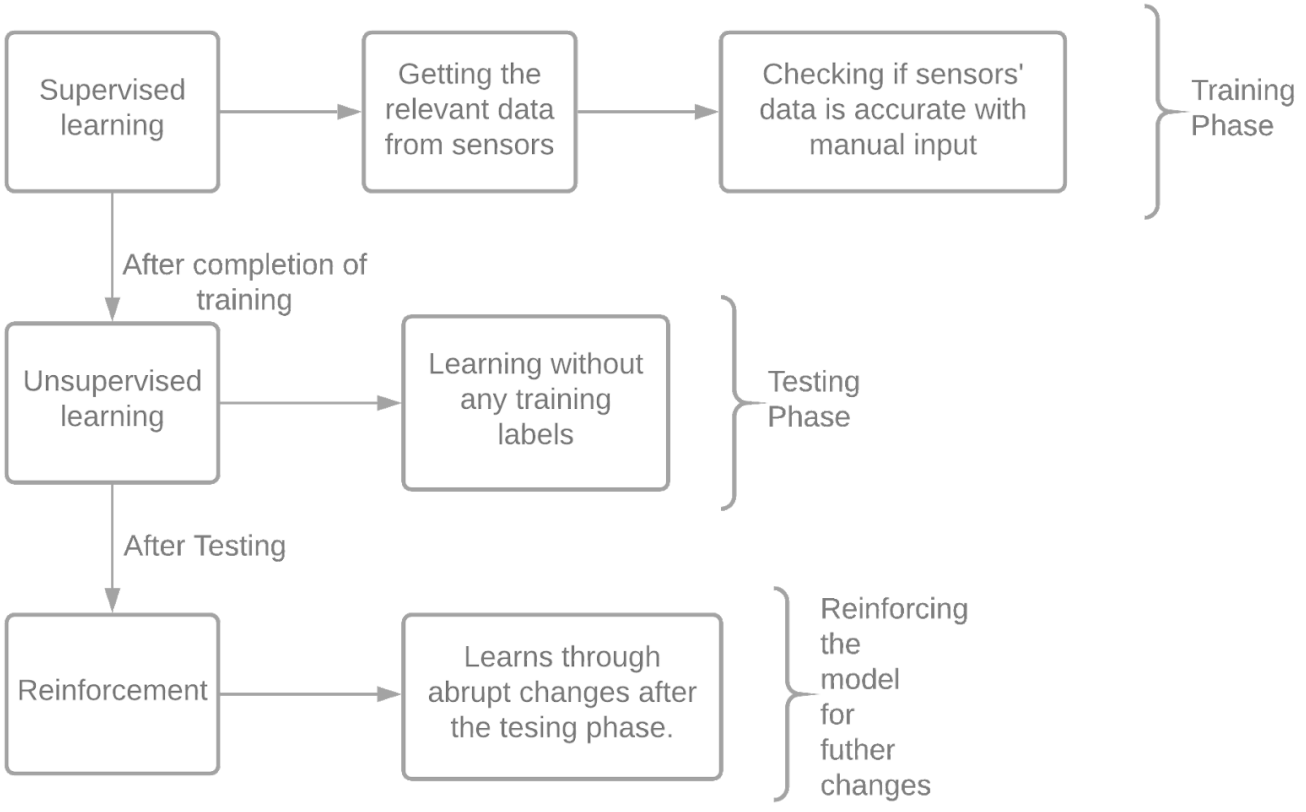
Calibration of Sensors for errors (Mostly will be incorporated end of PS on Quicksight):



- The above chart is yet to be fully implemented by the team.
- But still the idea is that given the data size it is possible for some valid errors to be prone, these are detected and highlighted but not removed from the dataset so as to leave the decision to the client regarding these data instances.
- The working of this process will most likely be done with help of AWS Sagemaker or we will resort to a local hosted server for the computations for the development period. All the changes made will be reflected in Quicksight.

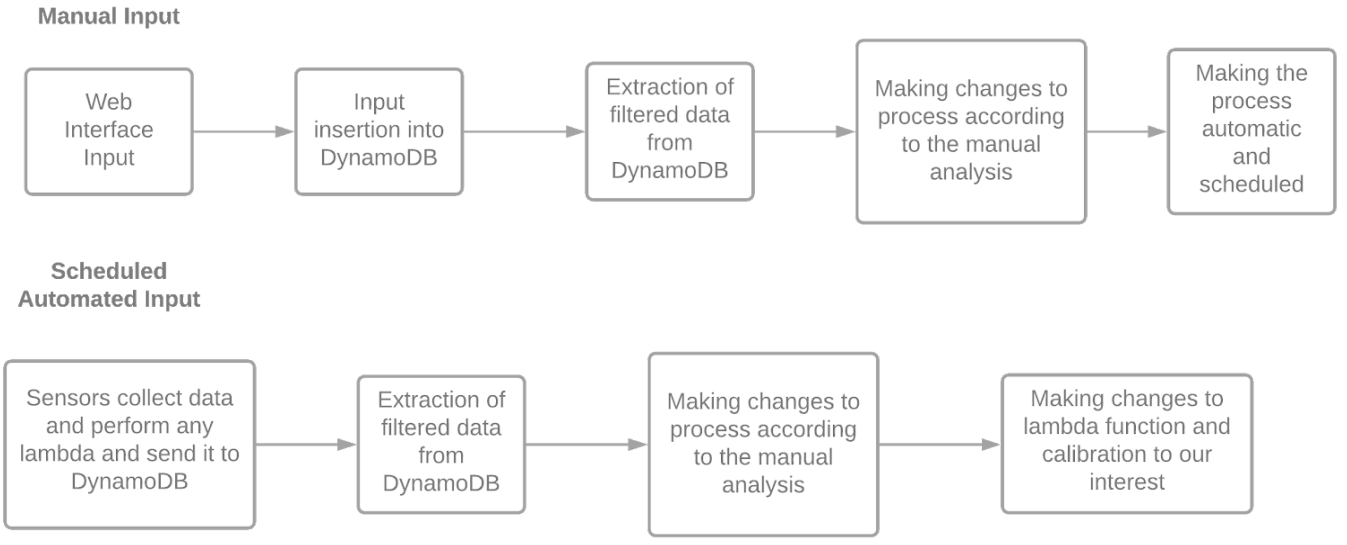
Technical Architecture of the solution

The core for the project (Motion Detection and Scanning purpose):



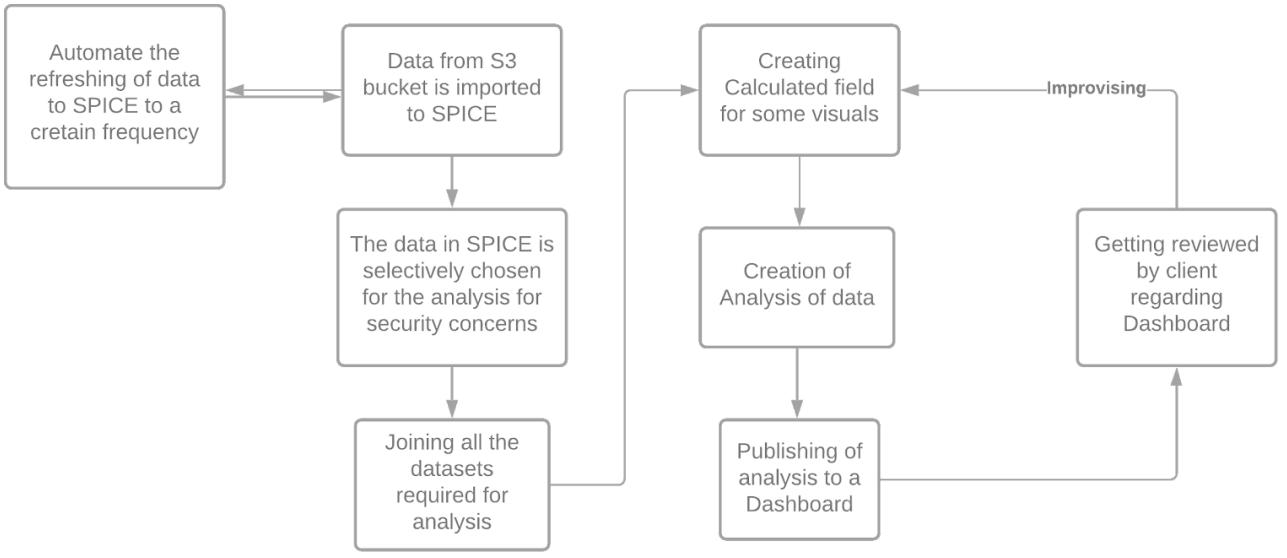
- The above LucidChart represents the core process of the project till discussed and done.
- The process is similar to any Machine Learning algorithm, first it trains from a subset of data (in this case total data) and tests with remaining data(in this case manual input).
- Next the model uses reinforcement learning for taking note of all abrupt changes in the data and alerting in advance.

Interface:



- Manual input for the core process is taken through a user friendly web interface.
- The data is inserted into DynamoDB.
- This manual input is taken so as to validate the scan count.
- The filtering and uploading of data to s3 bucket is being made into an automated process using lambda function.
- Insights on future data are being worked on using ML algorithms and will be incorporated into AWS sagemaker which in turn sends data to Dashboard for users to analyse.

Publishing the analytics on Quicksight:



- Data from s3 is imported to SPICE of Quicksight
- Then the data is selectively filtered.
- Any additional calculated fields required are added in this stage.
- Join of datasets as required.
- Creation of analysis of data.
- Publishing the analysis on a dashboard.
- Getting reviewed by clients and improvising the dashboard.
- Anomaly detection is to be added and is being currently worked on.

Outcome of the Project

The project is yet to be fully completed, and the results are confidential at the current moment as per the company wish.

Key Challenges Faced

- The secure access to raspbian devices needed VPN which was giving higher ping due to location of device and the connection was being reset during the initial days.
- SSH to the Raspbian devices was problematic at first but we found a work around till the issue was resolved.
- The GTFS files for particular month weren't upto the mark by the client so we had some issues in displaying the files in a web interface.
- The extraction of data from DynamoDB to required format took a lot of time due to the requirement of a python script rather than a AWS cli command since the process needed to be automated.
- Appended columns in filtered files of some functions led to disturbance of scheduled script created.
- Installing of core greengrass software and to make it run all the time regardless of system state created problems first.
- Whenever automated testing for S3 bucket hosted site led to always gave 100 percent accuracy which was rectified by changing the parameters of testing Postman to be more specific.
- Creating a dynamic reference line based on data filters in Quicksight turned out to be a problem which was solved with some help.

Key Learnings

- Learned about the inbuilt SSH in windows.
- Learned about the AWS IOT, Greengrass, Cli.
- Learned how to automate script with another script to make it run across all platforms.
- Learned to extract data from the DynamoDB and filter it according to our need and build a backup.
- Learned ReactJS and AngularJS in the process of making interfaces for GTFS files.
- Learned about Intel's OpenVINO toolkit for deep learning
- Learned about Postman in great detail and its versatility in API testing.
- Learned about Quicksight in AWS and different components and limitations of it.