Attendance Monitoring Using Amazon Web Services (A.W.S)

Sai Kishan Kranthi Kumar V[1], Manoj Ram Kumar K[2], Gnanapriyanka G[3], Sandhya Rani G[4]
[1,2,3,4] *Department of Electronics and Communication engineering, Aditya Institute of Technology and Management, Tekkali*

***Abstract*- Attendance has always been a conventional thing, right from the beginning at every sector. An Attendance monitoring system is an essential part to maintain a quick and accurate record of the user's attendance and provides timely summaries and records when it's needed. Using an Attendance monitoring system in our firm eliminates the need for manual attendance which is a long and cumbersome process. This system is faster and more accurate than a manual attendance system, avoids incorrect data entry, duplication and helps reduce mundane paperwork. An Attendance monitoring system always requires a physical server but for the first time we are going for virtual server. The proposed RFID based attendance system uses physical hardware for capturing the student data and sends it to the Amazon Web Service (A.W.S) a Cloud Computing platform thus sends the email attachment containing student data on a scheduled time (5 P.M) from Monday to Friday. The designed system has an access for Cloud Computing which makes the Attendance Management even more flexible.**

***Index terms*- Amazon Web Service, RFID Reader Module, ESP32 Module, LCD (inbuilt I2C), RFID Cards**

I.INTRODUCTION

Cloud computing is an emerging field and it has been growing rapidly over the past decade. The concept means that anything that can be hosted on the Internet, i.e., resources/services/data is available for use, when needed, for the composition and provision of more sophisticated services [1]. From the last few years, a lot of cloud computing providers are also offering IoT services. A recent communication infrastructure of IoT has envisaged the future where everyday life will be based on intra communication of transceivers, microcontrollers, digitally communicating via stacks of suitable protocols [2]. Many cloud providers offering IoT services including Amazon Web Services, Microsoft Azure, Google and IBM etc., AWS is very attractive for small IoT business applications because they provide large computing capacity quicker and cheaper [3]. The proposed works com-bines the AWS IoT with RFID technology for attendance monitoring. RFID (Radio Frequency Identification) is wireless automatic identification technology that is gaining attention and is considered by some to emerge as one of the pervasive computing technologies in history [4]. RFID based attendance monitoring system based on AWS (Amazon Web Services) is used to monitor the student's attendance through the proximity access RFID cards. They serve for both identification and for time entry. The system assigns a unique RFID card number for each individual. The RFID reader reads the data and stores the data in cloud via Microcontroller. As RFID sys-tem is passive type, the information on RFID cards is programmed so that the card must brought in proximity to the reader module. Once the information is gathered, it needs to be stored in a Cloud. Since we usually need a server, hereby we are using a virtual server platform I.e. Cloud Computing. Amazon web service (A.W.S) helps us to store and analyze the collected information and the email attachment will be send to the recipient on a scheduled basis. Using virtual service cloud platform, we can access as many resources as we need, almost instantly, and only pay for what we need. Such that the final data of the student is stored in AWS cloud and processed securely.

This paper is organized as follows: Section II describes the electronic components used along with experimental setup and its relevant block diagram, Section III describes about the Amazon web services and how those services are used in this project,

Section IV shows the obtained results, Finally Section V presents our conclusions.

II. HARDWARE

A. RFID Reader module and Tags:

Radio Frequency Identification (RFID) is one of the wireless technologies that use the detection of electro-magnetic signal as identification [5]. This technology is a non-contact type similar to that of bar codes and magnetic strips.

1. RFID TAGS:

This consists of an antenna and a chip for storing the data. The used cards are Passive RFID cards. The passive RFID card doesn't need any external power. The required power is drawn from the electromagnetic field generated by the RFID reader for the operation.

2. RFID READER:

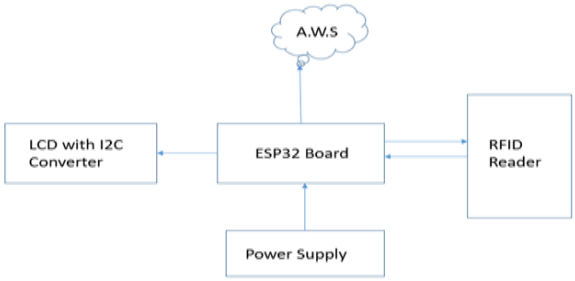
RFID reader module is a radio frequency transceiver. It reads and writes the data in to the RFID cards. The used RFID reader module is EM-18 with 125 KHz frequency range. The RFID doesn't need any physical contact, thus they are scanned against their corresponding RFID reader module.

3. ESP32 DEVELOPMENT BOARD:

ESP32 is a low cost SOC (System-on-chip), widely used in IoT based projects. It is an advanced successor of the 8266 chip primarily in the implementation of two cores clocked in different version up to 240 MHz [6]. This has both the Wi-Fi and Bluetooth capabilities, which make it an all rounded chip for the development of IoT projects and embedded systems.

4. LIQUID CRYSTAL DISPLAY with I2C Converter:

An LCD is used to display the information which is retrieved from the RFID cards when scanned over the reader module. LCD is used along with attached I2C converter to convert SPI communication to I2C communication thus obtaining the output with minimum wires (In case of I2C there are only two wires SDA, SCL for communication). The components of the project are ESP32, RFID reader module, RFID cards and an LCD display. The RFID reader module EM-18 consists of 4 pins: VCC, TX, RX and GND which runs on 12V.The communication between the reader module and micro-controller is implemented using UART protocol. The RFID reader reads the information from the card and transmits to the host i.e.ESP32 board. The TX pin of the reader must be connected to RX pin of ESP32. An LCD is used to display the information that is retrieved from the RFID card. The general block diagram is shown in the figure 1.



III. AMAZON WEB SERVICES

A.W.S has so many services which enables that the user's data to be encrypted and can be utilized in a manner that they wants. The services used in the proposed paper are:

1. AWS IoT Core:

AWS IoT Core is the most customizable and closest to integrating the desired characteristics [7]. In the pro-posed paper, IoT core is useful in direct communication between ESP32 board and AWS Cloud in MQTT (Message Queue Telemetry Transport) format. It is also helpful in connecting with other services when a Rule is created.

2. AWS DynamoDB:

Amazon is using Dynamo Databases, which is a fully managed NoSQL database that provides predictable and super-fast performance with unified scalability [8]. The incoming MQTT messages from IoT core are stored in a table named as 'Attendance'.

3. AWS Glue:

Glue consists of a crawler and acts as a pipeline in exporting or importing data from one database (Source) to other database (Target). In this proposed paper, DynamoDB is taken as source and S3 bucket is taken as Target.

4. AWS Lambda:

Lambda is a serverless computing platform that runs in response to events configured either internally or exter-nally. Lambda service is used in this proposed paper to fetch the data from S3 bucket and sends Email to the recipient.

5.AWS SES (Simple Email Service):

SES is useful in sending emails through cloud to the verified recipients. In this proposed paper, this service has been used for sending student data in the form of an email attachment.

The workflow for the proposed project would be:

- Establish a connection between the ESP32 to the computer and connect the RFID reader, LCD display to the ESP32 through wires.
- Run the code on ESP32 board to publish the MQTT messages to AWS IoT core.
- Now after linking ESP32 module with AWS IoT core, a 'Rule' should be created to link IoT core with the DynamoDB service.
- An AWS IoT 'Rule' consists of an SQL SELECT statement and a 'Rule act'.
- Micro controller sends the information to AWS IoT by publishing the messages in MQTT format.
- The SQL SELECT statement allows us to extract the data from an ordinary incoming MQTT message. The Rule action is triggered when MQTT message is received on a part that matches the topic filter.
- This rule action allows us to take the information extracted from the MQTT message and inserts the mes-sages in DynamoDB service.
- Using AWS Glue, the crawler on scheduled basis has been instructed to fetch the data from DynamoDB and exports it to S3 Bucket in the CSV Format.
- Finally, the lambda service which also being scheduled, fetches the CSV file from S3 bucket and using SES service sends the mail with CSV attachment to the authorized recipient. AWS IoT rule actions are used to specify what to do when a rule is triggered to it.

A. AWS IoT Core to DynamoDB:

Initially, the THING has been created in IoT core console with required certificates and policies. When the stu-dent scans his card to the device, the RFID reader reads the data and sends it to the ESP32 board for processing. ESP32 then sends the student data (Student name, Roll number) to A.W.S IoT core by updating the THING shadow as shown in figure 2. Once the shadow is updated, a RULE has to be created in a way that the incoming data needs to be inserted into the DynamoDB table thus creating a virtual database as shown in figure 3.

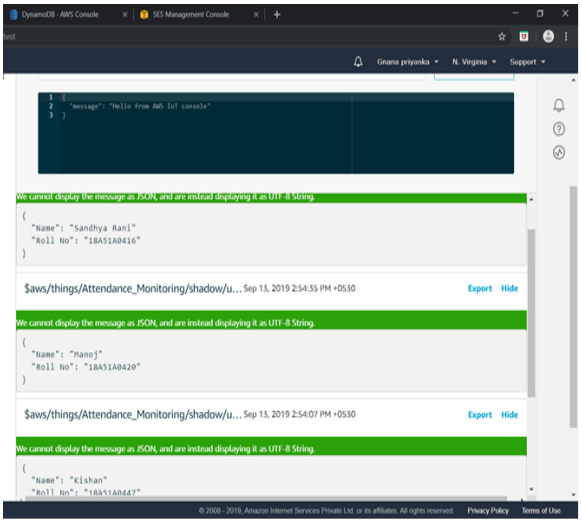


Fig 2. IoT core Thing Shadow Update

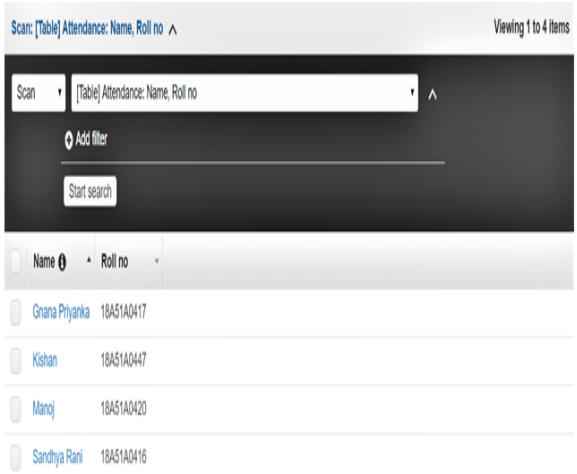


Fig 3. Data Inserted into DynamoDB Table

B. AWS Glue:DynamoDB to S3 Bucket:

The data from DynamoDB needs to be in the form of CSV file for sending the email as attachment. For this to accomplish, A.W.S glue helps in exporting the DynamoDB table to S3 bucket in our preferred format (CSV).

Now to export data from DynamoDB to S3 Bucket using AWS glue:

- Create a Database in AWS Glue.
- Create a table using Crawler.
- Create an ETL Job for mapping DynamoDB table to target S3 bucket in CSV format as shown in figure 4.

After this glue process is done the respective data will be automatically attached to the S3 bucket as shown in figure 5.

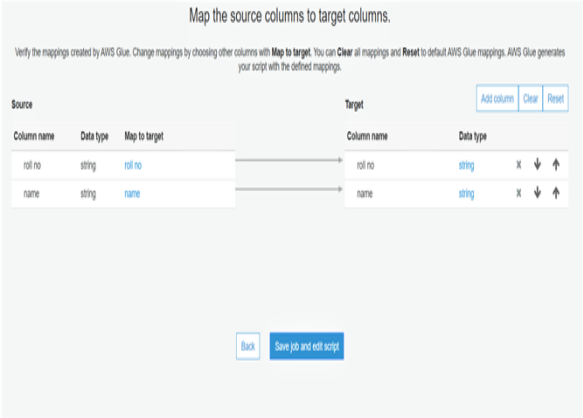


Fig 4. Mapping of DynamoDB (Source) to S3 (Target)

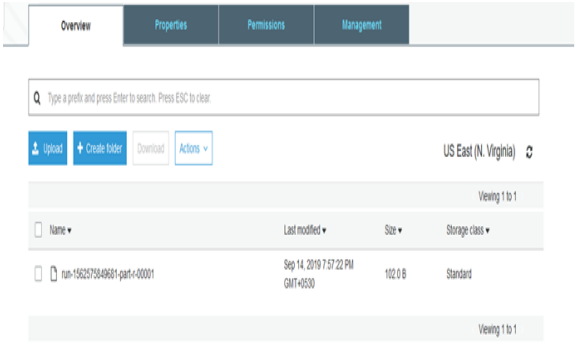


Fig 5. Exported DynamoDB data (CSV File) to S3 bucket.

C. EMAIL SENDING USING AWS LAMBDA

Lambda function which helps to fetch the data from the S3 bucket and send an E-mail to the respective recipi-ent. For this the procedure is:

1. Install the required modules in the local host machine, zip the module along with the code written in Node.js and upload it to the lambda console.
2. In order to have a scheduled functioning of lambda, create an event in the Cloudwatch.

By pushing the event notification to cloudwatch with target as the created lambda function, then lambda fetches the CSV file containing the student data from the S3 bucket. Once the fetching is done the E-mail will be sent through to the respective recipient and gets the acknowledgement in the lambda console as shown in the figure 6. For the lambda to fetch data from S3 at a specified time (5 P.M) every day from Monday to Friday, an event is created in the cloudwatch by making lambda as target. This attaches the schedule to the lambda that responses at the scheduled event.

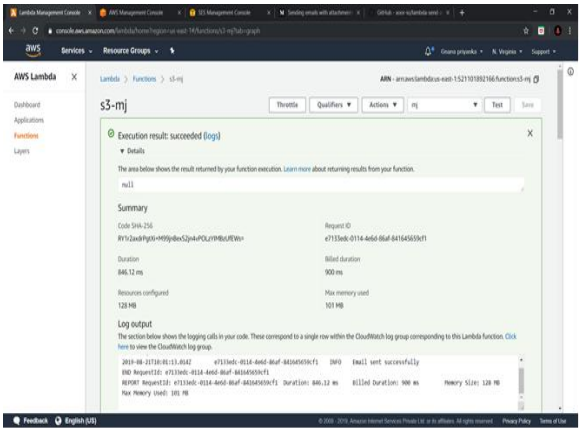


Fig 6. Email sent from lambda console

IV. RESULTS

As soon as the student scans his card over the RFID reader, the name of the student along with the roll number appears on the LCD display as shown in the figure 7. The ESP32 board then sends updates the THING shadow accordingly.

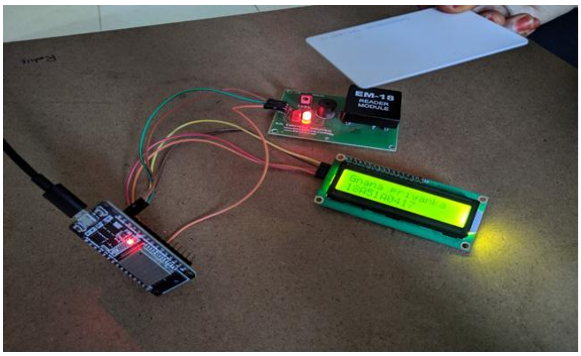


Fig 7. Hardware setup showing student name when scanned

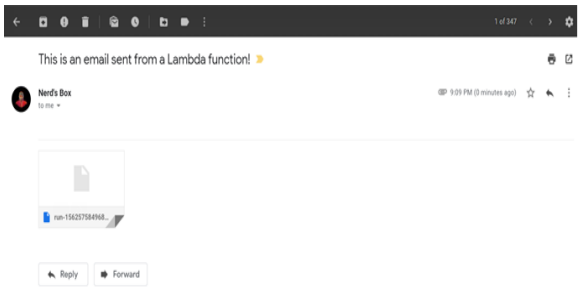


Fig 8. Email Received through SES and AWS Lambda

The proposed system achieves in not only capturing the student's data but also provides storage facility via a virtualized server platform A.W.S and the output is easily accessible to the recipient through the mail as shown in figure 8 through AWS lambda server less platform.

V. CONCLUSION

An Attendance Monitoring System is designed to manage a user's day to day attendance. Also, the proposed system provides an ease of access since there is no extra paper work. This results in achieving the usage of AWS cloud computing and RFID Interface in an innovative manner, helps us to overcome the traditional attendance.

REFERENCES

[1] Biswas, A. R., & Giaffreda, R. (2014). "IoT and cloud convergence: Opportunities and challenges". 2014 IEEE World Forum on Internet of Things (WF IoT). doi:10.1109/wf-iot.2014.6803194.

[2] Majeed, A., & Ali, M. (2018). "How Internet-of-Things (IoT) making the university campuses smart? QA higher education (QAHE) perspective". 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC).

[3] Kang, D.-H., Park, M.-S., Kim, H.-S., Kim, D., Kim, S.-H., Son, H.-J., & Lee, S.-G. (2017). "Room Temperature Control and Fire Alarm/Suppression IoT Service Using MQTT on AWS". 2017 International Conference on Platform Technology and Service (PlatCon).

[4] RFID - based Staff Control System (SCS) inKazakhstan, N Saparkhojayev 2015 J. Phys.: Conf. Ser.622 012050.

[5] K. Domdouzis, B. Kumar, and C. Anumba, "Radio-Frequency Identification (RFID) applications: A brief introduction," ScienceDirect: Adv. Eng. Informatics, vol. 21, pp. 350–355, 2007.

[6] Marek Babiuch, Petr Foltýnek, Pavel Smutný, "Using the ESP32 Microcontroller for Data Processing", 20th International Carpathian Control Conference (ICCC) 2019.

[7] N. Imtiaz Jaya, Md. Farhad Hossain, "A Prototype Air Flow Control System for Home Automation using MQTT over Websocket in AWS IoT Core", 2018 International Conference Cyber-Enabled distributed computing and knowledge discovery.

[8] S. Kalid, A. Syed, A. Mohammad, Malka N. Halgamuge, "Big-Data NoSQL Databases: "A Comparison and Analysis of "Big-Table", "DynamoDB", and "Cassandra", 2017 2nd International Conference on Big data analytics (ICBDA), doi:10.1109/IC BDA.2017.8078782.