2023 International Conference on Networking and Communications (ICNWC) | 979-8-3503-3600-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICNWC57852.2023.10127524

CLOUD BASED ATTENDANCE AUTOMATION SYSTEM WITH ANALYTICS AND REPORTING

Aayush Singh
Department of Networking and Communications,
SRM Institute of Science and Technology
Kattankulathur, India
as5294@srmist.edu.in

Pavan Sai. K
Department of Networking and Communications,
SRM Institute of Science and Technology
Kattankulathur, India
pk4981@srmist.edu.in

V. Lavanya
Department of Networking and Communications,
SRM Institute of Science and Technology
Kattankulathur, India
lavanyav@srmist.edu.in

Abstract* — Managing attendance has always been tedious for the organizations as it is time consuming and repetitive activity. Fake attendance is a major issue to be addressed. Existing bio-metric system is a decent solution but they have limited functionality and have many vulnerabilities. Our objective is to provide an efficient and scalable cloud-based solution with facial recognition which can detect fake attendance and provide alerts to the management in case of irregular attendance. Our solution follows a verification process to avoid fake attendance. Initially, students have to upload a photo with their ID cards into our application. Next, the teacher or supervisor of the class will take a group photo of all the students and upload them into cloud through a mobile application, which will act as a user interface of our solution. The cloud architecture deployed in our solution combines various services like Amazon Rekognition, Amazon Simple Storage Service, Amazon Relational Database Service, Amazon Simple Notification Service, Amazon Lambda, Amazon Textract and Amazon Amplify in the most efficient manner in terms of storage and cost by regular deletion of unwanted data. We can then use the required cloud services to compare the faces of the student's photo saved in our database with the group photo uploaded by the teacher to detect the students who are present. The proposed solution also analyzes the historical data within the database to prepare an aggregated report that consists of various fields like frequency of irregularity in attending classes, subject balance score and average attendance. While the back-end of the application is deployed in cloud, the front-end will be handled using Android as 70% of the general populace are dependent on this platform. Our solution uses Model-View-View-Model (MVVM) architecture. Execution time in MVVM applications is faster due to it supporting data binding with an average difference of 126.21ms.

Keywords: Attendance Automation System, Facial recognition, Cloud, Android, MVVM Architecture, Amazon Rekognition, RDS, S3, Lambda, Textract, SNS, Amplify, Custom label.

I. INTRODUCTION

Compiling attendance records for each student can be a time-consuming and demanding task for professors. It requires a considerable amount of effort to accurately record and calculate attendance percentages for each student [13]. The goals include maintaining records, assessing pupils and appreciating regular attendance in the classroom. Hardcopies were used for the system of manually documenting student attendance [14]. Other disadvantages to the old method are the loss or damage to the attendance sheets and time & effort wasted in documentation [9]. The proposed system represents a potential replacement for the conventional method of attendance tracking, as it offers improved simplicity, speed, and reliability, particularly in the context of the rapid advances in information technology and its widespread adoption within educational institutions. The proposed system is an Android-based tool designed to track daily student attendance in universities, simplifying the process of collecting attendance data for individual students [7]. The system is also capable of generating reports and providing insights into a student's attitude towards learning. Furthermore, the implementation of this method has the potential to conserve both human and material resources while enhancing work efficiency. [8]. In order to gather the data, we made an android application through which the students can upload their ID cards. This data will be saved in our AWS database for later use. Then, the teachers can take a group photograph of the students at the end of the class through our application which will then be uploaded into AWS S3 bucket. We then verify if the photo uploaded by students during registration, which is our source image, matches with any face in the group photo uploaded by the teacher, which is our target image.

II. RELATED WORK

Various researchers have conducted experiments in different fields to develop interfaces to allow for better systems to mark attendance. We have encapsulated several factors from each of this research to help build this application.

R. K. Kodali et al. [1] researched on the current biometric systems which are used to capture attendance. This solution was much better than the physical task of taking the attendance but still had its flaws. They discovered that despite the biometric systems using fingerprint to take attendance, people found ways to cheat the system by using fake fingerprints. This led to compromising of the system due to its lower authenticity and efficiency. To tackle this problem, they decide to use facial recognition system which will help detect such fraudulent behavior. But one shortcoming of this method is the detection of face in different areas. Since the face is recognized, the system will give attendance to the person, but people could capture the image from their home and still get the attendance. To conquer this problem, we have included custom labels in our application which will detect the background to ensure that the person is in the correct location.

Shailendra et al. [2] presented a paper which is made to design and build the framework for taking attendance in schools and colleges to prevent the tedious process of the teacher taking attendance physically in the class which is more time-consuming and less efficient. But by introducing different hardware components as well as a remote server, increases the cost and decreases efficiency by needing more space. In our solution, we have decided to use cloud server to store the data and developed a mobile application to work as its interface. This leads to us getting the pay-as-you-go service from cloud services as well as reducing costs by not needing physical structures to get this data.

P. P. Godbole et al. [3] have designed a system which is used to dial the attendance using wireless connection. This opens up a whole new ordeal for the area as this allows the authorities to record and save the attendance through the internet. Rather than relying on biometric and piercing, which are easily replaceable due to their lack human intervention and progress, the users can turn to this Wi-fi attendance system framework which utilizes IoT devices. This provides better security as well as speed in terms of organizing the data. Unfortunately, the data which is collected through this hardware and transferred over the internet are at a risk of being decrypted and hacked. It also increases the cost since physical devices are used for this process. To avoid such ordeals, we have created an android application which will replace the need of hardware to collect data. We have also used the cloud services for storage and computation which significantly reduced cost and increased efficiency.

A. G. Menezes et al. [4] considers student attendance assessment necessary for the classroom environment, despite it being a tedious and time-consuming work. They suggested utilizing facial recognition with deep one-shot learning to control student attendance and tested their method under various scenarios and with various image-capture equipment to confirm that such a pipeline might function in a practical environment. In addition, they proposed the implementation of a face detection stage using Histogram of Oriented Gradients (HOG) and a Convolutional Neural Network (CNN) with Max-Margin Object Detection-based features to enhance the accuracy of the system and mitigate the high rate of false negatives that are commonly encountered in uncontrolled settings. The proposed recognition pipeline can be designed to function with limited computational resources, such as those available on smartphones, or it can be offered as a "Software as a Service" tool, thereby promoting the effectiveness and accessibility of this approach to address the challenge of student attendance assessment. We have decided to add to this research by integrating it with cloud services and providing the analysis and reports for the registered users. This helps them keep track of all the changes.

N. Gupta et al. [5] Using the idea of face detection and recognition through open computer vision, a framework is built to handle the student's class attendance data. This approach was proposed primarily to enhance the established university attendance practices and prevent the waste of resources and time. The concept of open computer vision is used in this approach, which is entirely based on the general-purpose language Python. In order to identify faces, we employed a haar cascade system, and in order to recognize faces, we used an LBPH model. After training each individual student, the system generated a spreadsheet that included the number of students present in the classroom along with a picture or video that was captured live. Here, rather than relying on python language and algorithm or using the LBPH model, we decided to use Amazon Rekognition which fulfils the same purpose but with much higher accuracy.

D. Mijić et al. [6] One of the frequent but time-consuming jobs of teaching personnel at higher education institutions is recording student attendance. In order to reduce the administrative strain and partially automate this task, various

technical solutions are used. The majority of them make use of one or more of the following, namely cellphones, biometric tools like face and fingerprint recognition, and Radio Frequency Identification (RFID) technology. In this work, we demonstrate an enhanced version of our RFID-based system as well as some practical applications and outcomes. The hardware RFID reader and a web-based application utilized by the faculty staff have seen the most benefits. In addition to the typical components of previous systems for tracking student attendance, the system characteristics that enable integration with external systems, the recording of important information about teachers' work in class, and the generation of periodic reports on that work are also included. Instead of depending on the RFID, we have decided to use the facial recognition system which has statistically shown to have higher accuracy and efficiency.

III. PROPOSED WORK

The approach advocated in this paper entails utilizing face recognition technology to track student attendance. Figure 1 illustrates the proposed system architecture.

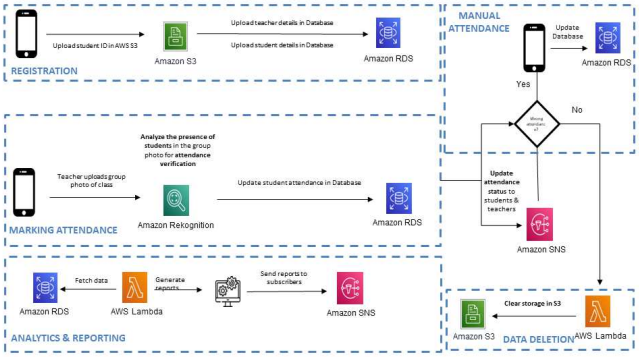

Figure 1. Architecture Diagram

The proposed methodology starts with the registration of students into the system. Following methodology has few main stages such as providing the images through an Android application, storage of the data in AWS S3 bucket, using AWS Rekognition for face detection, comparing faces and providing us with the similarity index to verify the student's attendance.

A. Registration through Android Application

First, the students and teacher will need to register through an android app. This will allow us to obtain data like their ID, name, photo, subject etc. Since we will be using the android application the interface to gather the user's data, we needed to follow an efficient architecture. Therefore, we chose the MVVM design. The Model-View-ViewModel architecture, or MVVM architecture, is based on the principle that the logical portion of the program should be held separately from the presentation data (UI). This is done to prevent failures that are encountered in alternatives, such as activities where all the data is saved in one spot, which could result in issues later down the road [11].

The MVVM architecture is divided into 3 main segments which are:

Model: This section tries to abstract all the data; in essence, it receives the data sent by the ViewModel and replies with a write response so that the UI can see it.

View: The data is displayed to the user on the UI using this layer. They can see this since it is the interactive portion of the architecture. Through this interface, the user can make modifications or take actions that are subsequently passed to the ViewModel for a response.

ViewModel: In essence, the ViewModel class stores and displays all of the application's data. It serves as the model's and view's connection. It receives user input through the view layer, passes it to the model layer, and then sends the model layer's response back to the view layer via the ViewModel.

B. AWS Services Utilized

We have used AWS Amplify for the handing of the backend of our system. It contains a large range of open-source libraries and drag-and-drop UI elements that we can utilize as the foundation for our applications [15]. We have used the integrated CLI to create our backend and set up the system. Figure 2 shows the deployment of our project.

```
? Please choose the profile you want to use AWS Amplify Tutorial Trial 1
Adding backend environment dev to AWS Amplify app: d35chmfht1ui1n

Deployment completed.
Deployed root stack AndroidAppwithAWSAmp [ ======================================== ]
        amplify-androidappwithawsamp-… AWS::CloudFormation::Stack     CREATE_COMPLETE
        UnauthRole                     AWS::IAM::Role                 CREATE_COMPLETE
        DeploymentBucket               AWS::S3::Bucket                CREATE_COMPLETE
        AuthRole                       AWS::IAM::Role                 CREATE_COMPLETE

√ Help improve Amplify CLI by sharing non sensitive configurations on failures (y/N) · no
Deployment bucket fetched.
√ Initialized provider successfully.
 Initialized your environment successfully.

Your project has been successfully initialized and connected to the cloud!
? Are you sure you want to continue? Yes
⚠ Amplify CLI now supports verifying a Cognito user email address that has been changed and will aut

Deployment completed.
Deployed root stack AndroidAppwithAWSAmp [ ======================================== ]
        amplify-androidappwithawsamp-… AWS::CloudFormation::Stack     UPDATE_COMPLETE
        authandroidappwithawsampe6a37… AWS::CloudFormation::Stack     CREATE_COMPLETE
Deployed auth androidappwithawsampe6a37227 [ ==========================================
        UserPool                       AWS::Cognito::UserPool         CREATE_COMPLETE
        UserPoolClient                 AWS::Cognito::UserPoolClient   CREATE_COMPLETE
        UserPoolClientWeb              AWS::Cognito::UserPoolClient   CREATE_COMPLETE
        UserPoolClientRole             AWS::IAM::Role                 CREATE_COMPLETE
        UserPoolClientLambda           AWS::Lambda::Function          CREATE_COMPLETE
        UserPoolClientLambdaPolicy     AWS::IAM::Policy               CREATE_COMPLETE
        UserPoolClientLogPolicy        AWS::IAM::Policy               CREATE_COMPLETE
        UserPoolClientInputs           Custom::LambdaCallout          CREATE_COMPLETE
        IdentityPool                   AWS::Cognito::IdentityPool     CREATE_COMPLETE
        IdentityPoolRoleMap            AWS::Cognito::IdentityPoolRol… CREATE_COMPLETE
```

Figure 2. AWS Amplify CLI Commands

We will put all the information provided by users through the registration page into an S3 bucket after our application has been configured with AWS and connected to its server. The AWS RDS, which will serve as our database, will get these details from this bucket. The teacher will next use our application to upload the class's group photo, which will then be transmitted to AWS Rekognition, where the service will identify every student's face. Deep learning is used by Amazon Rekognition to precisely analyses images, compare and locate faces in an image, and identify scenes and objects in photos and videos. When faces emerge in a picture, we can rapidly recognize them because to this. Additionally, it can identify each of those faces' gender, eyeglasses, facial hair, age range, and other characteristics. When comparing faces between two photos, the service records and makes use of these characteristics. This will allow us to verify students' attendance [12].

Using AWS Lambda, we can fetch data from AWS RDS which will contain the student's attendance attribute. We can generate a report with that data to showcase his/her attendance records. Once the reports are generated, we can use the AWS Lambda service to call SNS service which will help us in sending and notifying the users with them.

C. Challenges and Drawbacks

A drawback this system can face is when there are twins in the same class. This will lead to misinformation inside our database as attendance of one twin will incorporate into the other twin's attendance. Another drawback this system can face is the incorrect comparison of the photos. For these cases, we have decided to give the teacher access to Amazon RDS, so they can change the data manually if such a case arises. To make sure we do not waste resources, the access to change the data manually will be only allowed for 48 hours from the time of upload of group photo. This will give the student and teacher ample amount of time to change the data in case of an error while also avoiding any unnecessary wastage of resources. By using AWS Lambda, we can clear the storage in S3 bucket after every 48 hours of the group photo upload to free space and increase the efficiency of the system.

```python
from datetime import datetime, timedelta
from dateutil import tz
import pymysql


def lambda_handler(event, context):

    message = "Dear user, you are marked as absent today! Please check your attendance and try to be regular "
    topic_arn = "arn:aws:sns:ap-south-1:077497307037:Absent-Messages"
    snsClient = boto3.client('sns')
    max_age = datetime.utcnow() - timedelta(days=2)
    max_age = max_age.replace(tzinfo=tz.tzutc())
    bucket = event['Records'][0]['s3']['bucket']['name']
    prefix = 'target_images/'
    s3 = boto3.client('s3')
    objects_to_delete = []
    response = s3.list_objects(Bucket='pa1-test-bucket-1', Prefix='target_images/')
    for obj in response.get('Contents', []):
        last_modified = obj.get('LastModified')
        if last_modified and last_modified < max_age:
            objects_to_delete.append({'Key': obj['Key']})
    if objects_to_delete:
        s3.delete_objects(Bucket='pa1-test-bucket-1', Delete={'Objects': objects_to_delete})
        print("Deleted all target images that are older than 2 days!")
```

Figure 3. Lambda Function

D. Testing Face Detection and Comparison

We used the AWS Rekognition service, which is offered by AWS, to compare faces. It comes with a straightforward, user-friendly API that can swiftly analyze any image asset kept in Amazon S3. Using deep learning technology, it breaks down an image into many different attributes, such as open eyes, glasses, confidence, facial hair, etc., compares them with the target image, and then provides a similarity index.

We first uploaded the source image and target image into our S3 bucket which are shown in figure 4 and figure 5.

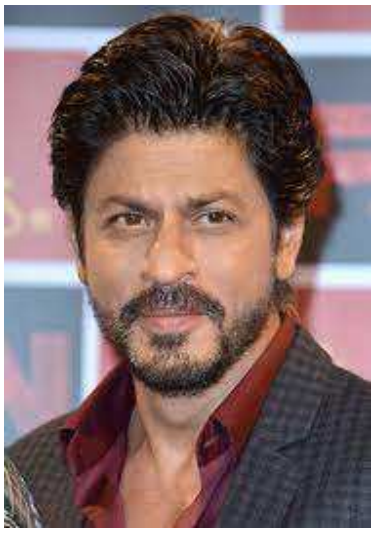

Figure 4. Source Image

Figure 5. Target Image

As shown in figure 4 and figure 5, the former is our source image and the latter is our target image. We uploaded both the images into S3 bucket as shown in figure 6.

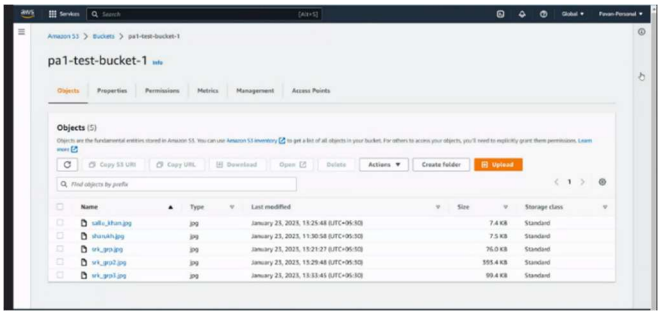

Figure 6. S3 Bucket

Once the images have been stored in our S3 bucket, we called AWS Rekognition which we connected with our bucket and compared the two images to verify if the person in figure 4 was present in the group photo that is figure 5. After the service completed the process, we were given the logs in CloudWatch, which showed the similarity index as approximately 99% which can be used to determine the presence of the person in the target image.

E. Analytics and Reporting

Analytics and reporting are integral components of our project. To ensure that we deliver on this front, we have developed custom algorithms that generate an integrity score based on the attendance data collected for each student. This score considers factors such as the frequency and distribution of a student's irregular attendance across all classes, as well as the balance of their attendance across multiple subjects. By using this score, we can assign each student a consistency score and a subject balance score which helps them in maintaining their attendance.

The consistency score showcases the student's regularity in terms of attendance. This helps us in tracking the student's attendance and his attitude towards the classes. The subject balance score, on the other hand gives us a report which shows his distribution of attendance across all subjects. There are instances where students favor certain subjects and don't show up for other subject classes. This needs to be addressed and the subject balance score helps us in doing that by providing a score, based on their attendance deviation from the ideal score. Other than these two scores, we will also provide the average attendance of the student.

All these scores will help us in generating a report unique to every student, showcasing their attitude and integrity towards learning and help them in improving or maintaining it.

IV. COMPARISION WITH EXISTING SYSTEMS

One of the primary concerns with fingerprint-based attendance systems is the potential for the spread of germs and diseases through contact with shared surfaces. This is especially true in high-traffic environments, such as schools or offices. As our solution doesn't have any common touch surface, it is safe to use and avoids spread of contact-based diseases like COVID-19. The cost of machines used to scan fingerprints of users is significantly high and also requires regular maintenance. The proposed solution doesn't require installation of any hardware device or sensors. Moreover, the operational expenditure is low as the lambda function integrated with the solution ensures that old target images are deleted at regular intervals thus saving space and reducing costs. Fingerprint based attendance monitoring systems can easily be faked by making a simple clone of someone's fingerprint with a glue or candle wax. Our proposed solution eliminates the chance of proxies and thus tries to improve a student's attitude towards learning [10].

Other existing systems that mark attendance using facial recognition overcome some of the above-mentioned problems. However, they still consume lot of time as each user has to scan his/her face sequentially. The proposed solution can mark attendance of all users in one click of a photo which saves significant amount of time. The integrity reports generated by our solution provides detailed statistics that help us in determining the attitude of students towards learning which is an additional useful functionality that no other existing system has implemented. Moreover, an easy-to-use android interface and notification of absence via emails add to the convenience of user.

V. RESULTS AND DISCUSSIONS

This research proposes a straightforward but effective method for calculating class attendance using facial recognition software. It is possible to take and record attendance with this automation system. We can quickly compare faces by using AWS Rekognition to leverage the data from our S3 bucket, thanks to the cloud computing provider AWS. This makes it possible to employ a quicker and more convenient approach to take attendance. We offer an accurate and speedy alternative to other existing systems by using the source image, which the students supplied using our Android application, and comparing it with the target image, which is the group photo of the students uploaded by the teacher. Since using proxies or counterfeit signatures is a prevalent practice among students today, traditional attendance marking methods like using a pen and paper, giving attendance on behalf of others or signing attendance forms are simple to go around, and students frequently take unfair advantage of this. It all boils down to one simple truth: Unless you are physically present in the lecture, there is no way to trick a facial recognition system since each person has a set of distinctive and idiosyncratic features that are universal to that person and cannot be copied or changed. This system will use the source and target images and compare the faces using deep learning techniques. The similarity index as shown in the figure 7 is based on several factors like visual geometry, including the connection between the nose, eyes, mouth, brow, and other features. To ensure the accuracy of the face comparison process by AWS Rekognition, we have set the similarity index cutoff at 98%. Any result below this threshold will be considered a mismatch, and the individual will be marked as absent. We took extra care to ensure the accuracy of the attendance, by integrating manual attendance manipulation and we have verified that all source and target images have been properly segregated into their respective folders. Additionally, we have established a regular schedule for deleting files from the S3 bucket to ensure that it stays organized and efficient.

If a student is absent, the AWS Rekognition system will not detect their presence. So, they will be marked absent, but sometimes this could lead to incorrect data entry due to blurred or unclear image. We utilize Amazon SNS to send an email notification to the student when they are marked as absent. This is done to inform them of their absence so that they may follow up with the teacher if necessary.

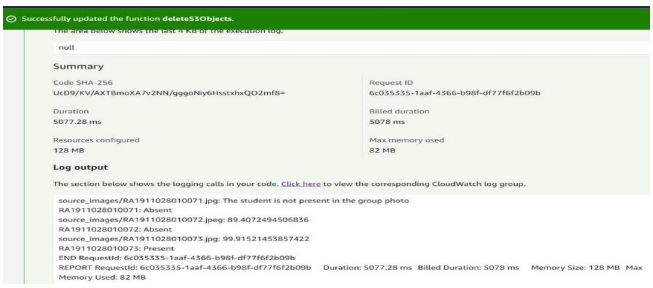

Figure 7. Attendance Configuration

There are many different institutions and universities, which will appreciate an attendance system like this. The harder to cheat system, easy to use and cost-effective system like this will be very beneficial to any organization that adopts this as their attendance monitoring system. The main objective is to foster a serious attitude among students towards their academic pursuits so that teachers may devote their time to completing the curriculum, without being hindered by a laborious attendance system. Insufficient attendance during lectures is a commonly encountered problem that results in a loss of valuable time amounting to a quarter of the lecture, thereby making it arduous for lecturers to complete the syllabus before the examinations. Our goal is to develop a model that is convenient for both staff members and students, characterized by efficiency, speed, and ease of accessibility.

V. CONCLUSION AND FUTURE SCOPE

Management of attendance is important for all companies, including educational institutions. By monitoring employees, such as students, to get the most out of them, it may manage and regulate the success of any firm. The proposed system offers a method for monitoring student attendance that aims to aid educators in managing and accurately documenting students' presence in classrooms or laboratories, without relying on paper lists, thereby saving time and effort. Additionally, the system includes an analytical capability that can generate statistical insights into student absences, generate reports on absence percentages, and issue student warnings for a predetermined timeframe. The developed system was built with the MVVM architecture to make insertions, deletions, and changes to data simple and straightforward to accomplish without interacting with the tables. It also features a beautiful and simple user interface.
AWS's cloud service has been leveraged to make the application more user-friendly and appealing in order to increase usage. The storage used by this system is an S3 bucket from AWS. Through AWS Lambda, all other services, like AWS Amplify, AWS Rekognition, and AWS RDS, carry out activities using this data. The system is functioning exhilaratingly and is available for use to control students' attendance for any department of the University or Institute, according to the application's test case. The suggested system makes use of a camera and face recognition to track a student's attendance. The faces are entered in a database once they have been matched to confirm their presence or absence. After comparing the stored photos in the database, it automatically stamps the student's attendance in the classroom. The faculty can utilize the date to keep track of a student's attendance status and get email notifications.

For future work, we will try to implement custom labels into our system. Custom labels are a feature provided by Amazon Rekognition which allows us detect particular products from the surroundings of our image. This can be used to determine the location and authenticity of the image taken and uploaded into our system to reduce the chances of it being cheated.

REFERENCES

[1] *R. K. Kodali and R. V. Hemadri, "Attendance Management System,"* 2021 International Conference on Computer Communication and Informatics (ICCCI), *2021, pp. 1-5, doi: 10.1109/ICCCI50826.2021.9402659.*

[2] *Shailendra; Manjot Singh; Md. Alam Khan; Vikram Singh; Avinash Patil; Sushma Wadar(2015) https://ieeexplore.ieee.org/document/7124938 doi: 10.1109/ECS.2015.7124938*

[3] *P. P. Godbole, A. Tomar, S. Bevinakoppa, P. Kundula, A. Aryal and B. B. Bhusal, "IoT Based Secured Online Attendance Management System," 2020 IEEE 7th International Conference on Engineering Technologies and Applied Sciences (ICETAS), 2020, pp. 1-6, doi: 10.1109/ICETAS51660.2020.9484219.*

[4] *A. G. Menezes, J. M. D. d. C. Sá, E. Llapa and C. A. Estombelo-Montesco, "Automatic Attendance Management System based on Deep One-Shot Learning," 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), 2020, pp. 137-142, doi: 10.1109/IWSSIP48289.2020.9145230.*

[5] *N. Gupta, P. Sharma, V. Deep and V. K. Shukla, "Automated Attendance System Using OpenCV," 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2020, pp. 1226-1230, doi: 10.1109/ICRITO48877.2020.9197936.*

[6] *D. Mijić, O. Bjelica, J. Durutović and M. Ljubojević, "An Improved Version of Student Attendance Management System Based on RFID," 2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH), 2019, pp. 1-5, doi: 10.1109/INFOTEH.2019.8717750.*

[7] *S. A. M. Noor, N. Zaini, M. F. A. Latip and N. Hamzah, "Android-based attendance management system," 2015 IEEE Conference on Systems, Process and Control (ICSPC), Bandar Sunway, Malaysia, 2015, pp. 118-122, doi: 10.1109/SPC.2015.7473570.*

[8] *Ferrara, L., Bergin, J.(2019, April 1) How Student Attendance Can Improve Institutional Outcomes Educause Review https://er.educause.edu/blogs/sponsored/2019/4/how-student-attendance-can-improve-institutional-outcomes*

[9] *The Guardian. (2018, April 13) Students Faking Attendance https://www.theguardian.com/higher-education-network/2018/apr/13/why-are-students-faking-attendance-they-feel-cheated-by-the-system*

[10] *Jacksi, Karwan & Mohammed, Falah & Zebari, Shahab. (2018). Student Attendance Management System. International Journal of Engineering and Technology. 6. 49-53. 10.21276/sjet.2018.6.2.1.*

[11] *OL. Google Android Developers(n.d.), Android Develop Guide. Retrieved October 28, 2022 from http://developer.android.com/guide/topics/fundamentals.html*

[12] *M. A. Hidayat and H. M. Simalango, "Students Attendance System and Notification of College Subject Schedule Based on Classroom Using IBeacon," 2018 3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE), Yogyakarta, Indonesia, 2018, pp. 253-258, doi: 10.1109/ICITISEE.2018.8720948.*

[13] *GreatSchools Staff (2023, March 1) Why attendance matters. https://www.greatschools.org/gk/articles/school-attendance-issues/#:~:text=The%20attendance%20rate%20is%20important,of%20students%20are%20frequently%20absent.*

[14] *Yaware (2015, January 24) 5 Old School Ways of Tracking Employee Attendance. https://yaware.com/time-tracking/5-old-school-ways-of-tracking-employee-attendance/*

[15] *Ak. S (2015, March 18) 6 Reasons Why We Chose AWS. Aptuz https://www.aptuz.com/blog/6-reasons-why-we-chose-aws/*