Logo

Description automatically generated**College of Computing and Informatics**

**Computer Science Department**

**University of Sharjah**

**fall 2021/2022**

**Face recognition-based attendance system for primary school and kindergarten kids**

#### by

Reema Alharbi (U17100706)

Khuloud Khalid (U18105068)

#### Supervised by

Dr Ibrahim Abaker Hashem

Dec 2021

**Table of Contents**

1. [Introduction 3](#_bookmark2)
2. [Motivation 3](#_bookmark2)
3. [Project Aim and Objectives 3](#_bookmark2)
4. [IDE’s, Tools and Technologies 4](#_bookmark2)
5. [Languages Used 4](#_bookmark2)
6. [Attendance System 5](#_bookmark2)
7. [User Interface 6](#_bookmark2)
8. [Facial Recognition Set Up 12](#_bookmark2)
9. [Database System 14](#_bookmark2)
10. [Restrictions 16](#_bookmark2)
11. [React Native code 18](#_bookmark2)
12. [Members Task Table 22](#_bookmark2)
13. [References 23](#_bookmark2)
14. Introduction

This project proposes that the system takes attendance automatically using face recognition cameras, an interface is provided for users to interact with the system to have a visual sense of the students’ attendance, the cameras are placed inside the buses at several locations to capture the faces of the students from various angles. The faces of students are previously stored in the database system. After the capturing of the student's faces, the camera detects and recognizes the faces of the students from the images and compares it to the images stored in the database, if the face matched with the stored image then the student is marked as present and no notification is sent to the parents via the application made for the system, if the image is not a match or undetected then the student is marked as absent and a notification is sent to the parents via the application and at the same time the image is stored in the database as a new image and marked as unknown before further calculations are done.

1. Motivation

There are incidents of children being left behind on school buses that happen every year, which can result in their deaths, especially in summer, due to high temperatures. Research has shown that children's bodies appear to heat up three to five times faster than adults, while children are unable to cool down, too. Pediatric associations worldwide have further documented the negative effects of high temperature on kids, with children frequently falling victim to dehydration of hyperthermia and other risks that may be fatal.

With the evolution of technology, we aim to create a system that decreases the number of children dying after being left behind on buses. Since it is easier to take attendance using face recognition than a manual attendance system and to make sure parents feel safe about having their kids riding school buses, our system makes sure there is a communication between them and their child’s status of whether they entered or left the bus by receiving notifications.

1. Project Aim and Objectives

With the evolution of technology, we aim to create a system that decreases the number of children dying after being left behind on buses. Our main objective is to create a successful system that helps in tracking the students attendance in a hands free-way and have a more accurate attendance system.

* Easy way to track students attendance.
* Attendance is easily reported to the school and to the parents.
* Have an integrated notification system.
* Providing a user-friendly UI for accessing the system

1. IDE’s, Tools and Technologies:

* AWS Rekognition API
* Android Studio: It consists of all Android SDK tools to design, develop, maintain, test, debug and publish our app.
* Android Software Development Kit (SDK)
* SDK Manager: It is one of the main tools to maintain the updates of all the installed components required to run the project. It also notifies us when the project is not compatible with device or any other compatibility issues and to download any component that is required.
* Firebase will be used to connect to the database and host the application.
* Visual studio code
* Java SE Development Kit (JDK)
* Expo Cli quickstart
* NPM Package manager
* Node.js 12
* Android Virtual Devices (AVDs)

1. Languages used:

* React Native was mainly used for creating the frontend of the project, the language is flexible enough to maintain code complexity, test, implementation, integration and support.
* Javascript
* Node.js

1. Attendance system

A facial recognition attendance system employs facial recognition technology to automatically recognize and verify a person based on their facial traits for the students. The data is recorded and stored in real-time by the system. The system take the attendance by face recognition of the students, coordinates are determined and then the system maps the endpoints and intersections of the face recognition.

We're utilizing a native react app to do face recognition for an attendance system. The attendance face recognition application will help students to:

1. Recognize when students enter the bus.
2. This application helps administration and parents in keeping track of their children & students.
3. knows if students are late or absent
4. The time when students arrive at school or enter buses will be recorded.
5. User Interface

User Interface Design for any application should be very simple. We should have only a few clicks or navigation among the features when using the application to avoid hassle. In this application, there are two main screens, the Login and Home screens. The login page is the first page which appears when the user uses the application. In that page, if he is a new user, he can sign up or if he is an existing user, he can login with the credentials.

If the user wants to use the RKM application, they must download the application from the play store, install and register it by providing login information. Once, they register the registered information is stored on the server and can be validated, checking the valid credentials for the next time they login with the application. The next screen is the homepage where the users can select features and use the app.

A person jumping in the air

Description automatically generated with medium confidenceA screenshot of a cell phone

Description automatically generated with medium confidenceDiagram

Description automatically generatedThese are the splash pages that will appear only the first time a user opens the application.

Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generatedThe user then will be directed to the Start screen and will be asked to choose the level of access, after that they will either login or signup.

Chart

Description automatically generatedGraphical user interface

Description automatically generated with medium confidenceSignup and Login pages.

After selecting student profile, new page appears i.e. Student Profile in which courses are shown along with the attendance percentages.

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated





1. Facial Recognition Set Up

Before we start writing our app, we would like to mention the API we will be using for face detection. AWS Rekognition API provides face detection and face recognition functionality via a cloud-based API. This allows us to send an HTTP request containing either an image or a URL of an existing image on the web and receive data about any faces detected in the image. It analyzes millions of images and videos within minutes and augment human visual review tasks with artificial intelligence (AI). We will be creating an attendance app with facial recognition features. It will have both server and client-side (React Native) components.

In this progress of the project, the facial detection in the app is responsible for the following:

* Scanning and connecting to AWS Rekognition peripheral.
* Asking for the user’s name.
* Asking the user to take a selfie to check if their face is registered.

There will be three pages to test the facial recognition system:

* First page will ask if the user wants to register a new face ID or to verify one.
* A page to register a new face ID
* Another page is for asking the user to take a selfie to verify if their face is registered.

Graphical user interface, application

Description automatically generatedGraphical user interface

Description automatically generated

When you connect to a peripheral, it will ask for your full name, you can access your camera and gallery by clicking on the *Capture Image* button. the image is sent to the API to check if the face is similar to one that is previously registered.

1. Database

A database is a logically organized collection of structured data stored electronically in a computer system. A database is usually administered by a database management system.

Data may then be accessed, managed, updated, regulated, and organized.

A database is a data structure that organizes and stores data. A database for a face recognition system, for example, might have tables for attendance data, leave time, and everything else about each student. Each of these tables would have its own set of fields that are relevant to the data it stores.

In a database setting, a database management system is used to store, manipulate, and manage data. Most data base management system packages allow users to run SQL queries to execute tasks such as database creation, data storage, and data updating.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

1. Restrictions

The functionality to preview the captured image is implemented using a library called **react-native-image-picker**that enables to capture a picture from the device's camera or to upload an image from the gallery, the library was successfully installed but we have faced some issues getting the files to interact with our current project files.

Text

Description automatically generated

Text

Description automatically generated

Adding these permissions to the **AndroidManifest.xml**file is what will enable to capture a picture from the device's camera or to upload an image from the gallery. Due to time restrictions, we weren’t able to implement that on time.

1. React Native code

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task | Status | Progress% | notes | name |
| Interface of the Facial recognition | Completed | 100% | - | Reema Alharbi |
| Facial Recognition Set Up | In Progress | 40% | Faced some restrictions | Reema Alharbi |
| Connecting App to Firebase | In Progress | 60% | - | Reema Alharbi |
| Credits Page | In Progress | 50% | - | Reema Alharbi |
| Database system in firebase | Completed | 100% | - | Khuloud khalid |
| Attendance system without coding | Completed | 100% | - | Khuloud khalid |
| Attendance system with coding using react native app | In Progress | 50% | - | Khuloud khalid |
| Connecting Database system to the app | In Progress | 60% | - | Khuloud khalid |

1. Members Task Table
2. References

[1] Engineering@ZenOfAI. (2020, January 20). Face Recognition App In React Native using AWS Rekognition. Retrieved November 11, 2021, from ZenOf.AI website: https://medium.com/zenofai/face-recognition-app-in-react-native-using-aws-rekognition-c10b188a6413

[2] How to Create a Face Detection App With React Native. (n.d.). Retrieved from Code Envato Tuts+ website: https://code.tutsplus.com/tutorials/how-to-create-a-face-detection-app-with-react-native--cms-26491