**Grade and Classroom Attendance Management system**

**(GCAM)**

SUBMITTED TO

UNIVERSITY OF MUMBAI

BY

Mr. TANMAY GURAV

T.Y.B.Sc COMPUTER SCIENCE

ACADEMIC YEAR 2020 -2021

D.G. RUPAREL COLLEGE

MATUNGA (W) MUMBAI - 400012

**A PROJECT REPORT ON**

ACKNOWLEDGEMENT

I am completely satisfied at this moment to present my project on Android App for Classroom Attendance and Grade System. This project has been a learning challenge process for me. The completion of this project would have been unattainable without the supervision given to me by all the Professors of my college and my friends.

INDEX

|  |  |  |
| --- | --- | --- |
| SR. NO. | INDEX TOPIC | PAGE |
| Preliminary Investigation | | |
| 1 | Current System | 5 |
| 2 | Limitation of Current System | 6 |
| 3 | Proposed System and its Advantages | 7 |
| 4 | Stake Holders | 9 |
| System Analysis | | |
| 1 | Event Table | 11 |
| 2 | Use Case Diagram | 12 |
| 3 | Activity Diagram | 13 |
| System Design | | |
| 1 | Component Diagram | 17 |
| 2 | Deployment Diagram | 18 |
| System Coding | | |
| 1 | Menu Tree | 20 |
| 2 | Structure of Database | 23 |
| 3 | Program Description | 22 |
| 4 | Validations | 24 |
| 5 | Future Enhancements | 25 |
| 6 | Reference & Bibliography | 26 |

**Preliminary**

**Investigation**

**CURRENT SYSTEM**

Marking attendance in the class session and entering the marks of the students are the essential tasks of the subject handlers, since marking the attendance can regulate the students to attend the sessions and verify the number of students in the class. Record of marks is inevitable to analyse the performance of the students in their exams. The management and maintenance of student information is a key task for any institution. The task of marking attendance and making entry of the exam makes are traditionally carried out manually with a log book. Later, this task is carried out by the desktop applications. The desktop application is a standalone application installed in the particular desktop or laptop and the tasks can be performed only with that particular desktop system.

Basically, attendance is taken in:

* By hand on paper.
* Standalone desktop application.

**Limitation of Current system**

* The main drawback of this system is that the computer systems are not portable hence it cannot be kept anywhere to perform the task such as mark and attendance entry.
* The entered marks can be viewed only on the particular system if the desktop is not connected with network. Another method for grade and attendance entry is web-based application.
* In this method, the attendance and the marks details are uploaded in a server through internet and the users such as students, parents, and teachers can view the marks and attendance through browsers with internet using any one of the devices such as desktop, laptop, and hand held mobile devices. This system is active only when the internet is on since the data are not been updated with the local database.

**Proposed System**

* The mobile application for the Student Attendance and Grade Management system is developed and deployed in the cloud server.

The users are divided into two groups namely student and staff.

* The student user can be either students or parents.
* The staff user can be the staff members, faculty member, principal, dean, etc. of the educational institution.

**Front end :**

Android Studio

**Back end :**

Firebase cloud service by Google

( [www.firebase.google.com](http://www.firebase.google.com) )

**Advantages**

* This application is installed in the mobile devices of the users such as student, staff and parents, faculty members for accessing the student marks and attendance details.
* The student users can view the attendance and marks that are uploaded in the database on cloud server.
* The staff users can enter, edit, modify and update the marks and attendance of the student through any mobile device.
* The student login enables the student users such as students and parents to view the class attendance details and marks that obtained by the students in the internal and external assessments.

**Stakeholders**

Stakeholders are the people who have an interest in the successful implementation of the system. We categorize them into the following groups:

1. **Technical Staff :** It includes those users who must ensure that the system operates in computing environment of organization.
2. **Service Provider :** This includes Google who’s mobile Computing platform is used to develop this system.
3. **Students :** This includes all the students and their parents who actively use this app for updates.

**System AnalysisEvent Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Event** | **Trigger** | **Source** | **Activity** | **Response** |
| User Login | Login | User / Staff / Admin | Checking Email and Password | Login successful |
| User navigates | Arrival of user | User / Staff / Admin | Navigates through Main Menu | Navigation done |
| Registration of User | Details taken | User / Staff | Adding new User | New user Added |
| Selecting Course | Course selection from DB | Staff | Course Selection | Course Selection done |
| Adding Course | DB Course Update | Staff / Admin | Course Update | Course Added to System |
| Taking Attendance | New Attendance Sheet | Staff | Taking Attendance | Attendance Recorded |
| Viewing Attendance | Download Document | User / Staff | View Attendance | Records Shown |
| Entering Grades | New Grade Sheet | Staff | Entering Grade | Grade Recorded |
| View Grade | Open Grade Sheet | User / Staff | Access Grade Sheet | Grade Sheet Displayed |
| Viewing Profile | Profile Page | User / Staff / Admin | Document Search | Profile Displayed |
| Editing Profile | Profile Page | User / Staff / Admin | Document Update | Profile Updated |
| Revoke Access | Firebase | Admin | User Access | Access Revoked |
| Forgot Password | Failed Login | User / Staff / Admin | Forgot Password Protocol | Password Reset |

**User** :- Student or Parents **Staff**:- Teachers and Faculty

**Use Case Diagram**

**LOGIN**

ADMIN

TEACHER

STUDENT

**ACCESS HANDLING**

**UPDATE PROFILE**

**VIEW PROFILE**

**VIEW GRADE SHEET**

**MAKE GRADE SHEET**

**VIEW ATTENDANCE**

**TAKE ATTENDANCE**

**ADD COURSE**

**REGISTER**

**Activity Diagram**

**User (Student Or Parent) :-**

**Start**

**View Profile**

**YES**

**Stop**

**View Attendance Sheet**

**Update Profile**

**View Grade Sheet**

**NO**

**Login**

**Authenticated**

**NO**

**YES**

**Register**

**Registered**

**Activity Diagram**

**Staff (Teacher) :-**

**Stop**

**Update Profile**

**Edit Grade Sheet**

**Edit Attendance Sheet**

**Make Grade Sheet**

**Take Attendance**

**View Grade Sheet**

**View Attendance Sheet**

**View Profile**

**Authenticated**

**Login**

**Register**

**Registered**

**Start**

**Add Course**

**NO**

**NO**

**YES**

**YES**

**Activity Diagram**

**Admin : -**

**Stop**

**Revoke User**

**Create User**

**Add Course**

**Access handling**

**NO**

**YES**

**NO**

**Login**

**Authenticated**

**Start**

**Registered**

**Register**

**System Design**

**Component Diagram**

Database JSON

Realtime Database

Documents

Collection

Firebase

Application

Cloud Firestore

**Deployment Diagram**

**CLOUD FIRESTORE**

**Real time DATABASE**

**FIREBASE**

**APP**

**USER**

**System Coding**

**Menu Tree**

**Structure of Database**

**Profile**

**Students**

**Edit Profile**

**Profile**

**Student**

**Grade**

**Attendance**

**Course**

**Login**

**Register**

**Application**

Data

Current Date

Attendance

Data

Students

Data

Courses

Teachers

root

Firebase Database JSON file

**Program Description**

|  |  |
| --- | --- |
| Activity Number | 1 |
| Activity Name | Splash Screen |
| Purpose | Display Logo |
| Input | - |
| Output | - |
| Validations | - |
|  |  |
| Activity Number | 2 |
| Activity Name | Login Activity |
| Purpose | To Login for all Users |
| Input | - |
| Output | Authentication Activity |
| Validations | All Users |
|  |  |
| Activity Number | 3 |
| Activity Name | Register Activity |
| Purpose | To Register Users |
| Input | - |
| Output | - |
| Validations | All Users |
|  |  |
| Activity Number | 3 |
| Activity Name | Main Menu |
| Purpose | Navigation through app |
| Input | - |
| Output | - |
| Validations | All Users |
|  |  |
| Activity Number | 4 |
| Activity Name | Course |
| Purpose | Select / Add Courses |
| Input | - |
| Output | Real time Database -Courses |
| Validations | Staff / Admin |
|  |  |
| Activity Number | 5 |
| Activity Name | Take Attendance |
| Purpose | Record Attendance |
| Input | Course Name, Today's Date |
| Output | Real time Database - Attendance -(current date) |
| Validations | Staff |
|  |  |
| Activity Number | 6 |
| Activity Name | Grade |
| Purpose | Assign Grade to Student |
| Input | Course, Student Name |
| Output | Firebase Documents, Real time Database-Grade |
| Validations | Staff |
|  |  |
| Activity Number | 7 |
| Activity Name | Profile |
| Purpose | Display Profile |
| Input | Firebase Documents |
| Output | Firebase Documents |
| Validations | All Users |
|  |  |
| Activity Number | 8 |
| Activity Name | Edit Profile |
| Purpose | Edit Profile |
| Input | Firebase Documents |
| Output | Firebase Documents |
| Validations | All Users (Only Owners) |

**Validation**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Index** | **Test Case** | **Test Data** | **Test Input Values** | **State** | **Expected Result** |
| 1 | The Email ID should be valid | Should satisfy regex pattern | user@domain.com user@domain.co.in user.name@domain.com user\_name@domain.com username@yahoo.corporate.in | **Valid** | Input Accepted |
|  |  |  |
| .username@yahoo.com username@yahoo.com. username@yahoo..com username@yahoo.c username@yahoo.corporate | **Invalid** | "Enter valid Email ID" message displayed |
|  |  |  |  |  |  |
| **Index** | **Test Case** | **Test Data** | **Test Input Values** | **State** | **Expected Result** |
| 2 | Password should be valid | Should be at least 8 characters | Password1 : p@s$w0r& 12345678 | **Valid** | Input Accepted |
|  |  |  |
| 1234567  qwerty  abc | **Invalid** | "Enter at least 8 characters" message displayed |
|  |  |  |  |  |  |
| **Index** | **Test Case** | **Test Data** | **Test Input Values** | **State** | **Expected Result** |
| 3 | Select Correct User type on Login | On login, the user type selected should match to the user type provided while registering | User type provided while registration = Selected User type | **Valid** | Input Accepted |
|  |  |  |
| User type provided while registration ≠ Selected User type | **Invalid** | "You are not a"+  Selected-user  message displayed |

**Future Enhancements**

* Most of the project is never completed.
* They always require future enhancements. It is a continuous process.
* The project needs to be developing all the time.
* The project is built such that the future enhancements are possible, activities and modules can be easily added and integrated with the project.
* The entire project followed the similar structure and methods were user so that we can prevent redundancy.
* Reports also can be created and integrated with the project very easily.
* In the future features like **Scan ID**, all the **processing done on the server** etc.

**Reference & Bibliography**

**Books Referred**

* **Firebase Docs**
* **Android Docs**
* **Android User Guide**

**Websites Referred**

* **<https://console.firebase.google.com>**
* **<https://en.wikipedia.org>**
* **<https://firebase.google.com>**