

**CS442 - MOBILE APPLICATION DEVELOPMENT**

**PROJECT PHASE 4**

**TEAM -2**

**Project:Proctor (Attendance Management System)**

**Team Members:**

|  |  |  |
| --- | --- | --- |
| **Name** | **A-number** | **Email Address** |
| Arvind Shekar | A20328640 | [anln43@haw.iit.edu](mailto:anln43@haw.iit.edu) |
| Abhijeet Kedari | A20353562 | [akedari@hawk.iit.edu](mailto:akedari@hawk.iit.edu) |
| Ajith Aravind Nayak | A20357090 | [anayak@hawk.iit.edu](mailto:anayak@hawk.iit.edu) |
| Desmond Appiah Kubi | A20333167 | [dappiahk@hawk.iit.edu](mailto:dappiahk@hawk.iit.edu) |

* **PROJECT DESCRIPTION**

Proctor is an easy-to-use attendance management system aimed at making the life of professors and students easy and the process of attendance tracking a painless one.

1. Professor can take and check attendance. So that processor will come to know of how students are doing in terms of attendance in his/her class.
2. Students can mark his/her attendance and they can check their status for any subject like in terms of attendance for that particular course.
3. Professor will initiate the process of attendance and student has to be in specific domain/range to mark the attendance otherwise they cannot mark the attendance this way professor will limit the proxies.

* **LIST OF FEATURES**

1. Simple and user friendly Interface
2. Response upon successful attendance
3. Professor should be able to take the attendance
4. Professor would be able to view the real time of attendance marked by a student
5. Student should be able to give the attendance
6. Student will be notified when it’s time to be in class
7. Check on proximity and allow user to mark the attendance
8. Student should see attendance score weekly.

* **COMPLETED FEATURES**

1. Professor is able to take the attendance
2. Student is able to mark the attendance
3. We implemented check on proximity so that student will not be able to mark the attendance if he/she is not within proximity.
4. Professor is able to view the real time of attendance marked by the students.
5. Notification reminder for class before 5 minutes

* **DESIGN DIAGRAM**

****

* **COMPONENT MODEL DIAGRAM**

****

* **USE CASE DIAGRAM**



* **TEST CASES**

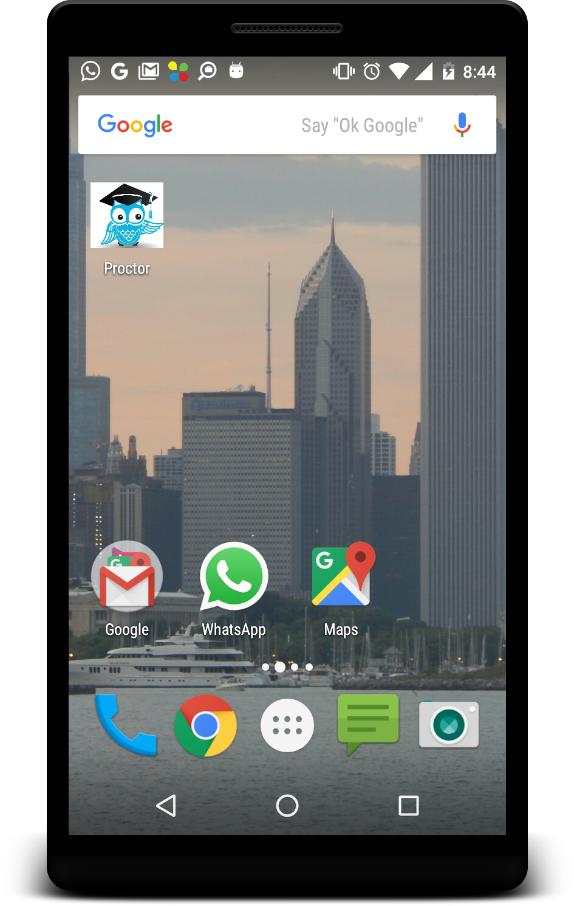
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case Id** | **Test Case Description** | **Execution Steps** | **Observation** | **Expected Result** | **Result** |
| 1 | Splash Screen Displayed on installing/opening the app | Install the Proctor application | Splash screen is displayed | A Splash screen with icon is displayed | Pass |
| 2 | Login Page | Upon installation a login page is displayed | The following fields are displayed userid and password | The userid and password should be displayed | Pass |
| 3 | Successful login | Enter valid user credentials at the login page | On entering valid credentials, the user is redirected to the home page | On entering valid credentials, the user should be redirected to the home page | Pass |
| 4 | wrong user credentials | Enter invalid user credentials | On entering wrong user credentials, a message is displayed indicating that the user has entered wrong credentials | On entering wrong user credentials, a message should be displayed indicating that the user has entered wrong credentials | Pass |
| 5 | Automatic Login subsequent to the first successfull login | 1. Install the Proctor app. 2. Enter valid credentials. 3. Close the App 4. Launch the app again | The user is redirected to the home page. | The user doesn’t have to enter the credentials after the first time | Pass |
| 6 | Successful logout | 1. Launch the app. 2. Click on the drawer on left side. 3. Click on the log out option | The user is successfully logged out of the app | The user should be logged out | Pass |
| 7 | Display of courses: Professor | 1. Enter valid credentials of user type Professor | 1.The courses taken by the Professor is displayed on the Home Page. 2. The courses schedule is also displayed | 1.The courses taken by the Professor should be displayed on the Home Page. 2. The courses schedule should also displayed | Pass |
| 8 | Display of courses: Student | 1. Enter valid credentials of user type Student | 1.The courses taken by the Student is displayed on the Home Page.  2. The courses schedule is also displayed | 1.The courses taken by the Student should be displayed on the Home Page.  2. The courses schedule should also be displayed | Pass |
| 9 | Navigation drawer menu | 1. Enter valid user credentials. 2. Swap in from left side of screen | A menu drawer would be displayed with options Home, Logout, Help, Alert | A menu drawer should be displayed with options logout, Home, help and alert | Pass |
| 10 | Receive Notifications as per class time | 1. Launch the app. 2. Enable the notification | The use receives notification 5 minutes before the start time of the class | The use should receive notification 5 minutes before the start time of the class | Pass |
| 11 | Turning of the Notification | 1. Launch the app.  2. Disable the notification | The user doesn’t receive any notification when the notification is disabled. | The user should not receive any notification when the notification is disabled. | Pass |
| 12 | Take attendance by Professor | 1. Enter credentials of user type Professor. 2. Click on the subject for which attendance is to be taken. 3. Click on the Take attendance button | A student would then be able to give his/her attendance | A student should be able to give his/her attendance | Pass |
| 13 | Check in by student | 1. Enter the valid credentials of user type student. 2. Click on the course. 3. Click on the check in button | The student is successfully checked in to the class | The student should be able to successfully check in to the class | Pass |
| 14 | Unsuccessful check in of student: out of proximity | Try to check in without being connected to the same network as Professor | The student will not be able to check in to the class | The student should not be able to check in to the class | Pass |
| 15 | Successful check in of student: Within Proximity | Try to check in after being connected to the same network as that of the professor | The student is able to successfully able to check in to the class | The student should be able to successfully able to check in to the class | Pass |
| 16 | Attendance sign in by multiple students concurrently | Check in attendance by 2 or more students concurrently | Any number of students can check in simultaneously | Any number of students should be able to check in simultaneously | Pass |
| 17 | Help page | 1. Launch the app. 2. Click on Menu Drawer 3. Click on help. | Help page is displayed with details about various features | Details about the app should be displayed | Pass |
| 18 | Home | 1. Launch the app. 2. Click on Menu Drawer 3.Click on the Home button | On clicking the Home screen for Professor and Student is redirected to the home page for respective user | On clicking the Home screen for Professor and Student is screen should be redirected to the home page for respective user | Pass |
| 19 | View Schedule | 1. Launch the App 2. For professor, select the course 3. click on view schedule button | On clicking the View Schedule button Professor is getting the runtime count of student with details who has marked the attendance | On clicking the View Schedule button Professor should be getting the runtime count of student with details who has marked the attendance | Pass |
| 20 | Start Discovery | 1. Launch the App 2. For Student, click on the course | On click of course, there is automatic backend check for network to which Student is connected and if student is not connected to any Wi-Fi network, will get toast message for failure message | On click of course, Network discovery should start and throw toast error message for failure case | Pass |
| 21 | Dynamic update in DB | 1. Change in backend DB 2. Reflect change in front end for user at runtime | On change in any Subject name or any other details, Web service taking care for dynamic update of those details | There should be a change in the page content if anything gets changed in backend | Pass |
| 22 | Multiple Attendance attempt | 1. Launch the App 2. Select the Course 3. mark the attendance 4. Again select the same course 5. Mark the attendance | On multiple attempt to mark the attendance for same subject on that particular day, doesn't make it double entry in present student list | Student should not be able to mark the attendance twice for same subject on same day | Pass |

* **SUPPORTED DEVICE TEST**

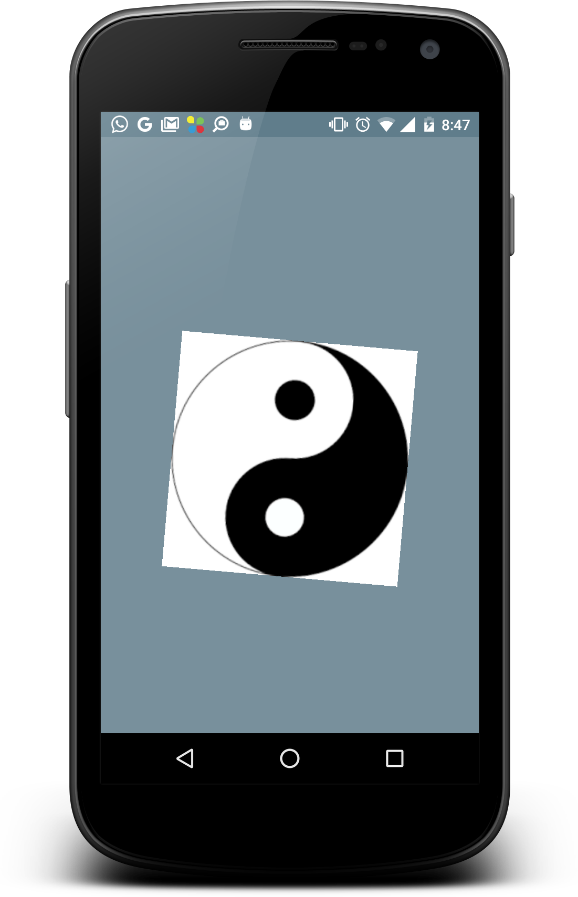
1. Samsung s7
2. One plus one
3. Moto g
4. Nexus 10

* **SCREEN SHOTS**

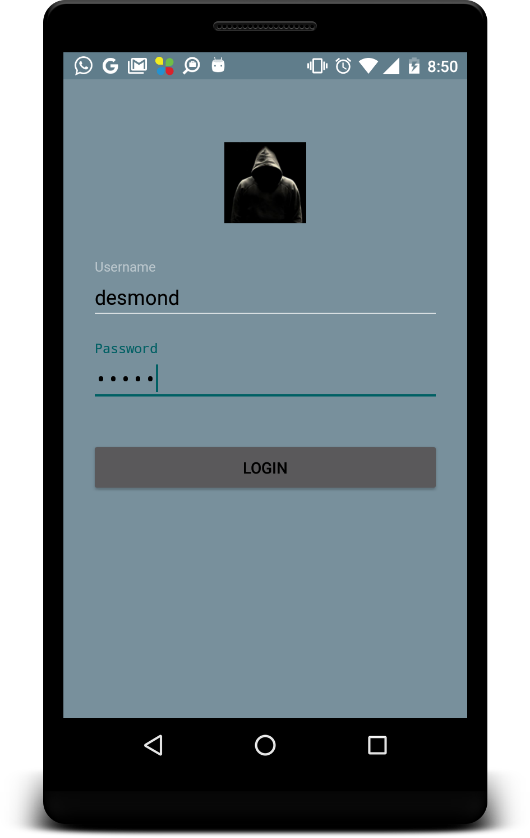
1. Logo



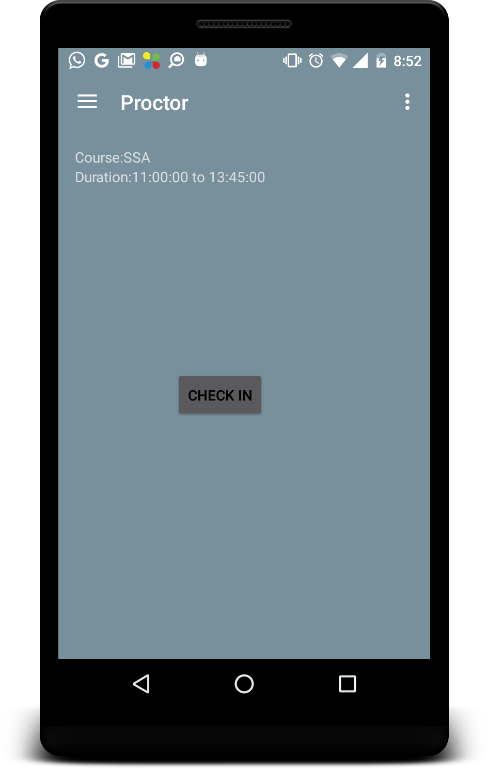
1. Splash



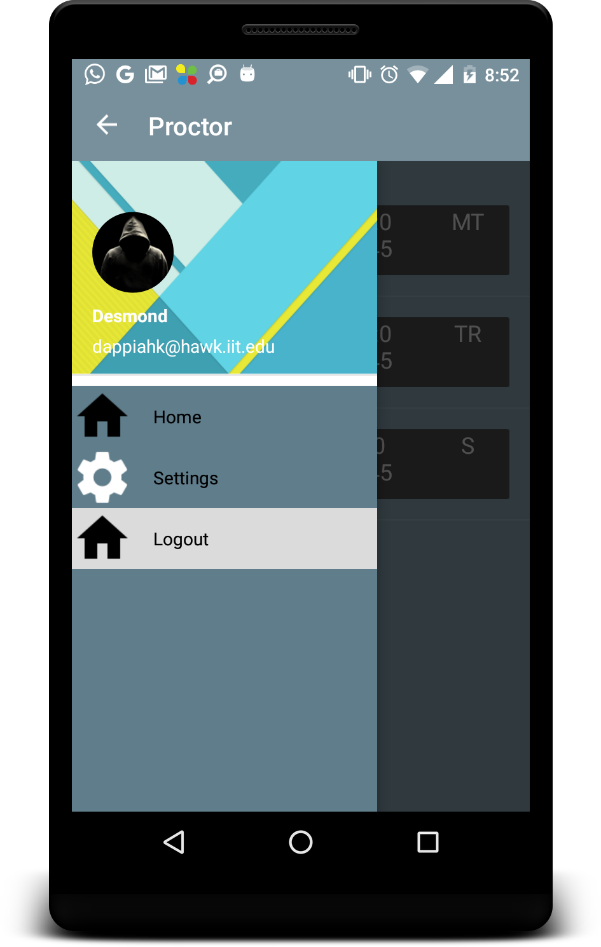
1. Login



1. Device search



1. Drawer Page



1. Student Main Page



1. Next one

* **List of main Classes and Fragments**

1. mainActivity
2. loginActivity
3. StudentFragment
4. ProfessorFragment
5. AttendanceFragment
6. ViewAttendanceFragment
7. HelpFragment
8. NavigationDrawerFragment
9. AlarmReceiver
10. SessionManager
11. JSONParser
12. Splash

**Revision History**

* Expanded classes list.
* Added one more feature to list of features