SECOND SEMESTER JAVA PROJECT

BENNETT
UNIVERSITY

Our Mentor:

Dr. Mayank Swarnkar
Assistant Professor

1.	Anshuman Agarwal	E18CSE022
2.	Anudit Nagar	E18CSE024
3.	Anurup Jalota	E18CSE025
4.	Anushka Upadhyaya	E18CSE026
5.	Bhavya Batra	E18CSE035

Bennett University (Batch EB-01)

1.Introduction

1.1. Purpose

The project aims to develop a working prototype of a system that will facilitate class control by automatically marking the attendance of students by using a self-mutilating QR code and a mobile app using a database of students. Attendr can be used to manage attendance and generate relevant statistics for the students and the teachers using an intuitive interface as well.

1.2.Document Conventions

While writing the SRS document for Attendr in order to make the document more readable and effective Arial font is used. The font size varies accordingly for the headings, subheadings and the write-up.

1.3.Intended Audience And Reading Suggestions:-

This Software Requirements document is intended for: −

1) Developers who can review project's capabilities and more easily understand where their efforts should be targeted to improve or add more features to it (design and code the application – it sets the guidelines for future development).
2) Project testers can use this document as a base for their testing strategy as some bugs are easier to find using a requirements document. This way testing becomes more methodically organized.
3) End users of this application who wish to read about what this project can do.

1.4. Product Scope

Attendr is designed to support multiple, simultaneous classes running across different schools of a university, allowing it to operate efficiently across any institution. With the ability to register multiple universities onboard the platform, Attendr can scale to adapt to the nature of how most educational institutions today are spread across multiple campuses or help to manage a fleet of private universities for any potential customer.

1.5. References

No references. The code was completely designed and implemented in-house.

2. Overall Description

2.1. Product Perspective

Attendr allows universities to simplify the monotonous procedure of manual attendance by automating the procedure by using self-mutating QR Codes and providing a dashboard to the universities for easily accessing and managing attendance of their students.

2.2.Product Functions

Class Portal

- Mark Attendance

Teacher Portal

- Start Attendance
- End Attendance
- View Courses and respective attendance

University Portal

- Add Teachers
- Remove Teachers
- View Statistics

2.3.User Classes and Characteristics

The following are the classifications of the users:

1) Students - Students are supposed to use the app for marking their daily attendance.

2) Teachers - Teachers are supposed to use the app for activating, deactivating and canceling or enabling the attendance.

3) Administrators - The Administrator is supposed to use the app in order to add and connect students and teachers.

2.4.Operating Environment

Class Portal is accessible via an Android App (developed in Java) which communicates over a REST API to the Attendr Backend. Attendr Backend uses Flask and Postgres in the backend to manage the student attendance, run on a Ubuntu Server Instance.

2.5.Design and Implementation Constraints

1) Attendr is currently limited to smartphones. Attendr's prototype can only work on Android devices, however, in the future, we will expand it to other operating systems.

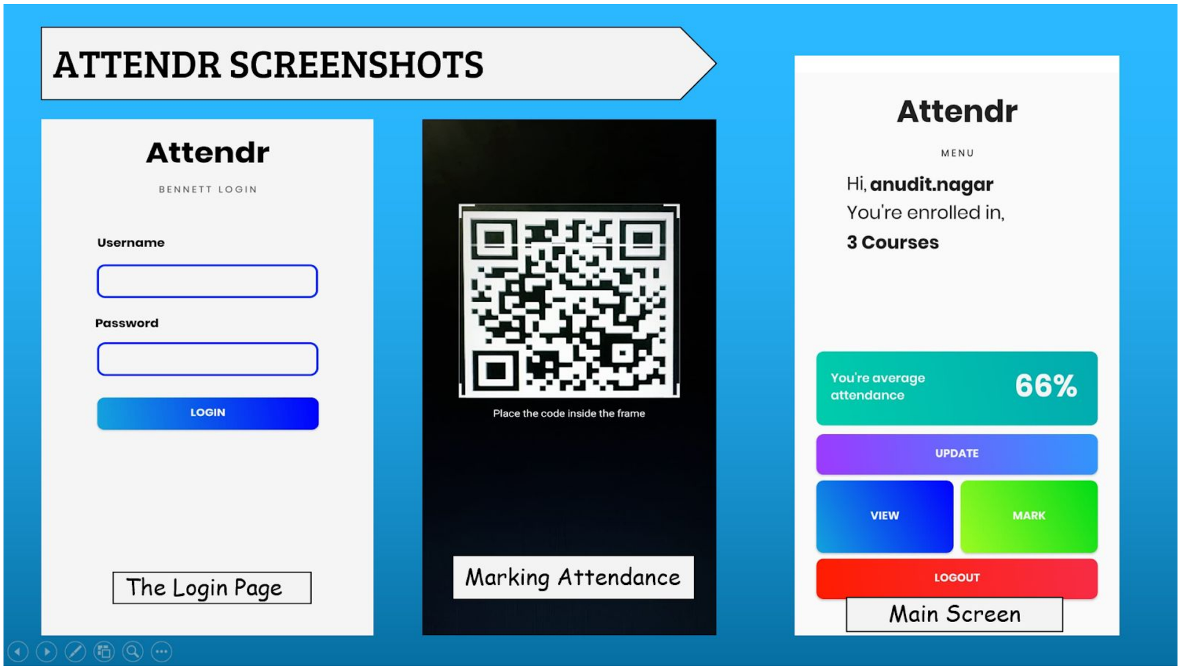
2) Attendr again is also limited by the ability to show the QR codes, which usually requires projectors or TV screens. Some universities may not have the infrastructure for screening

3) Attendr is only limited to Android and cannot be used on any other operating system.

3.External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

3.1.User Interfaces



ATTENDR SCREENSHOTS

The Login Page

Marking Attendance

Main Screen

3.2.Hardware Interfaces

1) The hardware interface includes all the supported that have a camera so as to scan the QR code.

2) A projector or a screen will be required for the display of the QR codes to be scanned.

3.3.Software Interfaces

An Android version of 7.1 plus

Visual Studio Code

Android Studio

PyCharm

3.4. Communication Interface

1) Attendr requires an internet connection
2) Communication standards to be used: HTTPS

4. System Features

4.1. Attendance

4.1.1. Description And Priority

Attendance is one of the highest priority in a university. Attendr takes responsibility of getting the attendance marked accurately.

4.1.2. Stimulus / Response Sequences

Stimulus: User clicks on Login

Response: User information and average attendance is displayed.

Stimulus: User clicks on 'Mark Attendance’

Response: QR code scanner opens.

Stimulus: User clicks on 'View Attendance’

Response: Attendance of all the respective courses is displayed.

Stimulus: User scans the QR code.

Response: Attendance is marked.

Stimulus: User clicks on 'Logout’

Response: User is logged out.

4.1.3. Functional Requirement

REQ 1: The user shall be able to Login.

REQ 2: The user shall be able to view all the contents.

REQ 3: The user shall be able to scan the QR code that is displayed on the screen.

5. Other Non-Functional Requirements

5.1. Performance Requirements

The system has been tested across multiple courses with multiple classes assigned to each course. The system was also stress-tested with multiple POST requests being sent to the server at the same time. The system performs well in all cases and carries out the required tasks, and produces expected results.

5.2. Safety Requirements

No potential threats or loss of property or harm can happen from this product, except to people’s attendance maybe.

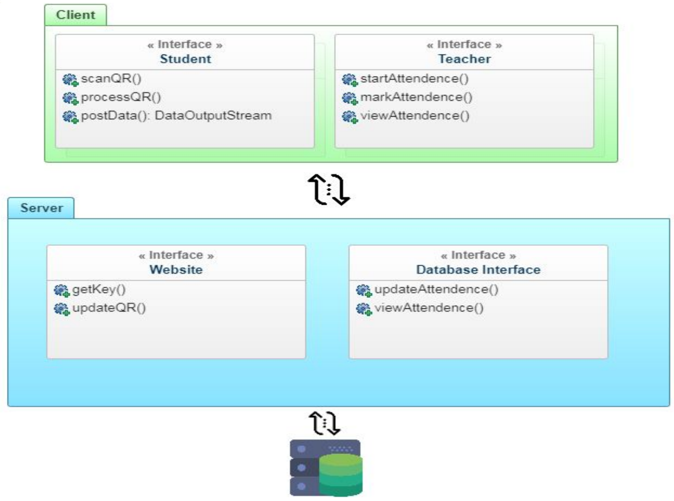
5.3. Security Requirements

Currently, the authentication happens using the LMS system of the University, and hence waives us off any liability of storing or using private or sensitive data of any sort.

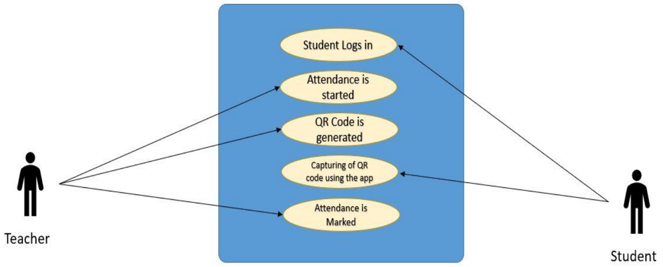
6. Other Requirements

6.1. Appendix A: Analysis Models

Class Diagram



UML Diagram


Student Logs in
Attendance is started
QR Code is generated
Capturing of QR code using the app
Attendance is Marked
Teacher
Student