Project Name: Automated Class Attendance System

Project Members: Kerschel James, Gerard Rique, Rondell Gulston

Course Code: COMP 3990

Course Name: Project

Department: Department of Computing and Information Technology

Lecturers: Phaedra Mohammed, Inzamam Rahaman

Date Submitted: Tuesday 11th April 2017

Table of Contents

[Abstract 6](#_Toc479662794)

[Introduction 7](#_Toc479662795)

[Problem Description 7](#_Toc479662796)

[Positioning 8](#_Toc479662797)

[Stakeholder Descriptions 8](#_Toc479662798)

[User Stakeholders 8](#_Toc479662799)

[Non-User Stakeholders 9](#_Toc479662800)

[Product Overview 10](#_Toc479662801)

[Product Perspective 10](#_Toc479662802)

[Needs and Features 10](#_Toc479662803)

[Alternatives and Competition 10](#_Toc479662804)

[Other Product Requirements 11](#_Toc479662805)

[Browser Compatibility 11](#_Toc479662806)

[Usability 11](#_Toc479662807)

[Responsiveness 11](#_Toc479662808)

[Visual Design 11](#_Toc479662809)

[Long Term Scalability 12](#_Toc479662810)

[Requirements Specification 12](#_Toc479662811)

[Functional Requirements 12](#_Toc479662812)

[Non-Functional Requirements 13](#_Toc479662813)

[User Stories 13](#_Toc479662814)

[Android Interaction 13](#_Toc479662815)

[Web Application 14](#_Toc479662816)

[System Diagram 16](#_Toc479662817)

[Use Case Diagram 17](#_Toc479662818)

[Entity Relationship Diagram 19](#_Toc479662819)

[Class Diagram 20](#_Toc479662820)

[Sequence Diagram 20](#_Toc479662821)

[Technical Constraints 22](#_Toc479662822)

[Design Specification 23](#_Toc479662823)

[Chosen System Architecture 23](#_Toc479662824)

[Discussion of Alternative Designs 23](#_Toc479662825)

[System Interface Description 23](#_Toc479662826)

[Detailed Description of Components/Subsystems 24](#_Toc479662827)

[User Interface Design 25](#_Toc479662828)

[Description of User Interface 25](#_Toc479662829)

[Screen Images 26](#_Toc479662830)

[Objects and Actions 27](#_Toc479662831)

[Implementation 29](#_Toc479662832)

[Languages 29](#_Toc479662833)

[Libraries/Technologies 29](#_Toc479662834)

[Navigation map of the System 30](#_Toc479662835)

[Screen shots of Android Application 31](#_Toc479662836)

[Screen shots of Web-Application 33](#_Toc479662837)

[Approach and Methodology 36](#_Toc479662838)

[Test Plan 41](#_Toc479662839)

[Test Approach 41](#_Toc479662840)

[Android Application testing 41](#_Toc479662841)

[Web Application testing 43](#_Toc479662842)

[Test Assumptions 45](#_Toc479662843)

[Features Tested 45](#_Toc479662844)

[Features Not Tested 46](#_Toc479662845)

[Business Aspects 47](#_Toc479662846)

[Individual Contributions 48](#_Toc479662847)

[Financial Considerations 49](#_Toc479662848)

[Conclusion 50](#_Toc479662849)

[Future Work 50](#_Toc479662850)

[References 51](#_Toc479662851)

[Appendix 52](#_Toc479662852)

**List of Figures**

[Figure 1: demonstrates how a student registers their attendance to a class using their student 17](file:///C:\Users\kersc\Desktop\Final%20Project\COMP%203990%20Project.docx#_Toc479638712)

[Figure 2: shows the interaction of the staff member with the web application 18](#_Toc479638713)

[Figure 3: The entity relationship diagram 19](#_Toc479638714)

[Figure 4: This diagram proceeds to illustrate the relationships 20](#_Toc479638715)

[Figure 5: Student interaction with the barcode scanner 21](file:///C:\Users\kersc\Desktop\Final%20Project\COMP%203990%20Project.docx#_Toc479638716)

[Figure 6:show the staff member interaction with the web application 22](#_Toc479638717)

[Figure 7:shows our student reports page 26](#_Toc479638718)

[Figure 8:show the general reports page 26](#_Toc479638719)

[Figure 9: shows the deliverable of the system 27](file:///C:\Users\kersc\Desktop\Final%20Project\COMP%203990%20Project.docx#_Toc479638720)

[Figure 10 Figure 11 31](#_Toc479638721)

[Figure 12 32](#_Toc479638722)

[Figure 13 33](#_Toc479638723)

[Figure 14 34](#_Toc479638724)

[Figure 15 34](#_Toc479638725)

[Figure 16 35](#_Toc479638726)

# Abstract

The attendance current policy at the University of the West Indies requires students to be present at 75% of their classes for a given semester. It is important that students can maintain their attendance in order to learn the extensive material needed for them to pursue their education. In order to keep track of students’ attendance at the University of the West Indies, we decided to create an automated system, in an effort to phase out the manual task of signing a role. This burden of a task is done during every class, can prove to be a very distracting and time consuming, as it requires students to turn their attention away from the lecturer to sign the roll. As such, the focus of this project was the development of a system, which uses barcodes on UWI ID cards to take roll for each class. The new Class Attendance System will comprise a web application for the tabulation of data and an android application for the retrieval of attendance records.

# Introduction

## Problem Description

Currently, at the University of the West Indies, it is required that lecturers take class attendance for administration and examination purposes. This process involves having students manually write their names and other relevant information on a register each class they attend. This process can become very repetitive and distracting to classes at times as it is done during the teaching session of the class. Occasionally, students may perform unethical practices such as forging signatures to register for their fellow classmate. In addition, staff members could misplace these registers before they can tabulate the necessary information about the class.

Due to this concurrent problem, the objectives of this system were to create an easier way for students to state their presence in class, provide an efficient way to store the data and provide a means for relevant authorities to access the tabulated information gathered. As such, allowing students to scan their UWI ID cards, whilst collecting their entry time into a class will provide staff with online access to this information via a web application. The resource required to complete our solution was a barcode scanner, which was created via an android application. Due to further research, the choice of using the barcode was a last resort due to the lack of authorization to use the proximity chip.

## Positioning

The Automated Class Attendance System offers a custom built android application for data capture of student’s attendance and a web application for the tabulation of the captured data. This system will benefit both the lecturers, who currently have to take attendance of their class and students who record their on the class roll. It will reduce the hassle of taking class attendance by removing the writing task to a simple scan of a card. Using the web application provides the user with a professional look on reports and gives them the choice of being able to export and print these reports for further usage. Additionally in includes the use of firebase authentication for the safety of students personal information.

## Stakeholder Descriptions

### User Stakeholders

|  |  |  |
| --- | --- | --- |
| Name | Description | Role |
| Staff Members | Stakeholder that interacts mainly with the web interface. | Able to generate tabulated data on courses for class attendance purpose |
| Examination staff | Stakeholder can view attendance of students | This stakeholder can view the students who are eligible to write examinations and ban the unqualified ones |
| Students | Students being relieved of the manual task of signing a register | Students use their student ID cards as a means of registering their presence in a class with the system |

### Non-User Stakeholders

|  |  |  |
| --- | --- | --- |
| Name | Description | Role |
| Software Architect | Stakeholder that is in main actor in the system development process | Takes the architecture of the system and transitions it into a software solution based on the relevant requirements. This will include the process of design, implementation and testing. |
| Project Manager | Stakeholder in charge of overall system management | Specifies relevant resources and administers their uses. These resources include other project members, software, hardware etc. |
|  |  |  |

## Product Overview

### Product Perspective

The class attendance system consists of two subsystems:

1. Web-based for the tabulation of the data and generation of reports on classes and students
2. Android application for the recording of data. The students’ ID cards will act as an identifier for their presence in a class. As a student enters the classroom, they scan their cards on the android barcode scanner to register their presence. The information on students’ attendance will then be recorded in the database and queried to get relevant reports.

### Needs and Features

* UWI ID cards with barcode
* Register time and presence of the students’ arrival in classes.
* Student information management system
* Attendance data presentation for evaluation
* Inform students if they are in the correct class
* Allowing staff members to search for specific student data

### Alternatives and Competition

The “Automated Class Attendance System” is not a new idea as it was already created using various technologies to identify students. A popular technology used was the RFID scanner as it is preferred over the traditional barcode scanner since it provides a more accurate means of scanning the cards. On the other hand, our system is tailored specifically for the University of the West Indies (UWI) with the custom classrooms and courses added.

## Other Product Requirements

### Browser Compatibility

The web application should work on browsers such as Firefox and Google Chrome and requires JavaScript enabled.

### Usability

The system should be functional to both staff members and students. The students have a simple interaction with the system, scanning their cards. The staff members should have access to data retrieved and stored in the database for further usage. A simple and user-friendly interface is required to ensure the application is easily used by staff.

### Responsiveness

The android barcode scanner must be able to take the ID card data quickly and store without any delays in order to overthrow the old manual system.

On the other hand, the web application deals with a large database therefore a small delay will be acceptable when retrieving the data for reports.

### Visual Design

The mobile application will be built in such a way that minimal interaction will be needed to allow the user to navigate to its main component, the barcode scanner, quickly.

### Long Term Scalability

The system can be improved for future use, by making the system accessible to not only classes but also clubs and meetings in order to keep track of students’ activities.

The system could also be used for lecturers to upload marks for individual students on the web app and allow the students to use their ID credentials to access their course-work grades.

# Requirements Specification

## Functional Requirements

|  |  |  |
| --- | --- | --- |
| # | Task | Description |
| 1 | Identification of a student | The system must be able to uniquely identify each student based on their UWI ID cards |
| 2 | Time stamping of identification | The system should record the time the user attends the class |
| 3 | Selection of Course | Allow the staff member to select the course to which the attendance information is about to be recorded |
| 4 | Multiple data sending to database from multiple barcode scanners | The system must cater for multiple users using it simultaneously |

## Non-Functional Requirements

|  |  |  |
| --- | --- | --- |
| # | Task | Description |
| 1 | Ease of Use | The system should be easy to use |
| 2 | Data Integrity | Attendance and personal details about students should be kept secure from unauthorized personnel |
| 3 | Reliability | The system should be able to recover smoothly in times when there are network connection drops |
| 4 | Responsiveness | The system must be able to respond quickly to events ( scanning a student’s ID) in order to give real-time feedback and to ensure that it is an improvement from the previous system |

## User Stories

### Android Interaction

Staff member is ready to start taking attendance

1. Open the application on the android device
2. Select the room that the course is going to be taught in
3. Select the Course
4. Press the Camera icon to begin taking attendance records

Student Enters class and signs register (Barcode scanner)

1. Student places ID card on the barcode scanner vision
2. Student checks on device screen to ensure that they are in the correct class

### Web Application

Staff member searches for a student’s class attendance

1. Visits the website
2. Logs in with their email and password
3. Navigates to the Student tab
4. Writes in the student’s ID or name
5. Writes in the Course Code
6. Select a time period to show attendance between
7. Presses Generate Graph

Lecturer Uploads a Course List

1. User Logs into system
2. User Navigates to Course tab
3. User selects course to which they are going to upload list for
4. User presses the Upload button

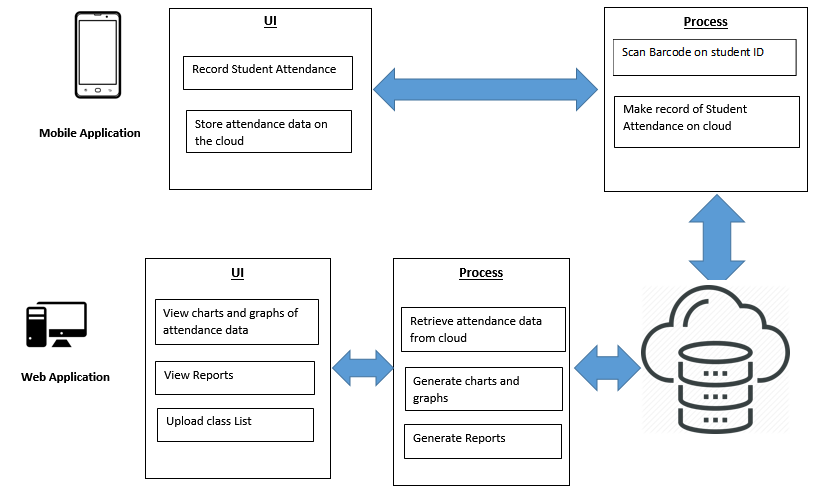
Staff user generates a report

1. User logs into the system
2. User Navigates to the Reports tab
3. User selects the type of report to which they would like to report
4. User has the option to download the generated report

Staff member creates an account

1. User clicks Sign up
2. User fills out the necessary information on the form
3. User presses the Submit button

## System Diagram



## Use Case Diagram



Figure 1: demonstrates how a student registers their attendance to a class using their student

Figure 1.1

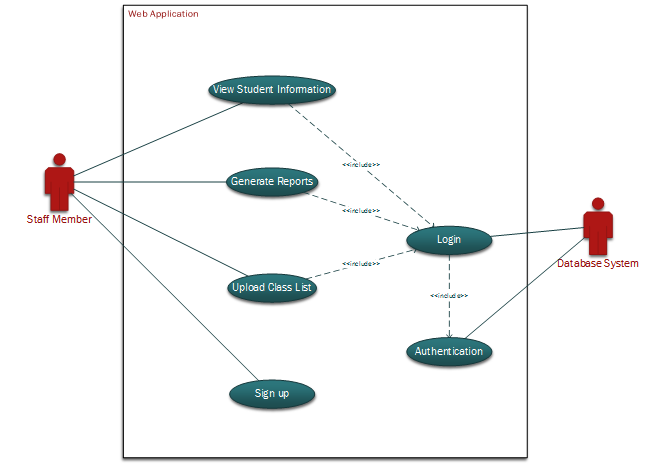


Figure 2: shows the interaction of the staff member with the web application

## Entity Relationship Diagram



Figure 3: The entity relationship diagram illustrates the relations that each system component has with each other

## Class Diagram

The class diagram displays three main entities of the system, which are students, staff and courses. 

Figure 4: This diagram proceeds to illustrate the relationships that these share, in which they are a part of one another. The other entities are aids in system’s functionality.

## Sequence Diagram

The sequence diagrams provided, shows the system comprising of the two main scenarios. The first being the collection of data and the other being the retrieval of information. The first diagram would show the interaction when a student uses his/her ID card in order to state his/her attendance on the system. The second diagram shows the different procedures that are available to the staff members after the login process.

Student interaction sequence



Figure 5: Student interaction with the barcode scanner

Staff Members



Figure 6:show the staff member interaction with the web application

## Technical Constraints

The project was done by students, our team, which restricted our access to the proximity scanner on the UWI ID cards, therefore we had to use the traditional barcode. We therefore had to create a barcode scanner using an integrated development environment (IDE) for the Android platform, Android Studio.

Additionally, when creating the Android application there was a problem with the Zxing library which disabled the use of Intent class on the application device tested, (Huawei P9), therefore an alternative to the Android Intent activity had to be made. The use of multiple Java class pages could not be used due to this problem. This reduced the level of modularity in the Android system.

# Design Specification

## Chosen System Architecture

The architectural system chosen was the Layered Architecture. This design pattern was chosen as it provides a logical structure and communication between our two sub-systems; the android application and the web application. In a layered system, the layers communicate with each other by sending relevant information through the different components of the system.

## Discussion of Alternative Designs

The initial implementation of the system would have involved the use of the UWI ID card’s proximity chip along with relevant scanners, but was not used due to the difficulties in gathering the needed resources and the lack of authorization by the UWI, St. Augustine campus. We therefore made use of the ID card’s barcode as an alternative.

## System Interface Description

The system interface design includes the web application and the android mobile application as its two main systems. The user inputs the data into the android system via their ID barcode, which is then sent to a database. From the database, the data can be retrieved and queried by the web application for relevant reports.

### Detailed Description of Components/Subsystems

#### Component (1-n)

#### Mobile UI Component

The mobile Android application featured three main pages namely, the Classroom Selection page, the Course Selection page and the Course Information page. The home page is the Classroom page since the user has to first select their class location to be able to select the course. The Course page displayed a list of courses that are being taught in a specific classroom. After the classroom is selected, the user is then given a list of courses that are taught in that classroom to be selected. The Course information page then allows the user to see details about the class and enable them to begin scanning student’s identification.

#### Mobile Process Component

The mobile application process component consist of the database connection and the barcode scanner camera functionality. We used the firebase database in order to save students attendance data retrieved from the barcode scanner.

#### Web UI Component

The UI Component for the Web application would provide the user with the three main functions that can be performed, in which the Home Page provides the information regarding them. The Courses Page provides the user (lecturer) to be able to upload, for each course that they teach, a file containing a list of the students that are currently enrolled in the course. The Students Page would allow the user (lecturer) to view the information for a student pertaining to a specific course. The Reports Page will present different reports based on the subcomponent selected, based either on all courses, a particular course or general information on a student.

#### Web Process Component

The Process Component for the Web Application would ensure the functionality of the selected tasks in the UI Component. Behind each of the three main features, there would be a number of sub processes that in their completions would ensure the efficiency and capabilities of the main processes. For example, in order to upload a list of the students that are enrolled in a class, a list of the available courses must be retrieved from the database and then can be selected as an option.

## User Interface Design

### Description of User Interface

#### Mobile Application

The mobile application featured and very simple layout, which enables the user to select a classroom and the course to which class attendance is about to be taken for. Simplicity was the key aspect for this interface to allow ease in usage. The flow of events are as follows : Select a classroom -> select a course -> see course details -> click the camera icon to begin taking class attendance. It should also be noted that this device requires an internet connection in order to access the information.

#### Web Application

The web application firstly provides the user (staff) with a classic log-in and sign-up page in which they can easily authenticate themselves being able to access the attendance information. Upon validation, the user will then be able to effortlessly transition, using the top navigation bar, between the different pages of the application, each providing a different function that can be selected, all explained on the home page.

## Screen Images

Initially the design of the web system was to look like these screen images, but as we progressed through development out design and functionality of the system changed.

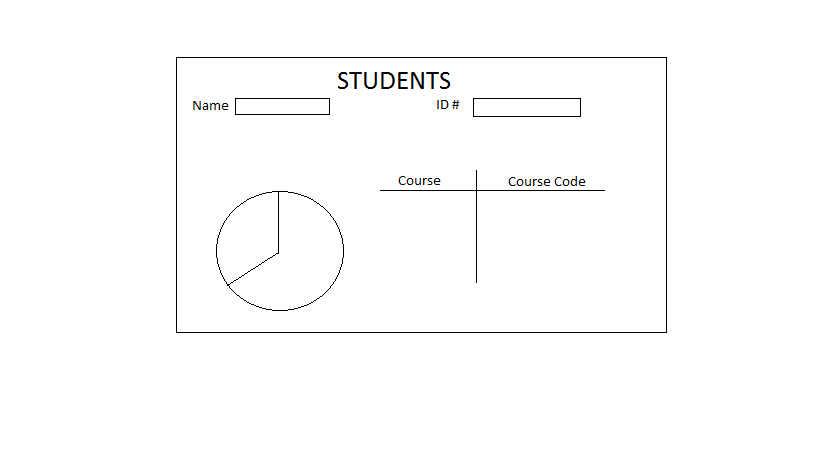


Figure 7:shows our student reports page

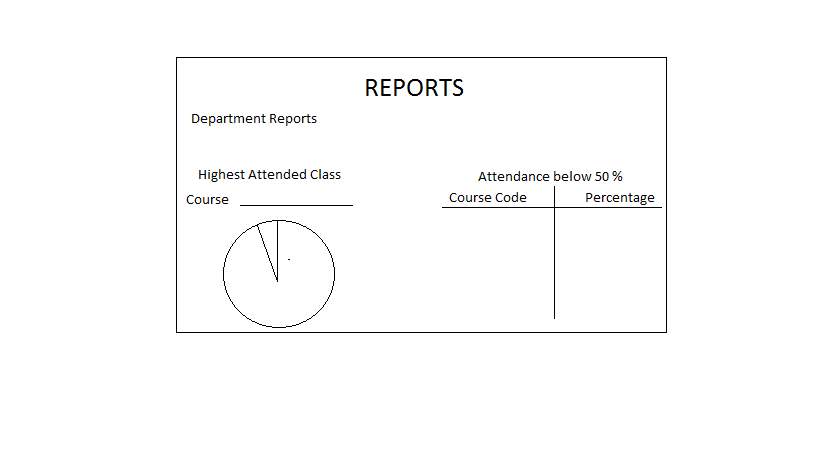


Figure 8:shows the general reports page

Scanning

Initially the system required a physical scanner but due to the lack of funding and resources, we decided to create an Android application that could perform the appropriate task.

### Objects and Actions

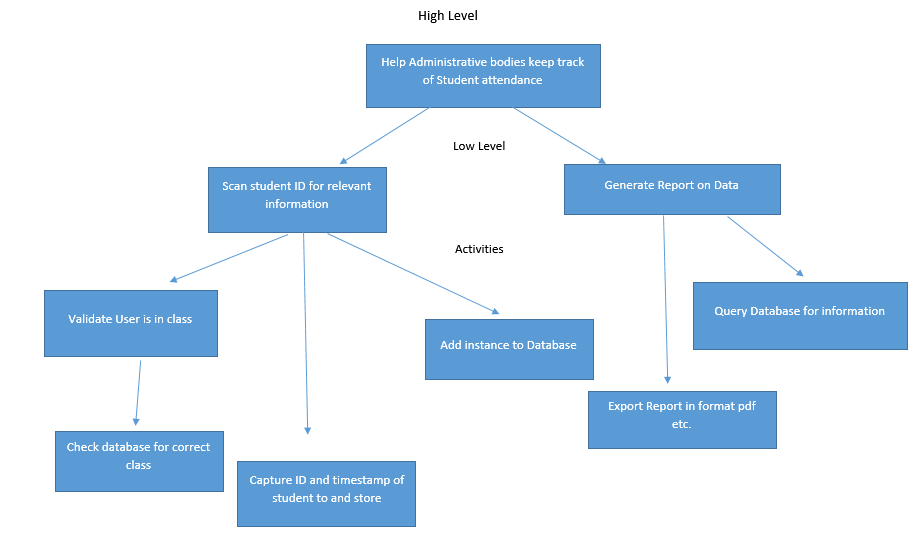
The diagram below shows the deliverable of the system which is broken down into hierarchy of high level, low level details and its activities.

Figure 9: shows the deliverable of the system

* Objects

The objects of the system would be the high level concepts that the user would be interacting with. The only high level concept of the system would be to tabulate and maintain student attendance data for use by staff members

* Actions

The actions of the system would be involve the lower level concepts along with the activities associated with them. The low level concepts of the system would be the scanning of the UWI ID card and the generation of reports. These two concepts can be broken down into the different activities stated above, for example, validating that a student exists within a specific class.

# Implementation

The Class Attendance System consists of both a mobile and web application. Both synchronized to an online cloud hosted database. The cloud hosted database used is used for this project is called firebase.

## Languages

The android application was built with Extensible Markup Language (XML) and Java. XML was used to design the layout of the application, and Java was used to implement the logic of the system.

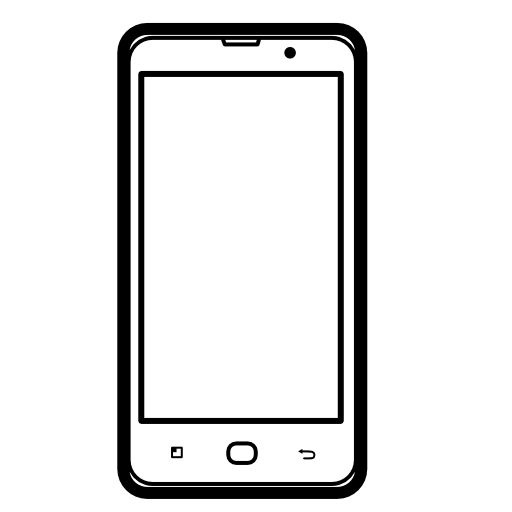
For our web application, HTML and CSS were used for the layout and design of the application, whereas JavaScript was used to implement the functionality of the web application.

## Libraries/Technologies

Since we wanted the web application to operate in real time, we implemented a third party framework called AngularJS, which allowed us to provide dynamic changes to the application’s displayed content. For the design of the application, we implemented a third party styling framework called Bootstrap, which allowed for a better user interface design. We also used a cloud database Firebase.

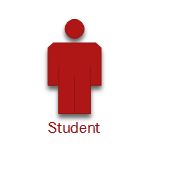
For the android application a third party software, Zxing, was used as the barcode scanner.

## Navigation map of the System



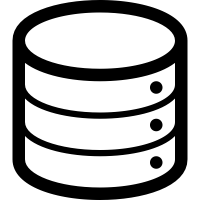
Set up for recording data

Scan Barcode



**Mobile Application**

Data stored in database



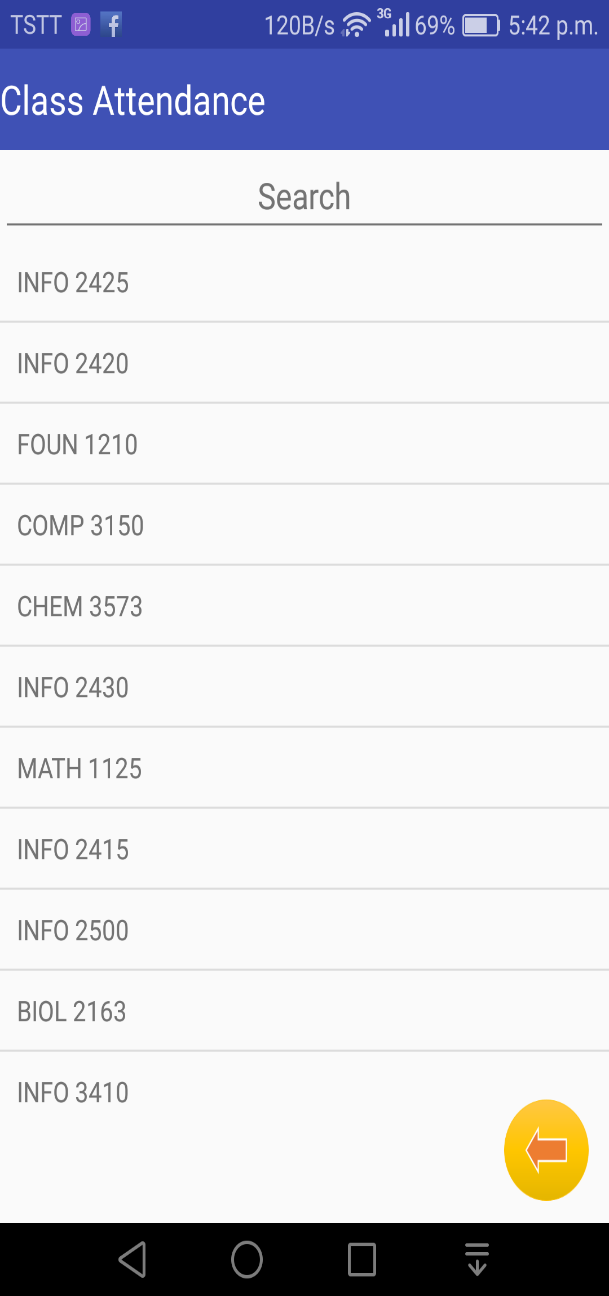
Data retrieved from database to for reports





### Screen shots of Android Application

User can select class room User selects Course from list



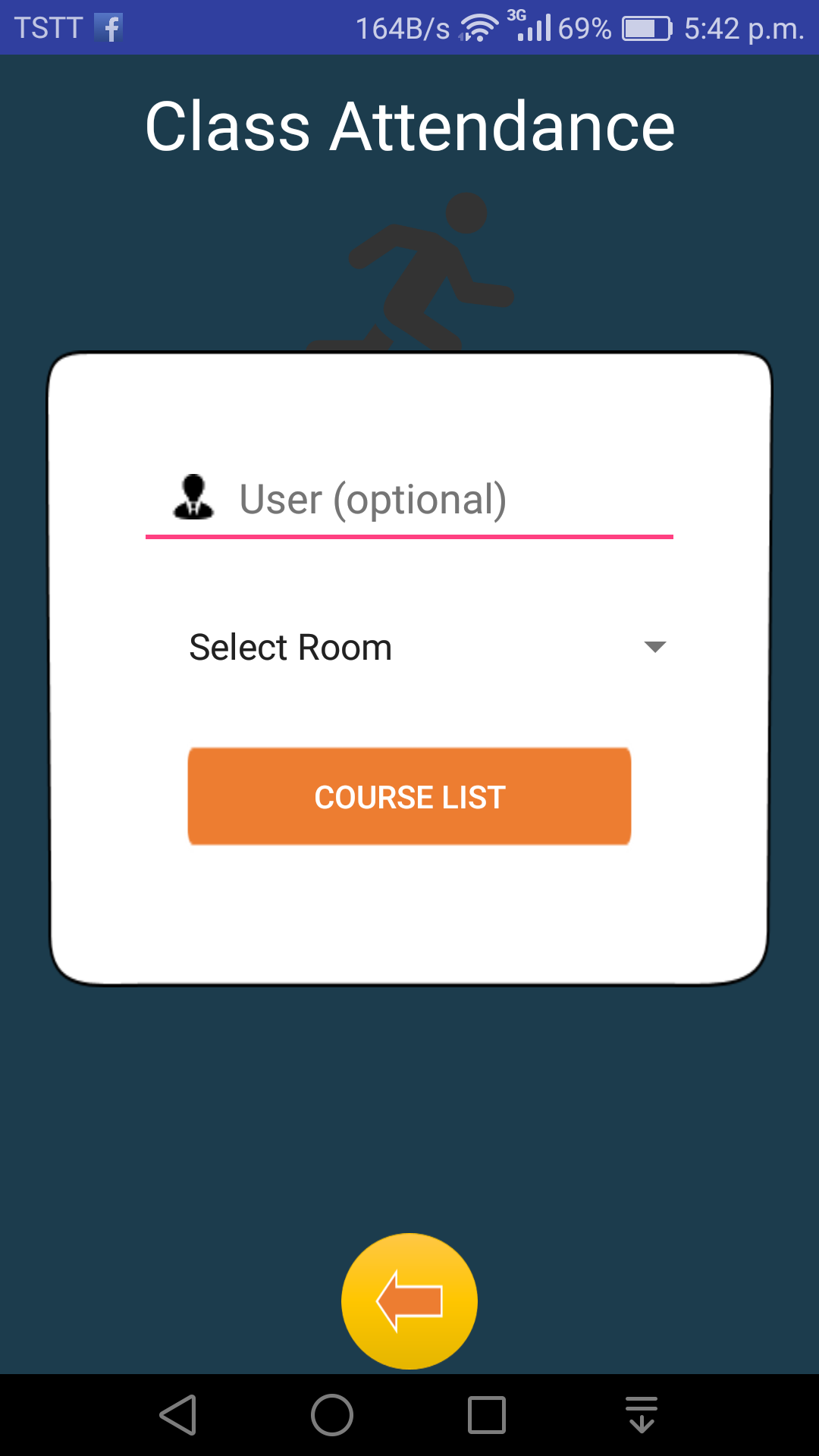


Figure 10 Figure 11

The User(optional) tab was to be used for the web application for the user to be able to view their class information. Unfortunately it was not implemented on the Web therefore it was disabled on the android application

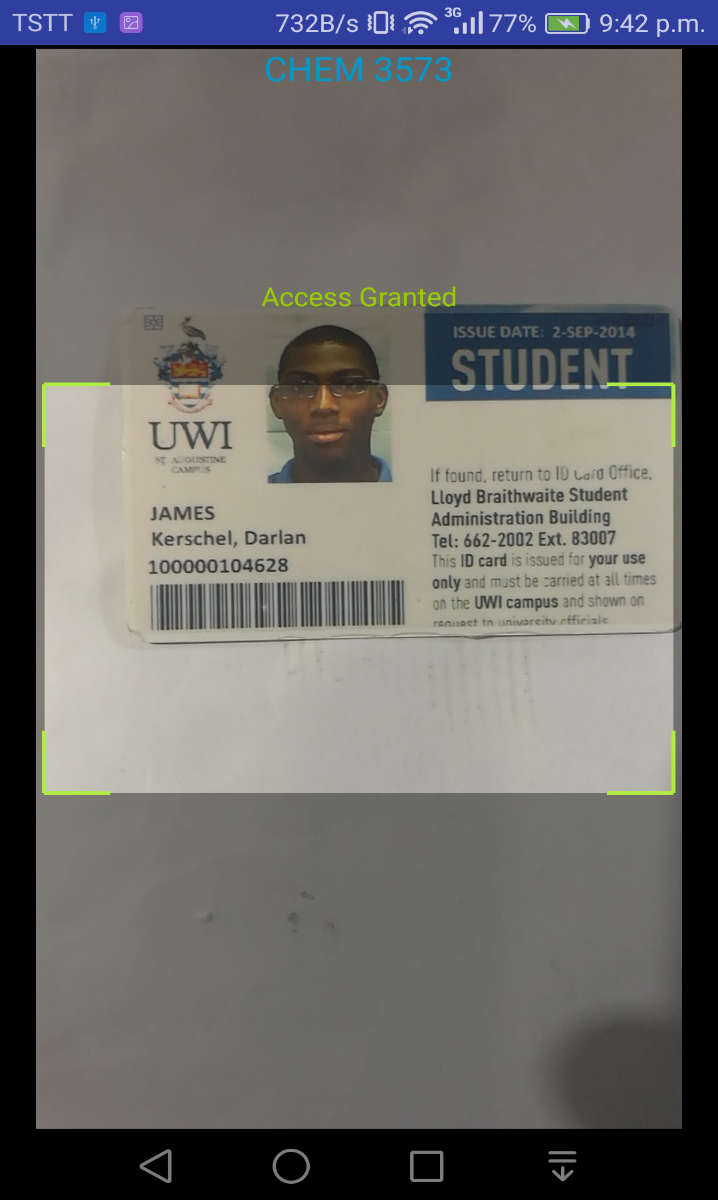
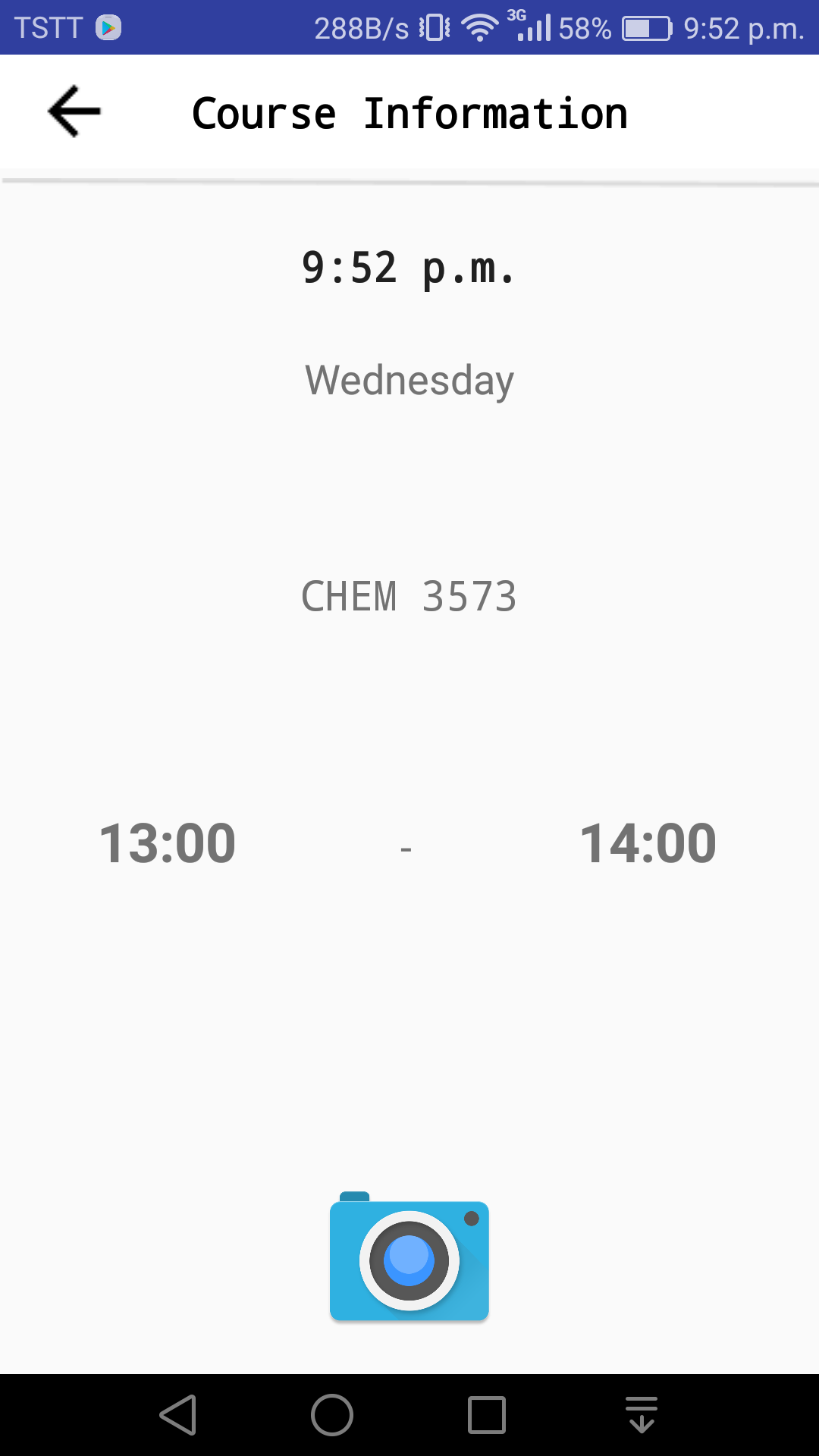
** Screen shot of Course Info Screen Screen shot of barcode scanning** 

Figure 12

### Screen shots of Web-Application

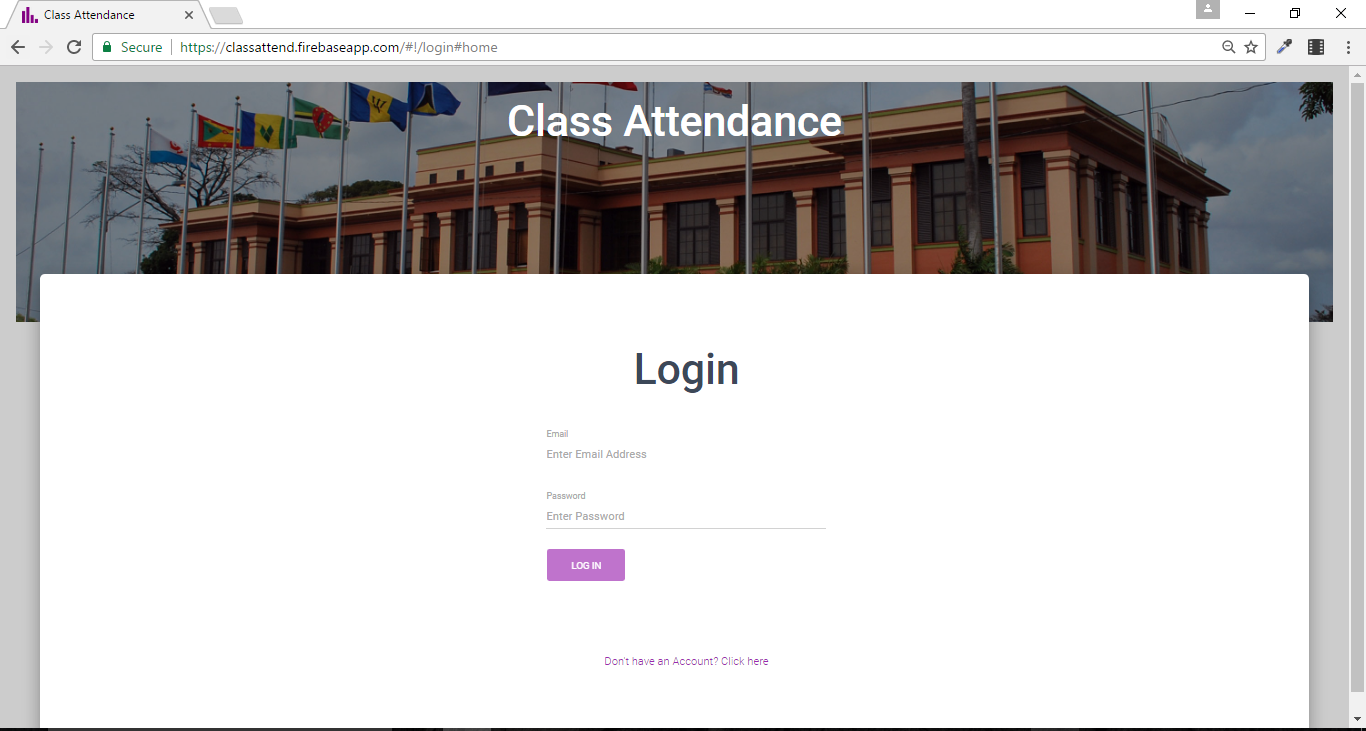


Figure 13: Login Page

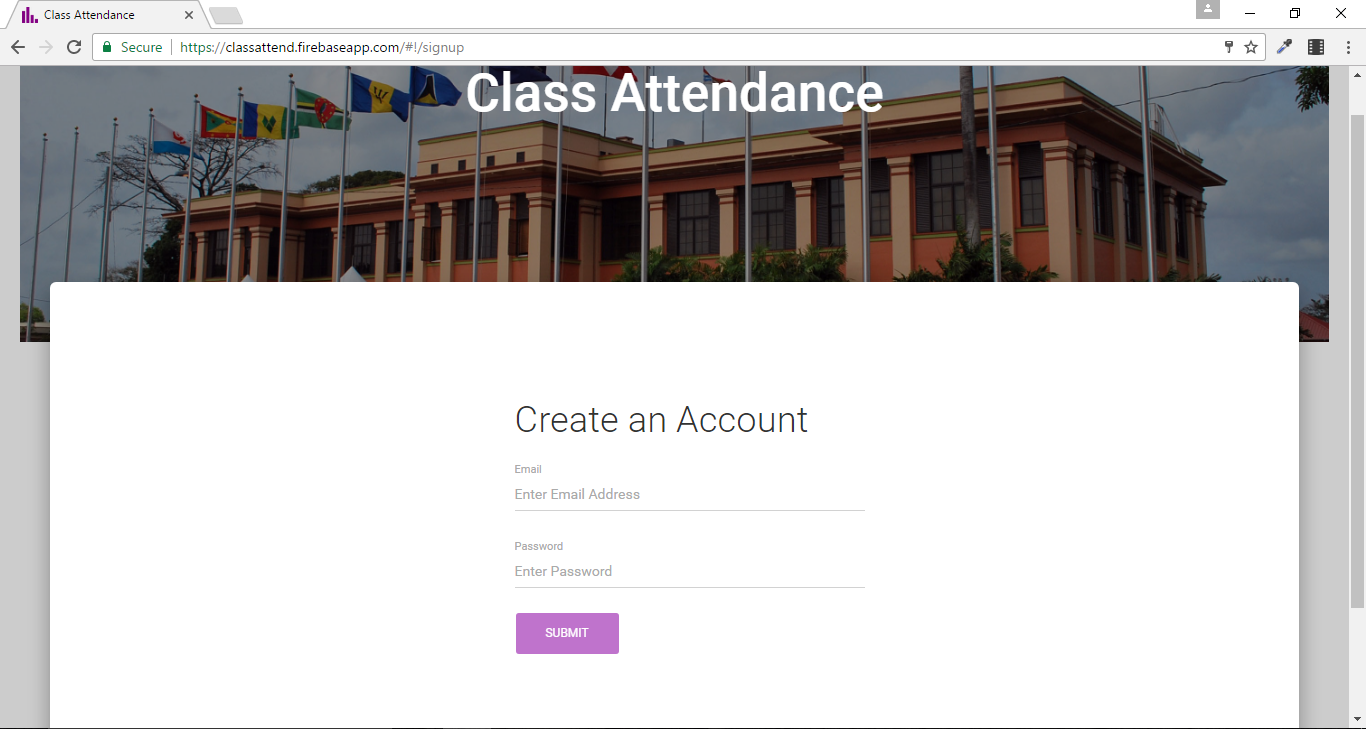


Figure 14 : Sign Up

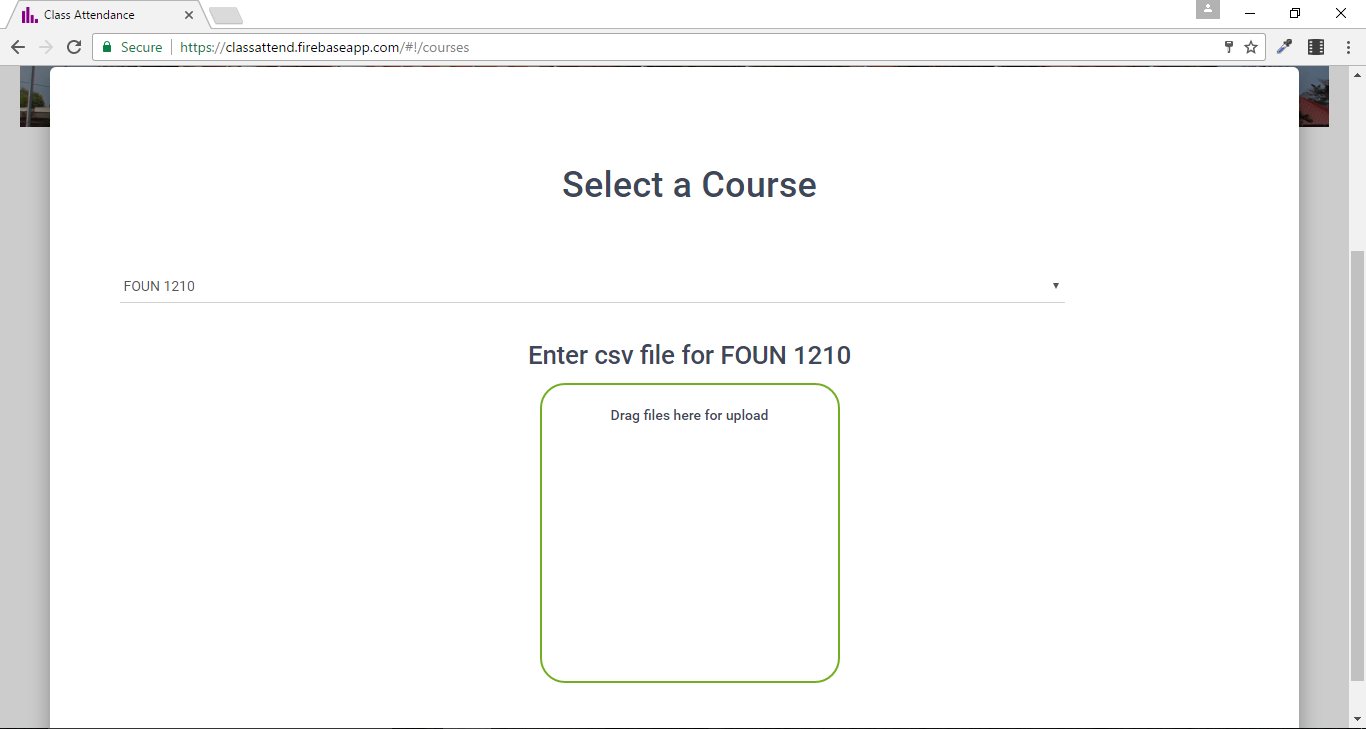


Figure 15: Upload Course page

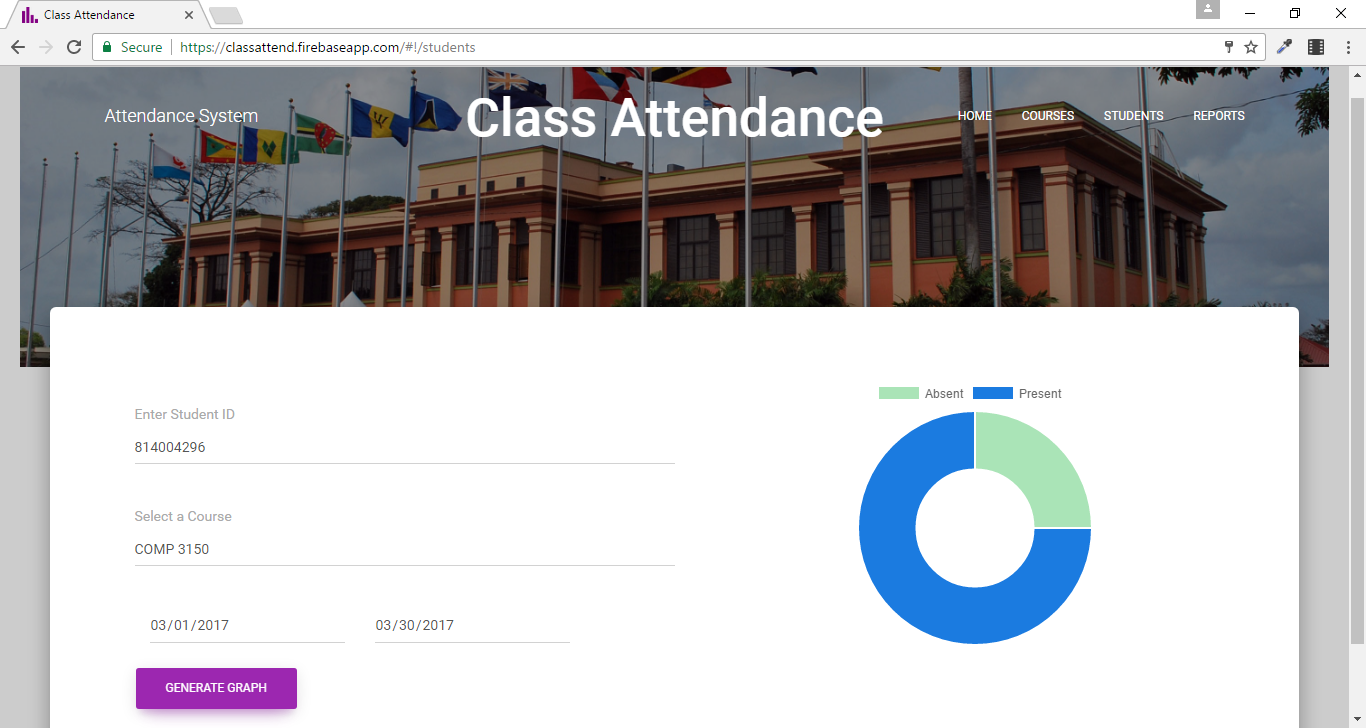


Figure 16 pie chart for student

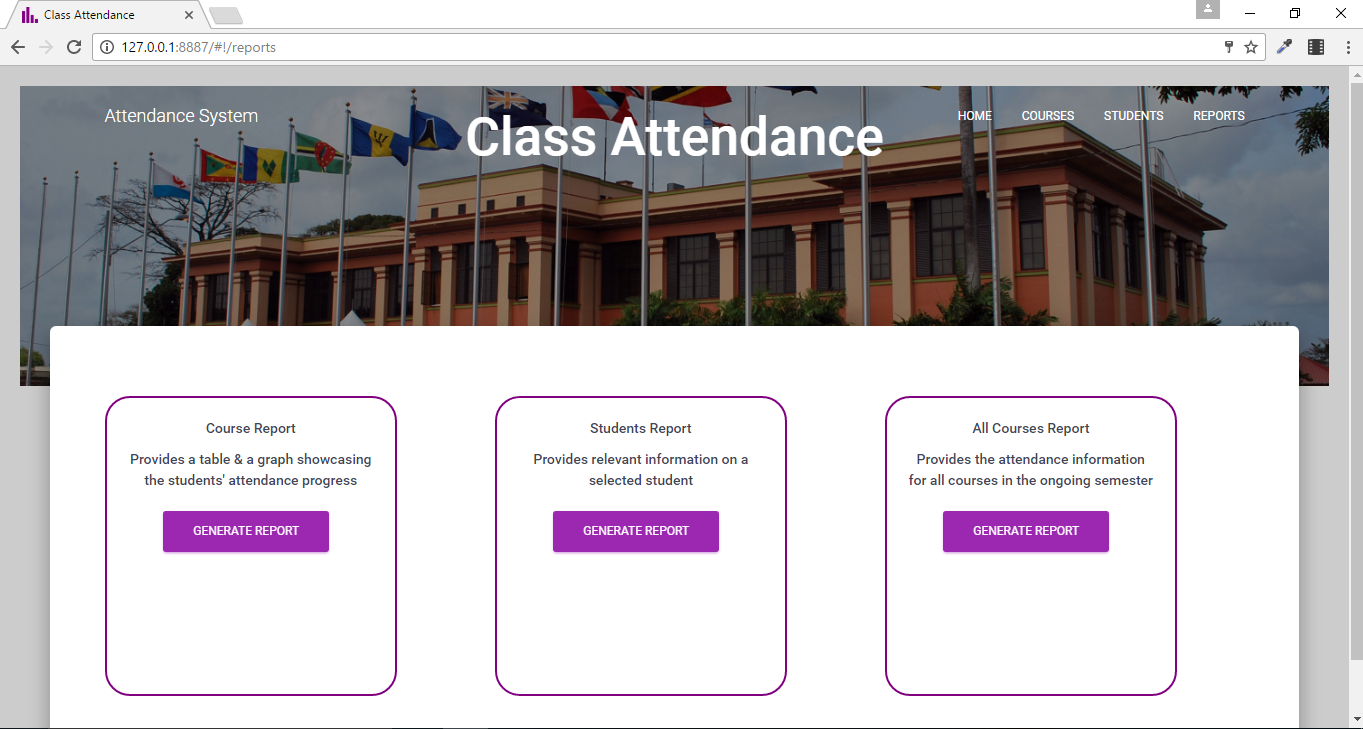


Figure 17: Reports

See Appendix for remainder of Screenshots of the Report results

## Approach and Methodology

For the Class Attendance System, there was a need to get a list of classes offered at the institute. The timetable office provided us with a data file with the relevant information. We wrote python script in order to parse the file and sift out the important data. The python script also sent the data to the database in order for the application to query it. This python script is

**Parse data**

Code Snippet 1 is from the readcsv.py file. It extracts data from the timetable.csv file and stores it in an array, dict­­\_list, to be sent to the database.

Code Snippet 1

**Android**

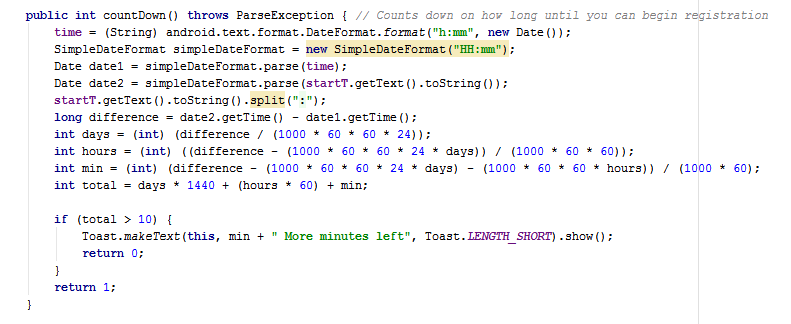
The android application was built using Android Studio. It features multiple views as it was found that the user may need to select different options in order to record attendance for a class such as the classroom and the course being taught. Outlined are some of the key coding which helps the system to work.

Code Snippet 2 was taken from the android MainActivity.class. The searchForStudent function is where the processing of the barcode information is done. It checks to see if the student is in the class chosen and notifies the student.



Code Snippet 2

Code Snippet 3 shows the countDown function. This function is to block the user from starting the class attendance barcode scanner before the class is ready to start. It provides a window period where you can start the class at most 10 minutes before it starts.



Code Snippet 3

**Web Application**

The web application is used for the staff members to view and get reports on students class attendance. A security feature of short login was also implemented because of the sensitivity of the data on this page. Therefore onces refreshed the user will be asked to login to the system again.

Code Snippet 4 was used to write data to the various pdf documents. This was written to facilitate modularity and separation of concerns within the application code. Functions defined above can write text to a pdf and add tables to them, which were key functionalities in developing reports.



Code Snippet 4

The function defined above can be found in the Courses Controller of the main.js file. On the courses page, the user is required to upload a csv file to record data for all students registered for a course. After the csv data has been parsed, the function above takes the data retrieved and uploads it to the cloud database. After parsing the data, a $$hashkey attribute was added to each student object by the csv parser code. This attribute had to be removed before the data could be stored.



Code Snippet 5

Code Snippet 6 segment is used to calculate attendance data for a single student given a date range.



Code Snippet 6

# Test Plan

## Test Approach

### Android Application testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case ID | Test Case Description | Expected Result | Actual Result | Pass/Fail | Remarks |
| T001 | Student scans ID card multiple times | Records the first instance to the database | Records the first instance to the database | Pass | The system notifies the user that they are granted access recorded |
| T002 | Internet Connection drops and comes back | Data is sent to the database when the connection restores | Data is cached on device (Courses) but ID cards not sent when restored | Fail |  |
| T003 | Student not registered to class tries to record attendance | Deny the student | Displays Access Denied | Pass |  |
| T004 | Student registered to class tries to record attendance | Accept the record | Displays Access granted and records the data | Pass |  |
| T005 | Teacher selects the room they are in | Should show the courses that are taught in that room | Shows the courses taught in the room | Pass | In an ideal situation we would want the class at the specific hour show alone |
| T006 | Attendance for the class starts before the actual class starts | Should allow the lecturer start the attendance at most 10mins before the class starts | Gives a countdown if not in 10min range for class to start | Pass |  |
| T007 | User tries to start recording class data before the class is ready to start | Blocks them from starting until the class is ready | Gives a error message with a count down until the class can start | Pass | This was only implemented on the “COMP 3150” course because when marking it would not be practical to only be able to use the application at certain class times |

### Web Application testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Case Description | Expected Result | Actual Result | Pass/Fail |
| T1001 | Insufficient network connection at log in | Unable to access application | “Network error has occurred” message displayed to user | Pass |
| T1002 | Unauthorized user tries to log into system | No access granted | "Invalid password" error message displayed to user | Pass |
| T1003 | Upload file for unregistered course | User not allowed to make upload | The action is not completed as existing course was not selected | Pass |
| T1004 | Upload file of incorrect format | Provide error to user informing to use the provided format | Data added to incorrect fields, updated to database | Fail |
| T1005 | From Student page, select student that doesn’t exist | Provides error stating student does not exist | Does not allow the user to continue | Pass |
| T1006 | From the Student page, select student and course that exists | Provides user with data collected form database | Displays the graph representing the student’s attendance records | Pass |
| T1007 | In Course reports, user selects a course that does not exist | Advise user to select a course that is available | “No Data for this Course exists” error is displayed to user | Pass |
| T1008 | In Course reports, user generates a report from no input value | Inform user to enter a course to get requested data | Inform user to enter a course to get requested data | Pass |
| T1009 | In Students reports, select student that doesn’t exist | Provides error stating student does not exist | Provides error stating student does not exist | Pass |
| T1010 | Generate All Courses reports before data is loaded | Don’t allow user to generate report till all data is loaded | Report generated with the data loaded | Pass |
| T1011 | Generate All Courses Reports after data is loaded | User only allowed to generate report after loading of data | Report generated with the appropriate data values | Pass |

### Test Assumptions

* The Android device should have easy access to Internet services in order to store the collected data via the online database.
* The web application should have a reliable Internet access in order to efficiently make use of its features.

### Features Tested

* Barcode scanning and recording data
* Generating a report
* Rejecting a user if they are late for class

### Features Not Tested

* Other classes outside of CSL1 & CSL2 (Computer Science Lab 1 & 2)
* For the web application each staff member has a different course assigned to them to view
* On the android application from the Course Information page you are not allowed to go back
* We were not able to test the situation where one class can have a tutorial and a class session immediately after it making them one class.

The project can be considered a success once students are able to scan their ID cards to timestamp their attendance in a class and then providing that information in a graphical and easy to use format to the staff. Once all the tests mentioned above are implemented and passed, then the success of the project is guaranteed.

# Business Aspects

Within the University of the West Indies, there are many applications of the UWI ID cards, exam validations, borrowing library books, for example, but not for the purpose, of taking class attendance. A similar implementation of our system is one used by a few lecturers of the Department of Computing and Information Technology (DCIT), in which the lecturer/s would generate QR codes in order to gather the students’ class presence. Our solution would solve the problem in its entirety, extending to both students and lecturers by collecting attendance data and providing tabulated information. Due to this, the Class Attendance System has no competing system. This solution currently is geared towards DCIT lecturers and students attending classes in the computer science labs and can be enhanced to entertain others classes and departments. The solution would only require Internet access in the classrooms, so the device can store the gathered data online. Since UWI, St. Augustine already has available Wi-Fi throughout the campus; the only major expense would be to provide an Android device in each of the classrooms, as the lecturers could use any device capable of using the Internet whether belonging to them or being provided to them.

# Individual Contributions

Kerschel James – Kerschel was the main creator of the android application to allow users to scan their barcode. He was also the team leader to ensure that operations were done in a timely manner. Task that were unsure of by team members were redirected towards him for further clarification and assistance. Kerschel also assisted on the web application design, which had to be redone using the Angularjs framework. Kerschel created a script to randomly add data to the firebase database for relevant testing of the system.

Rondell Gulston – Rondell has contributed to the success of the project by setting up the angular framework connections with the server and by providing a design for the web application. He also helped in the creation of the firebase database structure.

Gerard Rique – Gerard worked with both the web and the database. He was able to do the functionality of the web application and help in the creation of the database structure. The reports was his main job to complete.

# Financial Considerations

Project Budget

|  |  |
| --- | --- |
| Work Type | Estimated Cost (%) |
| Research and Design | 5 |
| Development and Testing | 40 |
| Production | 35 |
| Project Management | 10 |
| System Administration | 10 |

Cost Projections

|  |  |
| --- | --- |
| Criteria | Cost ($) |
| Requirements | 1,500 |
| Firebase w/ Domain Name & Hosting | 1,029 / year |
| Hardware | 2000/ classroom |
| Maintenance (per class) | 1,500 / year |
| Risk | 1,000 |
| Employees (5) | 6,000 |
| TOTAL | 12,029 |

# Conclusion

The Automated Class Attendance System provides a solution to the everyday problem of teachers and students of the University of the West Indies, providing a means to easily track students’ class attendance. Students would now be able to use their UWI IDs for attendance registers instead of having to provide signatures. Members of staff would have the ability to store and access the students’ information online in an appropriate format, giving quick and understandable reports. This solution will provide an easy transition of data gathering from class to class for students as well as lecturers. Although our initial solution was to use a physical barcode scanner, creating this android application made the system work much better as it allows the user to choose the necessary fields such as the course for attendance.

## Future Work

This solution provides information only on classes that belong to the Department of Computing and Information Technology, that occur in the Computer Science Labs 1 and 2. The system was only implemented in these areas to provide a proof of concept and can, in the future be applied to other classrooms and departments throughout the UWI, St. Augustine campus and to any other interested campuses. Additionally, a notification center can be applied to the project where, when a student misses a large portion of their classes an email could be automatically sent to them as a warning.

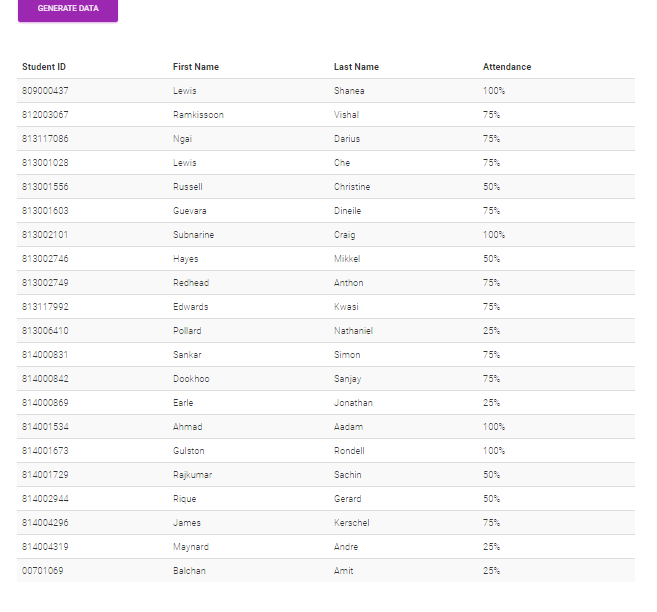
# References

UWI ID card sensor technologies : <https://sta.uwi.edu/idcards/IDcardFAQ.pdf>

Firebase database Pricing : <https://firebase.google.com/pricing/>

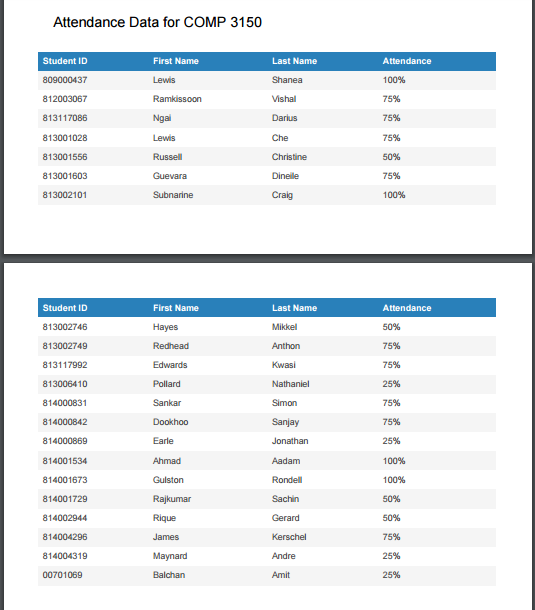
# Appendix

Course Report

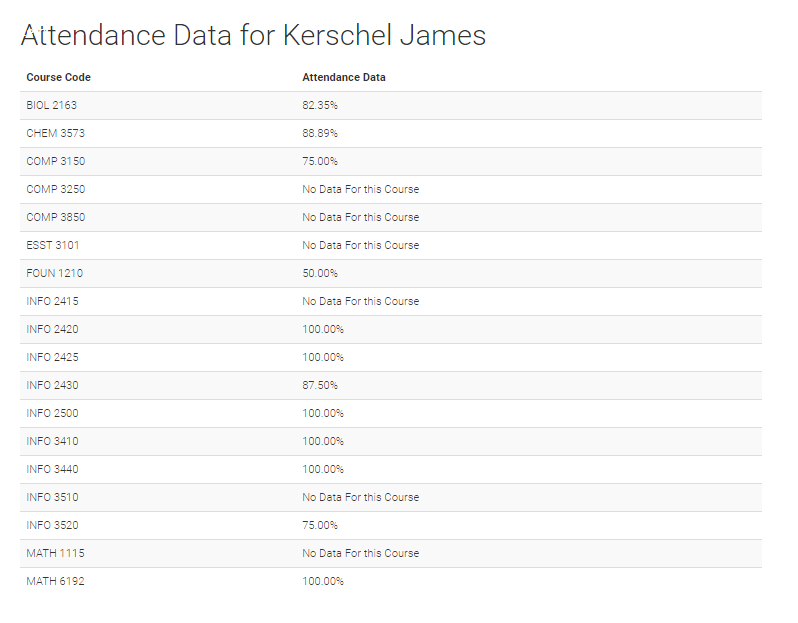




Course Report as PDF



Student Report



Student Report PDF

All Courses Report

