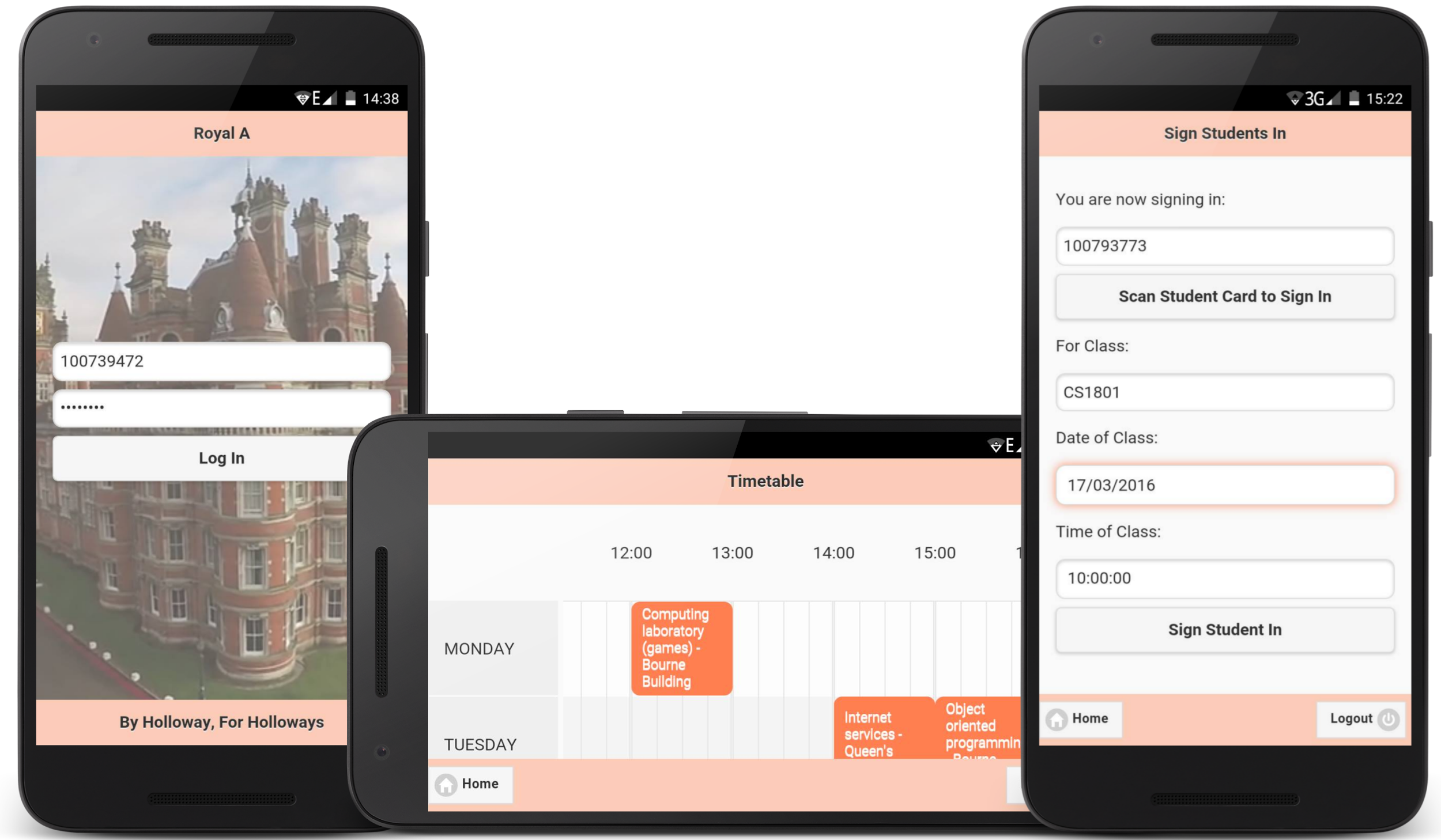
CS2810

**Team 6 Final Report**

horizontal line



Team Members:

Kok Kiat Han 100793773

Daniel Coles 100790104

Shaofeng Liu 100793482

Jiha Kim 100769889

João Nuno Mota 100795238

Hsin Je Chen 100781328

Bartal Veyhe 100801274

**DOCUMENT INFORMATION & CONTROL**

**Document Information**

|  |  |
| --- | --- |
| **Prepared By** | **Kok Kiat Han, Shaofeng Liu, Daniel Coles, Jiha Kim, João Nuno Mota** |
| **Reviewed By** | **Kok Kiat Han** |
| **Date Prepared** | **20 March 2016** |

**Distribution List**

|  |  |  |  |
| --- | --- | --- | --- |
| **Copy No** | **Name** | **Appointment** | **Organisation** |
| **1** | **Kiat Han Kok** | **Member** | **Royal Holloway, University of London** |
| **2** | **Dr Carlos Matos** | **Lecturer** | **Royal Holloway, University of London** |
| **3** | **Mr Nuno Barreiro** | **Lecturer** | **Royal Holloway, University of London** |
| **4** | **Dr Johannes Kinder** | **Lecturer** | **Royal Holloway, University of London** |

# **Table of Contents**

[Table of Contents](#h.m918gjqszl52)

[Statement of Relative Contribution](#h.i9gp6kuix2ln)

[Introduction](#h.5e31qrdqszo5)

[Purpose of Document](#h.wrztmuj4vdp2)

[The Scope](#h.nbg45dabsuf1)

[Why Royal-A?](#h.5vz9yrig82qw)

[Royal-A](#h.rs2kax9atemn)

[Architecture Diagram](#h.ogt1948o5z3i)

[Entity Diagram](#h.rwq1llr47pgv)

[Mobile Application Packages](#h.zba2d29o48ui)

[Web Application Packages](#h.cjuz81z36fw)

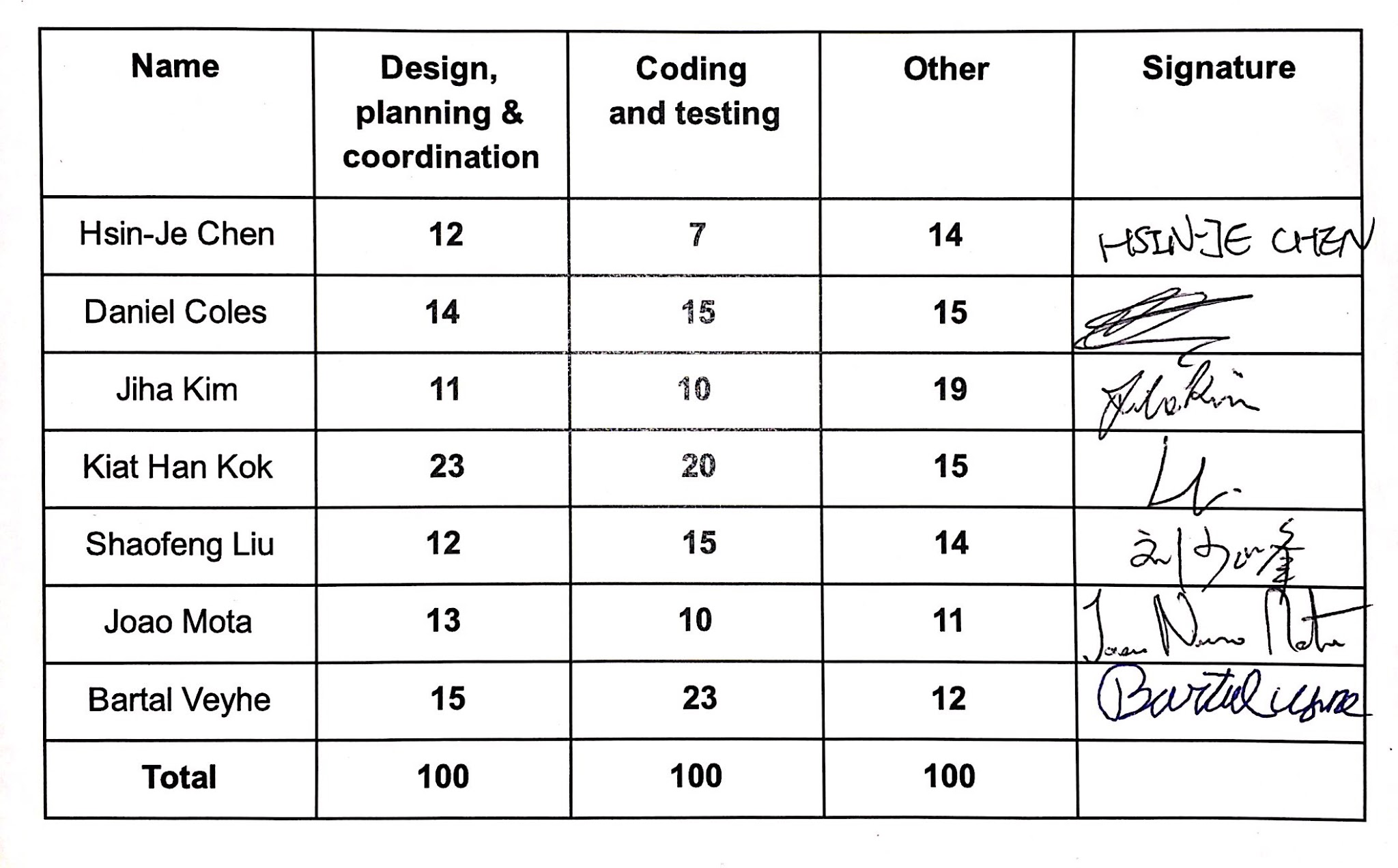
[Installation and Setup](#h.ojisr785l0qf)

[Mobile Application](#h.nlgmwx4h1ij6)

[Web Application](#h.205laqr37809)

[List of Completed User Stories](#h.143vkiutf9iu)

# Statement of Relative Contribution



# **Introduction**

## **Purpose of Document**

This document is to summarise and document the technical aspects of CS2810 Team Project, Royal Holloway Attendance System, (Royal A). It will contain the instructions for the installation of the solution. It also addresses the shortcomings, recommendations and continuity.

## **The Scope**

The scope of this document is to address the following:

· Summary of Royal A and the architecture

· Instructions on setup

· List of user stories completed

## **Why Royal-A?**

The current student attendance system is slow and easily taken advantage of by non-attending students. With an average of 13 hours of lectures per week and 300 students in all 4 years of the course, leading to a huge amount of manual data entry. An automated system would allow for an easy and efficient process to record the attendance of all students.

## **Royal-A**

In this project we developed a solution that is easily accessible to all students attending the lectures. It is integrated with a backend system for the course administration showing statistics on the student attendance.

A student attending a lecture can confirm their attendance using their smartphone to scan a QR code presented by the lecturer within the duration of the lecture.Security measures, like unique QR code and Geo-fencing restrictions, are in place to limit the abuse of the attendance system. They would be able to view their weekly time tables and their attendance. Students in lectures will then be able to access a “Question & Answer” feature that allows answering of questions asked by lecturer. This is a software version of a clicker-style solution, yet more cost effective and adaptable. Given the data required for this implementation students will also be able to access a timetable that will display the lectures they have.

Administrators would be able to manage all the systems including adding/removing students from lists and courses, input necessary information associated to the course such as lecture venues and course leaders and so on. Lecturers would be able to manage their courses and view student attendance. All users would have their personalized timetable.

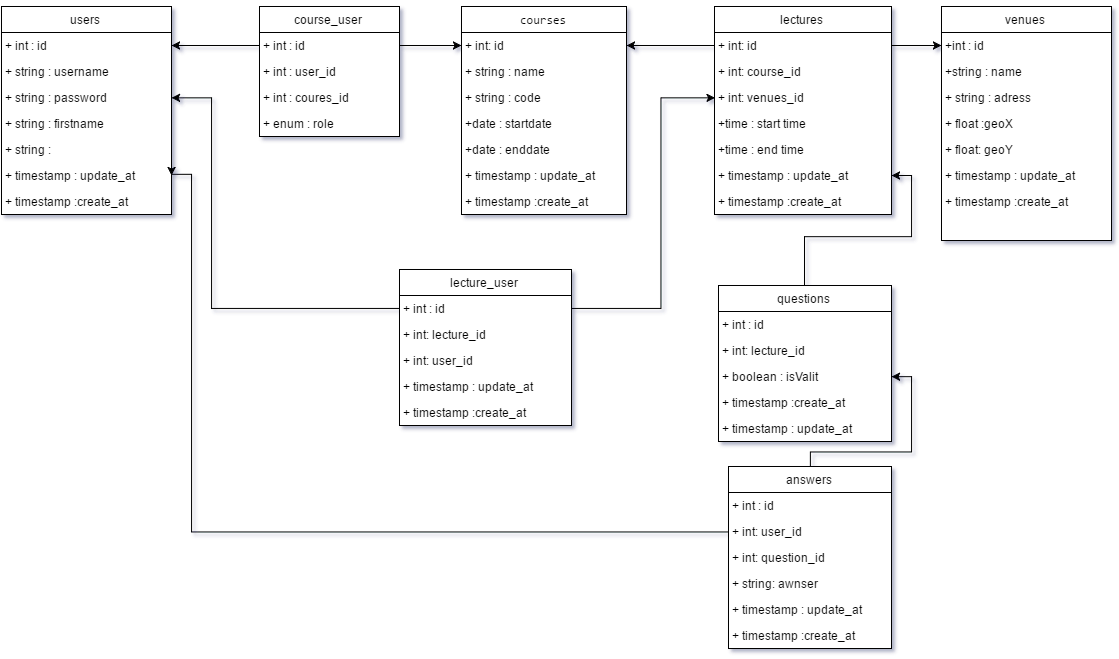
# **Architecture Diagram**

We utilize a server-client architecture for our project, where all the clients would connect to the server via web services.

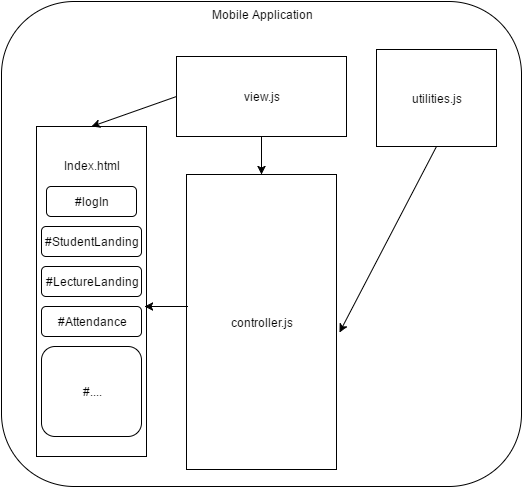
# 

# **Entity Diagram**

The entity diagram belows shows the layout of the entity and it’s relationship in our project.

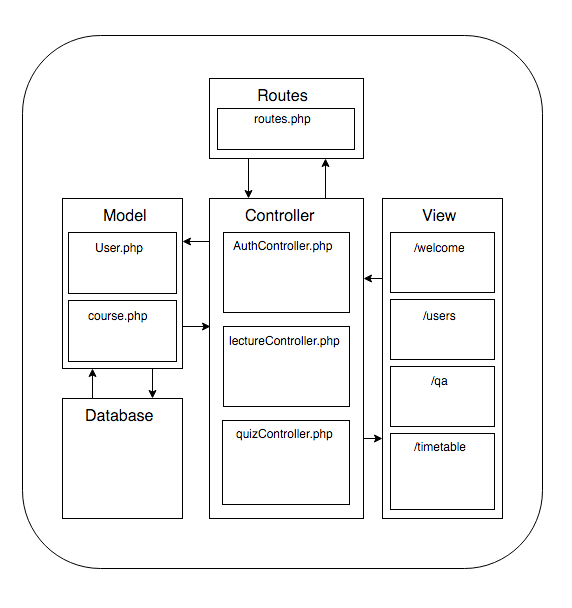


## **Mobile Application Packages**



In the mobile application, the application is refactored into a few major files. According to Apache Cordova, it attempts to replicate a real mobile application. It tries to avoid loading of a new HTML, that will look like a mobile web application. Therefore, it stores all different screens into different div tags in a single *index.html* file. *Index.html* contains the layout of all the screens in the mobile application. *View.js* sets all the actionlistener for all the buttons and global wide functions. *Controller.js* is the main file that contains all the implementation of the functions in the application. *Utilities.js* contains independent functions that assists the application, but is not related to the application, like calculating distances between 2 points.

## **Web Application Packages**



The web site keeps all of the files that consist of either the view, controller or model in their respective directories. This prevents the size of the files from being too big and facilitates the running of the app. Furthermore, the *routes* file that takes the url’s submitted either through get or post methods and directs them to the correct file in the controller directory. Other than the main files, there is the rest of the laravel framework that supports the languages used in the files.

# **Installation and Setup**

## Mobile Application

1. Install Node.js
2. Install git client
3. Install apache cordova using “sudo npm install -g cordova”
4. Clone project directory, and go into Cordova project directory
5. Run “cordova platform add ios/android” in the command line
6. Run “cordova run ios/android” in the command line to run the application

## Web Application

1. Install Composer with the code snippet provided in https://getcomposer.org/download/
2. Install Laravel using the command: composer global require "laravel/installer"
3. Download and install PostgreSQL from <http://www.postgresql.org/download/>
4. Install Git client
5. Clone project directory, into any directory you wish to use
6. Edit .env file to accommodate your system
7. Migrate tables using the following command in the web-server directory:

php artisan migrate:refresh --seed

# **List of Completed User Stories**

**Sprint 1 (4 completed)**

i. Setup development environment for web and mobile

ii. Learning Laravel 5.1, and going through PHP tutorial

iii. On Apache Cordova, finding tutorials and complete them

iv. Set up test cases for web and mobile

|  |  |  |
| --- | --- | --- |
| **1** | User Story | Setup development environment for web and mobile |
| Description | In this user story/task, we decided on the environment, programming language, and tools that we are going to use. The necessary development tools and environment are setup on all the developer working stations. |
| Task(s) | Download and install all required development tools. |
| Test(s) | All members are able to develop on their workstation with the required tools. |
| **2** | User Story | Learning Laravel 5.1, and going through PHP tutorial |
| Description | In this user story/task, members in web team went through Laravel 5.1 and PHP tutorials. |
| Task(s) | Watch the videos on laracast and complete the tutorial on the laravel website. |
| Test(s) | All members in web team are able to develop the programme using Laravel and PHP |
| **3** | User Story | On Apache Cordova, finding tutorials and complete them |
| Description | In this user story/task, members in mobile team went through Apache Cordova tutorials |
| Task(s) | Complete the tutorial on the apache cordova website. |
| Test(s) | All members in mobile team are able to develop the programme on mobile using Apache Cordova |
| **4** | User Story | Set up test cases for web and mobile |
| Description | Tests cases are required the mobile and web application |
| Task(s) | Complete a list of test cases to be done for web and mobile. |
| Test(s) | Check all the test cases are made for web and mobile |

**Sprint 2 (3 completed)**

Web Team

i.As a lecturer, I would want to be able to sign in to my application

Mobile Team

ii. As a student, I would want to be able to sign in to my application.

iii. As a lecturer, I would want to be able to sign in to my application.

|  |  |  |
| --- | --- | --- |
| Web Team | | |
| **1** | User Story | As a lecturer, I would want to be able to sign in to my application. |
| Description | Most functions can be done by user who sign in to his/her application |
| Task(s) | Create login forum |
| Receiving JSON |
| Handling sign in information |
| Sign in to mobile app |
| Set up restful API for sign in |
| FORM input for sign in |
| Test(s) | Check that right username and password passes JSON |
| Check that wrong user or password gets rejected JSON |
| Check that SQL injections do not pass |
| Check that page exists |
| Check that wrong user or password gets rejected form |
| Mobile Team | | |
| **2** | User Story | As a student, I would want to be able to sign in to my application. |
| Description | A student will be able to sign into the mobile application and presented with the student landing page. |
| Task(s) | Student signs in and launches student screen with the correct username and password |
| Test(s) | Student fails to login with wrong username and password |
| Student logins with correct username and password and presented with student screen |
| **3** | User Story | As a lecturer, I would want to be able to sign in to my application. |
| Description | A lecturer will be able to sign into the mobile application and presented with the lecturer landing page. |
| Task(s) | Lecturer signs in and launches student screen with the correct username and password |
| Implement SHA256 for the password |
| Test(s) | Lecturer fails to login with wrong username and password |
| Lecturer logins with correct username and password and presented with lecturer screen |

**Sprint 3 (3 completed)**

Web Team

i. As a student, I would want to be able to see the weekly timetable.

Mobile Team

ii. As a student, I would want to be able to sign in to a lecture.

iii. As a student I want to be able to view my weekly timetable of lectures.

|  |  |  |
| --- | --- | --- |
| Web Team | | |
| **1** | User Story | As a student, I would want to be able to see the weekly timetable |
| Description | A student can see the weekly timetable which is updated |
| Task(s) | Model |
| Database |
| JSON |
| Create visual template |
| Setup API calls for mobile team |
| Test(s) | Student is displayed with the correct timetable on the website. |
| Mobile Team | | |
| **2** | User Story | As a student, I would want to be able to sign in to a lecture. |
| Description | A student should be able to sign in to a lecture for attendance |
| Task(s) | Retrieves the current class that the student is going to sign in |
| Signs in to lecture |
| Implements the QR code reader |
| Scans the lecture QR code |
| Implements the geolocation API |
| Checks that student is at the right place |
| Write the sign in protocol to talk to the server |
| Test(s) | User at the right location can sign in successfully into the lecture |
| User at the wrong location cannot sign in |
| User who scans the wrong QR code cannot sign in |
| **3** | User Story | As a student I want to be able to view my weekly timetable of lectures |
| Description | The user should be able to view a timetable of the present week’s lectures on the timetable page of the application. |
| Task(s) | Retrieve timetable data via an AJAX call to the database hosted on our server. |
| Present the retrieved data in the timetable JS JQuery plugin |
| Test(s) | The timetable page displays all lectures for courses which the user is enrolled on. |
| Timetable displays correct lecture information |

**Sprint 4 (7 completed)**

Web Team

i. As a manager I would want to be able to sign in to system.

Mobile Team

ii. As a student i want to be able to view my upcoming lectures for the current day and next day.

iii. As a lecturer, I would want to be able to add/remove students from attendance in a lecture.

iv. As a student, I would want to be able to sign in securely.

v. As a lecturer, I would want to be able to sign in securely.

vi. As a lecturer, I would like to view the percentage of the student attendance

vii. As a student, I would want to be able to check my timetable for detailed venue location and

the lecturers teaching

|  |  |  |
| --- | --- | --- |
| Web Team | | |
| **1** | User Story | As a manager I would want to be able to sign in to system. |
| Description | A user should be able to sign in to system to manage |
| Task(s) | Login to system |
| Logout system |
| Implemented UI on blades |
| Test(s) | Login to system |
| Logout from system |
| Mobile Team | | |
| **2** | User Story | As a student i want to be able to view my upcoming lectures for the current day and next day. |
| Description | Produce a ‘What’s next’ page within the app displaying timetable information for next 2 days |
| Task(s) | Retrieve timetable data via an AJAX call to the database hosted on our server. . |
| Display data in presentable manner using CSS and JavaScript |
| Test(s) | A student can view their upcoming timetabled events. |
| **3** | User Story | As a lecturer, I would want to be able to add/remove students from attendance in a lecture. |
| Description | As students may at times, forget to bring their mobile devices. Lecturers have the option to sign students into lectures |
| Task(s) | Add student to attendance |
| Remove student from attendance |
| Create the view for lecturer sign in student |
| Test(s) | Add correct student to attendance |
| Remove correct student from attendance |
| View sign in students |
| **4** | User Story | As a student, I would want to be able to sign in securely. |
| Description | Security measures like SQL injection prevention, and session token are required to prevent cyber attacks on the system. |
| Task(s) | SQL injection prevention |
| Implement session token |
| HTTPS |
| Test(s) | Prevents OWASP TOP 10 attack |
| **5** | User Story | As a lecturer, I would want to be able to sign in securely. |
| Description | Security measures like SQL injection prevention, and session token are required to prevent cyber attacks on the system. |
| Task(s) | SQL injection prevention |
| Implement session token |
| HTTPS |
| Test(s) | Prevents OWASP TOP 10 attack |
| **6** | User Story | As a lecturer, I would like to view the student attendance |
| Description | Lecturer would want to view the current attendance |
| Task(s) | View student attendance in percentage |
| Lecturer can set class details parameters to view which class attendance they want to see |
| Talk to server to retrieve class attendance |
| Retrieve current class and set the parameters |
| Auto create lecture instances |
| Test(s) | Display student attendance properly |
| Retrieves current class details |
| **7** | User Story | As a student, I would want to be able to check my timetable for detailed venue location and the lecturers teaching |
| Description | Timetable is essential for all students. With this function, a student will be see his schedule for the week. |
| Task(s) | View a weekly timetable with detailed information |
| Test(s) | Correct timetable information is displayed |

**Sprint 5 (12 completed)**

Web Team

i. As a manager I would want to be able to add user to system

ii. As a manager I want to be able to remove user from system

iii. As a lecturer I would like to start a question when I want to ask a question to the class

(server side)

iv. As a school administrator, I want to sign in to the system

v. As admin, I want to be able to review aggregate/individual student attendance for each

lecture and throughout the year

vi. As a manager I want to be able to view attendance of individual or multiple users

Mobile Team

vii. As a lecturer, I want to be able to view the results/graphs of the question I posted

viii. As a lecturer, I would like to send questions for students to answer

ix. As a lecturer, I would want to be able to see the timetable

x. As a student, I want to be able to check my timetable for detailed venue location and the

lecturers teaching

xi. As a student, I want to be able to answer question that lecture post online

xii. As a student, I want to be able to view overall attendance for each session in this week

|  |  |  |
| --- | --- | --- |
| Web Team | | |
| **1** | User Story | As a manager I would want to be able to add user to system |
| Description | A manager should be able to add new user to system. |
| Task(s) | Add user to system |
| Database entry |
| Model |
| Update user |
| Delete user |
| Test(s) | Add user |
| Update user |
| Read user |
| Delete user |
| **2** | User Story | As a manager I want to be able to remove user from system |
| Description | A manager should be able to remove user which is not in use |
| Task(s) | Remove user from system |
| Test(s) | Remove user |
| **3** | User Story | As a lecturer I would like to start a question when I want to ask a question to the class (server side) |
| Description | A lecturer would want to prepare questions |
| Task(s) | Tell server to start count down and receive answer from user |
| Set up server to get start flag |
| Create table entry |
| Create model |
| Create controller |
| Test(s) | Test is server start question section |
| **4** | User Story | As a school administrator, I want to sign in to the system |
| Description | Most functions can be done by user who sign in to the system |
| Task(s) | Login user |
| Logout user |
| Test(s) | Log in/out user |
| Display and get form |
| **5** | User Story | As admin, I want to be able to review aggregate/individual student attendance for each lecture and throughout the year |
| Description | With this function, an administrator can review and see a high level view of which students are not meeting the required attendance easily. |
| Task(s) | View single student attendance |
| View attendance for course |
| View attendance by year of study |
| Test(s) | Test display individual student |
| Test display multiple student |
| **6** | User Story | As a manager I want to be able to view attendance of individual or multiple users |
| Description | Attendance of individual or multiple users should be able to be controlled by a manager |
| Task(s) | View single student attendance |
| View multiple student attendance |
| View attendance according to course |
| Test(s) | Display individual attendance |
| Display multiple student attendance |
| Mobile Team | | |
| **7** | User Story | As a lecturer, I want to be able to view the results/graphs of the question I posted |
| Description | The number of users able to answer the question should be displayed, after the question has been submitted. |
| Task(s) | Display percentage of result on screen |
| Get statistics from server |
| Send statistics to mobile |
| Model |
| Database |
| Display Q&A statistics on website |
| Test(s) | Display statistics properly |
| Receive data from server |
| Display on website properly |
| Display on mobile properly |
| **8** | User story | As a lecturer, I would like to send questions for students to answer |
| Description | A lecturer wants to send questions to be answered by students during the class |
| Task(s) | Server notice student that there is new question |
| Receive notice from server at mobile end |
| Create event that the user will look for |
| New question fire question update hook |
| Test(s) | Get noticed |
| Send message to mobile |
| **9** | User Story | As a lecturer, I would want to be able to see the timetable |
| Description | A lecturer should be able to see the timetable to check her/his schedule |
| Task(s) | Lecturer is able to see the timetable |
| Test(s) | Correct timetable information is displayed |
| **10** | User Story | As a student, I want to be able to check my timetable for detailed venue location and the lecturers teaching |
| Description | Attended lectures should appear highlighted in green on the timetable whereas unattended lectures should appear red. |
| Task(s) | Student is able to see the upcoming event in the timetable and whether or not they have attended the lecture |
| Test(s) | Information of the upcoming event is displayed accordingly to the student |
| **11** | User Story | As a student, I want to be able to answer question that lecturer post online |
| Description | Question lecturer posted online should be answered by students in the lecture. |
| Task(s) | Finish student page for answering question |
| Send answer to server end |
| Get response when succeed |
| Set up api for mobile end |
| Database |
| Model |
| Create new question |
| Create new answer |
| Add answer to user and question |
| Update answer if resubmit |
| Test(s) | Test with server response |
| Test that question after close do not get store in the database |
| Test that question inside send when database is open get store |
| Test reset answer update the answer in row in the database |
| Completed Q&A controller on server end |
| **12** | User Story | As a student, I want to be able to view overall attendance for each session in this week |
| Description | A student should be able to see overall attendance to check how many classes he/she missed in this week. |
| Task(s) | Checks if a student has attended the selected lecture |
| Display attendance information on the timetable |
| Display green if lecture has been attended |
| Test(s) | Display the correct attendance information on the timetable |