NORTHWEST MISSOURI STATE UNIVERSITY  
Student Attendance Tracking system

December 6, 2017

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| University | Northwest Missouri State University |  | Project Sponsor | Michael Oudshoorn |
| Project Name | Student Attendance Tracking System |  | Project Manager | Michael Oudshoorn |
| Client Name | Dr. Michael Oudshoorn |  | Team Name | Code Green |
| Term | Summer.  Graduate Directed Project. |  | Team Logo |  |
| Start Date | June 12th 2017 |  | Authors | Sirisha Vanamali  Rohith Babu Sadhu  Sri Sai Ram Kumar Mamidala  Vipul Reddy Madadi  Vamsi Devalla  Subba Reddy Pothireddy  Shankar Rao Vallapurapu |
| Last Updated On | 6th December 2017 |  | Version | 1.8 |

Contents

[Table of Figures 3](#_Toc500350820)

[Purpose 4](#_Toc500350821)

[Scope 4](#_Toc500350822)

[Glossary 4](#_Toc500350823)

[Basic Functional Requirements 5](#_Toc500350824)

[Detailed functional requirements list 6](#_Toc500350825)

[Non-Functional Requirements 12](#_Toc500350826)

[Static Requirements 12](#_Toc500350827)

[Dynamic Requirements 12](#_Toc500350828)

[I. Software system attributes: 12](#_Toc500350829)

[II. Design Constraints 13](#_Toc500350830)

[III. Logical Database Design 13](#_Toc500350831)

[System Core Features 14](#_Toc500350832)

[Additional Features 15](#_Toc500350833)

[Assumptions and Dependencies 15](#_Toc500350834)

[Use Case Diagram 16](#_Toc500350835)

[Module 1: Instructor Use Case Diagram 16](#_Toc500350836)

[I. Instructor: Login Successful 17](#_Toc500350837)

[II. Instructor: Login Unsuccessful 18](#_Toc500350838)

[III. Instructor: Display QR code 18](#_Toc500350839)

[IV. Instructor: Manage Students 19](#_Toc500350840)

[V. Instructor: Manage Courses 19](#_Toc500350841)

[Module2: Student Use Case Diagram 20](#_Toc500350842)

[I. Student: Login Successful 20](#_Toc500350843)

[II. Student: Login unsuccessful 21](#_Toc500350844)

[III. Student: Scanning QR 21](#_Toc500350845)

[IV. Student: View Course Details 22](#_Toc500350846)

[V. Student: Attendance Percentage and feedback 22](#_Toc500350847)

# Table of Figures

[Figure1: Instructor Use case diagram 17](#_Toc488113247)

[Figure2: Instructor Login successful Use case description 1](#_Toc488113248)7

[Figure3: Instructor Login Unsuccessful Use case description 18](#_Toc488113249)

[Figure4: Instructor Display QR code Use case description 18](#_Toc488113250)

[Figure5: Instructor Manage Students Use case description 19](#_Toc488113251)

[Figure6: Instructor Manage Courses Use case description 19](#_Toc488113252)

[Figure7: Student Use case Diagram 20](#_Toc488113253)

[Figure8: Student Login Successful Use case Description 21](#_Toc488113254)

[Figure9: Student Login Unsuccessful Use case Description 21](#_Toc488113255)

[Figure10: Student Scan QR Use case Description 22](#_Toc488113256)

[Figure11: Student View Course Details Use case Description 22](#_Toc488113257)

[Figure12: Student Attendance Percentage and Feedback Use case Description 23](#_Toc488113258)

Introduction

# Purpose

The purpose of developing attendance tracking system is to computerize the traditional way of taking attendance. Another purpose of this system is to reduce the complexity of taking attendance for the students by the instructor and to eliminate duplicate data entry, errors in time and attendance entries. It improves visibility to track and manage student attendance.

# Scope

The scope of this project is to track the student attendance by generating QR code. To elaborate, we have provided more functionalities to the instructor, they can view the student attendance as well as they can see the number of students present in class on a day, they can manage courses as per CRN and they can add a course likewise they can remove the course too. Students can view the attendance and can able to register to the courses. This system is limited to the Northwest Missouri State University.

# Glossary

|  |  |
| --- | --- |
| **Key Word** | **Description** |
| QR | Random code generated by the Instructor using which the students scan to mark their attendance. |
| Software Requirements Specification. (SRS) | A document that completely describes all functions of a proposed system and the constraints under which it must operate. For example, this document. |
| Stakeholder | Any person with an interest in the project who is not a developer. |
| Database | Collection of all the information monitored by this system. |

# Basic Functional Requirements

“A functional requirement defines a function of a system or its component. A function is described as a set of inputs, the behavior, and outputs.”

The main purpose of this project is to build an easy and a faster way to gain and track attendance of the students. The student must scan the QR code for his attendance, the details such as student ID number, student name, student course and semester details needs to be captured.

The requirements are listed below:

* The system must allow the user to generate the QR code
* The system should be able to scan a QR code
* The system must decode the QR code that is scanned
* The system must compare the QR code generated by the instructor and scanned by the student.
* The system should be able to connect to the database for sending and retrieving the student information
* The system should be able to show the student the registered courses
* The system should be able to allow the instructor to change the percentages of the student
* The system must allow the user to Login into the account
* The system should allow the user to register for a new account.
* The system should allow the instructor to generate one QR code for one class.
* The system should allow the instructor to check the number of students in the class
* The system should allow the user to check his/her attendance percentage.
* The system should allow the user to check the feedback of the attendance percentage
* The system must allow the instructor to generate the QR code using the CRN of the course.

# Detailed functional requirements list

|  |  |  |
| --- | --- | --- |
| **Requirement ID** | **Module** | **Functional Requirement** |
| Student | | |
| R1 | Student Login | The mobile application shall let the student to login with a correct admin ID and correct password.  1. User ID Requirements - Student must have a unique user ID with any number of character length. Can include special characters  2. Password requirements – Allows any kind of password. Doesn’t restrict to one format.  3. Encryption - All communications with external systems should be encrypted using parse server.  4. All data Username and password combination will be stored into a database for future reference. |
| R2 | Login Successful | On successful login, the student should be redirected to 'student view' page where the student should be able to capture the QR code and view attendance and courses the student is registered for during that semester. |
| R3 | Login Failure | The mobile application should let the student login with a correct username and password following the above-mentioned rules. If the student fails to login, he should get a popup message saying that username/password is incorrect. |
| R4 | Scan QR | The mobile application shall let the user scan the unique QR code displayed by the Instructor. If the user scans the right QR then  1. The application should popup a message saying that the QR code is scanned successfully. 2. The application should identify the student and save further details based on unique student ID.  3. The application should show the total attendance percentage after scanning is completed. |
| R5 | Scan QR Failure | If Scanning the QR is failed, he should get a popup saying that QR scan is failed. Then the student should be in a position to consult the instructor for the attendance. |
| R6 | Attendance Recorded Feedback | Once the QR code is scanned, the mobile application shall send a proper feedback to the user saying 'QR code scanned and total attendance should change accordingly. |
| R7 | View Courses | The mobile application shall let the student view the courses he/she has registered to in the department. This view should be made available when user clicks on 'View Courses' menu. The view is populated with data from the course table and student table. |
| R8 | View Attendance Percentage | The mobile application shall let the student view his/her current attendance percentage. This view should be made visible only when the student clicks on View 'Attendance percentage' menu. The view is populated with data from the attendance table and student table. |
| R9 | Student Logout | The mobile application shall let the student logout of the application. If the user clicks on the logout button, he/she should be in a position to view a popup showing that the user logged out of the session and displays the login page again. |
| Instructor | | |
| R10 | Instructor Login Precondition: Instructor is logged out of the application. | The Web application shall let the Instructor login to the system with given user id and password. 1. User ID Requirements - Instructor must have a unique user ID of min 5-character length. Can include special characters. 2. Password requirements - Must be 5-character length, with no further limitations.  3. Encryption - All communications with external systems should be encrypted using a parse server. 4. All data Username and password combination will be stored into a database for future reference. |
| R11 | Login Successful | On successful login, the Instructor should be redirected to 'Instructor View' page, where he/she can able to view courses, and student’s assigned to that course and will be able to scan QR code generated by the instructor. |
| R12 | Login Failure | If instructor fails to login, he should get a popup saying that there is an incorrect username/password. |
| R13 | Add Course | The Web application shall let the Instructor to add courses across departments. The instructor should be able to select and add courses from the 'Instructor view' page and the changes should be saved and preserved in the instructor database. |
| R14 | Add sections (Feature to be thought about and implemented in the final sprint) | The Web application shall let the Instructor to add multiple sections for a course. Whenever the instructor selects a course using a CRN, multiple sections should appear and instructor should be able to add the sections to his view. This should add necessary changes to instructor database and changes have to be preserved. |
| R15 | Invite another instructor (Feature to be thought about and implemented in the final sprint) | The Web application shall let the Instructor invite another instructor. There are instances where the instructor might invite another professor to give a lecture. In this case, provision should be provided for the Instructor to invite another instructor. This should:  1.Notify the students with an email notification saying ‘This so and so instructor has been invited for the lecture tomorrow.’ 2. Notify the invited professor with an email notification. |
| R16 | Generate QR | The Web application shall let the Instructor generate a unique QR code when the instructor clicks on 'Generate QR button' on the Instructor view. The instructor is redirected to 'display QR Page'. The display QR page has the provision to select the subject from the dropdown, no of QR’s required dropdown, QR time active dropdown and time interval dropdown. The QR will be generated only when the instructor fills all these dropdowns. The dropdowns default to a number to make instructors job easy. And once the instructor clicks on generate QR button, these selection details are saved on to the QR database and The timestamp at which the QR is generated should be saved onto QR table and should be active for not more than selected 'timer' by the instructor. |
| R17 | Display QR | The Web application shall let the Instructor display that unique QR code to the students in the class. This 'display QR' page will have a button saying 'Generate another QR code' which will redirect the instructor to 'generate qr' page. |
| R18 | Select Timer | The Web application shall let the Instructor choose the time interval from the 'Time Interval between QR' dropdown. Will default to 20minutes. Unique QR code will be generated and displayed to the students in the class after the instructor choice of time Interval. These dropdown details are saved onto the QR database. |
| R19 | Select the no of QR's | The Web application shall let the Instructor choose the number of QR's to be generated by the application in the 'generate qr' page. The Web application shall let the Instructor generate as many as unique QR codes according to Instructors choice. These details should be saved onto the QR table and the QR codes should be generated after the chosen time interval. |
| R20 | QR Active Time | The Web application should let the instructor choose the QR active time from the 'generate qr' page. This should be a dropdown defaulting to 2 mins. Once the instructor clicks on 'generate qr', this field should be saved onto the QR table. |
| R21 | Home Page button | The Web application shall let the Instructor redirect to the 'instructorview' page when the instructor clicks on home page button on 'generate qr' page. |
| R22 | Modify Attendance | The Web application shall let the Instructor modify student attendance records. On the 'instructorview' page, there should be a button named 'Student' which should redirect the instructor to student view page where the instructor can mark attendance of particular student and finally changes must be saved onto attendance database. |
| R23 | Delete Attendance | The Web application shall let the Instructor delete student attendance records. On the 'instructorview' page, there should be a button named 'Student' which should redirect the instructor to student view page where the instructor can unmark attendance of particular student and finally changes must be saved onto attendance database. |
| R24 | Analyze Attendance | The Web application shall let the Instructor analyze student attendance records. There should be a 'Student' button on 'instructorview' page. When the instructor clicks on it, the instructor is redirector to 'studentattendancetracking' page where an analysis of the student’s attendance over the semester is displayed. This analysis is presented as a bar chart over time and attendance percentage. |
| R25 | No of students | The Web application shall let the Instructor view the number of students in each course. The instructor can view the no of students registered for the course when the instructor clicks on 'course' button on the 'instructorview' page. This action will retrieve no of students registered for a course from the course table and display that on screen. |
| R26 | View Attendance Count | The Web application shall provide the appropriate count of the students in the class. The instructor can view the no of students who could capture the QR successfully when he clicks on 'student' button on the 'instructor view' page and then click on 'view present' button. This will retrieve the no of QR captures from the QR database and displays on the screen. |
| R27 | Feedback | The Web application shall provide appropriate feedback to the Instructor after modifying any student records. The Instructor is notified saying 'Attendance records modified' whenever the student attendance records are modified successfully. |

# Non-Functional Requirements

**“**A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors.”

**Performance**: Easy tracking of attendance can be done by the instructor effortlessly.

Static Requirements**:** These requirements do not impose any constraints on the execution characteristics of the system. They are:

1. Number of terminals: This system makes complete use of a database, while the front end will be available in the form of an application of the student and a web application for the instructor.
2. Number of users: The number of users for this system are mainly three they are the student, instructor and the admin who can also be referred as the department head.

Dynamic Requirements: These specify some of the requirements which may be changed based on the constraints or the limitations of that particular requirement. They typically include response time and throughput of the system.

1. Software system attributes:
2. **Availability**: This system is available for the users of any educational institutions such as the colleges and high schools for marking the attendance of the students in an easy and flexible way.
3. **Security:** This system is highly secure since it allows the instructor to generate multiple QR codes and also ask for an image of the QR code scanned by the student to ensure that the student is in the class and also reduces the abuse of the system.
4. **Portability**: This system is an application which can be installed in any system satisfying the system requirements of the software.
5. **Reliability:** The system will not crash on invalid data. If data is not recognized, the system will ask the user to scan the QR code in the correct format or the system will produce an error. The data will be stored on a nonvolatile storage device such as a hard drive, so that the data can be retained when the system is shut off. If there is an error the system will ask the user to scan the code again or seek assistance. The user can generate different codes any number of time changing the time of generation of the QR.
6. **Maintainability:** Backups for the database can be done for future use.
7. Design Constraints

**User Interface and Human Factors**

More than one user can use this system simultaneously since there are many students present in class, all should scan the attendance with in the time frame. The usage of this system is easy to use and easy to handle. We are going to have all the constraints so that student cannot abuse the system.

**Hardware considerations:**

We would recommend using the hardware system with the latest configuration for the fast and easy access of the applications and for the students to scan the QR code we recommend to use an iOS applications with the latest update.

**Performance Characteristics:**

There are neither throughput, speed nor the response constraints for the proposed systems. There are no size or capacity constraints for the system.

**Error handling and extreme conditions:**

If there are any input error in the systems, then the system provides feedback to the user specifying the error. If there are any extreme conditions, then the system would provide notification to the user.

1. Logical Database Design

There are five main data entities. The following classification is in no way a suggestion of design but rather a logical classification of those data entities as well as their attributes.

|  |  |  |
| --- | --- | --- |
| **Data** | **Attributes** | **Use** |
| Student input data | * Student ID * Student Name * Contact Number * Student Attendance | This data is used to track the student attendance an also able to view his attendance percentage. |
| Instructor data | * Instructor Name * Instructor ID * Instructor Contact | This data is used to determine the Instructor details and to know the time interval between the QR codes. |
| Course | * Course ID * Course Name * Course Description * Course Semester * Course Schedule | This data is used to provide details and timings of the course for a department. |

# System Core Features

1. **User Sign in and Welcome:** Allow the user to sign in with the application. With the acceptable credentials and with a valid password with all the constraints specified.
2. **Scan QR**: The user should can scan the QR code which is generated by the system. Student should also be able to capture the QR code as soon as the student scans the QR.
3. **Manage QR:** The instructor has the flexibility of generation or deletion of the QR code as per the constraints specified such as the time at which the class meets special code for that section.
4. **Manage Student:** The instructor has the flexibility of managing the student details such as checking the student details such as name, id and other credentials of the student.
5. **Mark Attendance:** Instructor has the flexibility to mark the attendance of the student as per the request of the student if he is not able to scan the QR code for any specified reasons.
6. **Manage Courses:** Instructor has the feature for managing the courses such as they can add new courses to the system which they are likely to teach in that semester.

# Additional Features

1. **Feedback:** Whatever may be the system the user using the system would love to see if his action has been working properly or not, for such feature we provided the feedback option to the users acknowledging his/her actions.
2. **View Attendance Percentage**: One of the user i.e. Student has an extra option of viewing the attendance percentage of all the registered courses for that semester, showing the statistics on the daily basics.
3. **View number of students capturing the QR code**: Instructor has an option of checking the total number of student who have scanned the QR code for that particular session comparing the total number of student of that class.

# Assumptions and Dependencies

1. The main Assumption of the system is that we consider that all the students use mobile phone with iOS compatibility.
2. All the instructors bring their laptop for creating their web application.
3. Student should contain iPhone.
4. Student capturing the QR code is present in the class.
5. No student can carry more than one mobile device.
6. The student attending the class has registered for that class
7. The mobile device is in the working condition for capturing the QR code
8. Instructor has the proper knowledge to generate the QR code.
9. Having a valid internet connection.
10. The instructor can project the QR code on the screen.
11. Student can able to scan the QR code in straight angle.

# Use Case Diagram

The use case diagram is usually a graphical description of interactions between the elements of a system. This is also a methodology which is used in system analysis to identify, organize and clarify the requirements of the system.

##### Purpose:

The main functionality of the use case diagram is to show in what way a user can communicate with the system, this may help in developing a prototype of the system and identifying specific requirements for that particular task. A use case diagram is similar to that of a flow chart. A use case diagram mainly consists of four basic components, they are:

**Actor:** Individuals who are involved in the system, defined as per their roles.

**Use Cases:** these are said to be the specific roles played by the actors within the system or around the system.

**Boundary:** This defines the system of interest in relation with the world which is around that system.

**Relationships:** These are the specific relation between the actors and the use cases of the system.

A use case diagram generally consists of an actor and all the task which can be performed by the actor which are represented in the form of an oval called use cases. In the proposed system there are mainly three actors around which the complete system is based up on they are Student, Instructor and the Admin.

## Module 1: Instructor Use Case Diagram

This actor can manage the student and his attendance. He is also capable of updating the attendance details of the student. He has many vital roles such as managing the courses, managing the students, and displaying the QR codes. In the process of managing the student instructor can view the details of the students and check the percentage of the student and update it. In the process of managing the courses he can add or delete the courses with his own sections. He is also allowed to display the QR codes according to the time at which the class meets and the CRN number and he also has access to generate multiple QR codes based on his requirements for a specific class at any point of time.

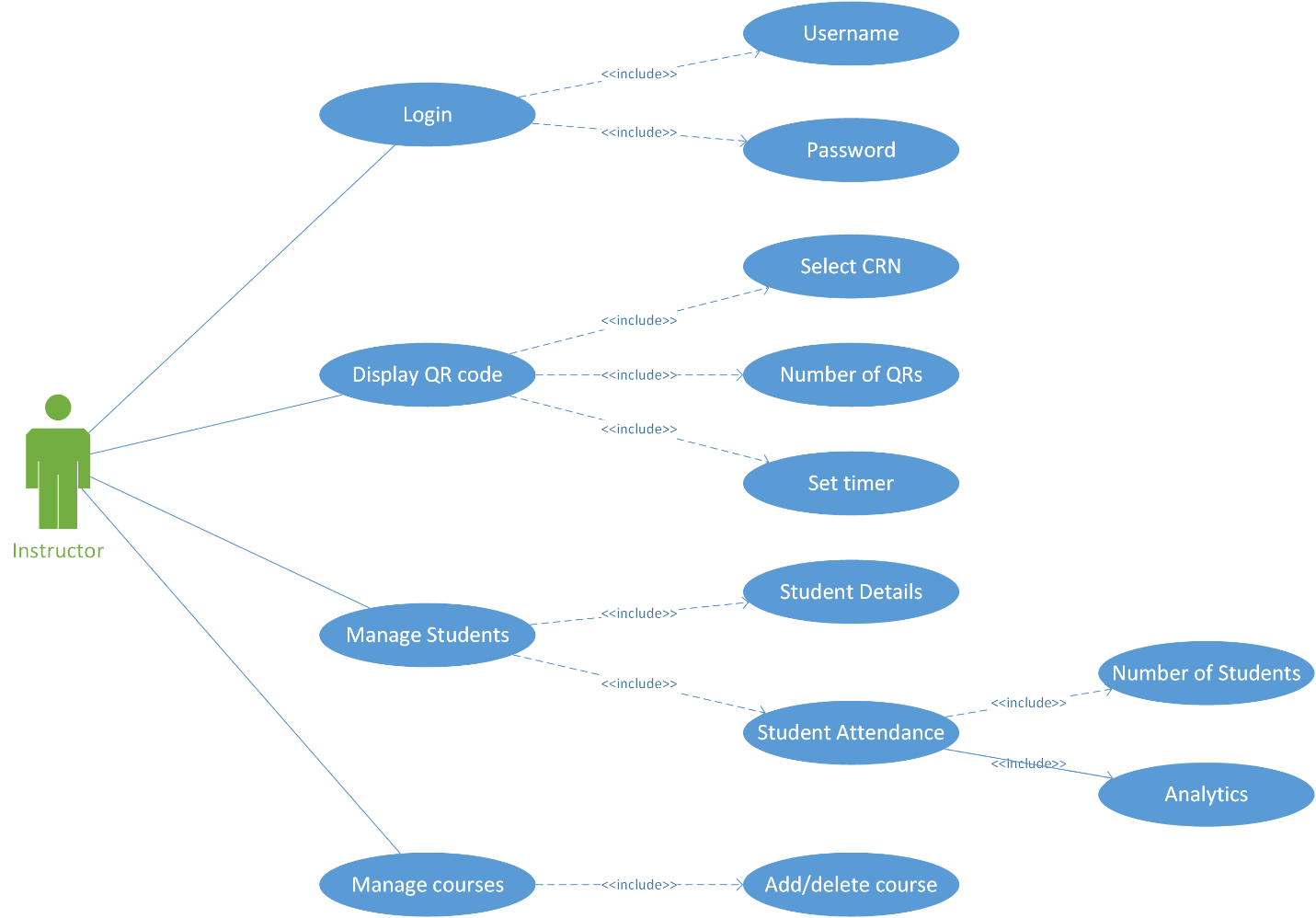


Figure1: Instructor Use case diagram

1. Instructor: Login Successful

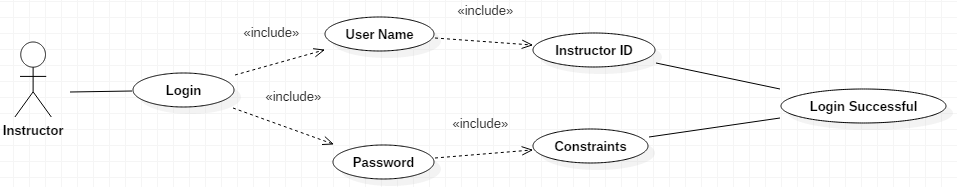


Figure2: Instructor Login successful Use case description

In the initial stage the Instructor must login with his User Name and Password. User Name of the admin is his Instructor ID and Password has few constraints. Once logged in the Instructor will get a popup as login Successful. In the Instructor view once he logged in he can see four modules like Manage Course, Manage Student and Display QR.

1. Instructor: Login Unsuccessful

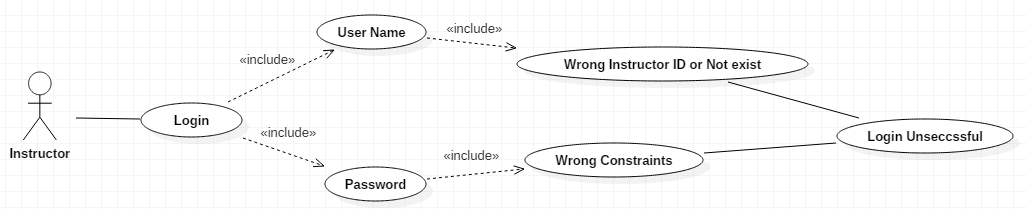


Figure3: Instructor Login Unsuccessful Use case description

In the initial stage if the Instructor enters wrong username or wrong password or if they have not entered any username or the password they will get a popup as login unsuccessful

1. Instructor: Display QR code

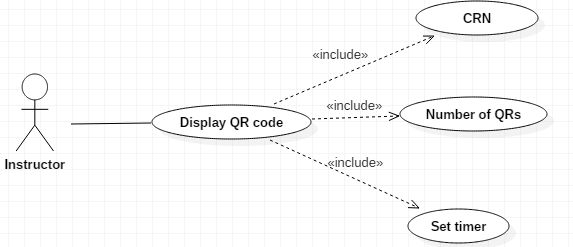


Figure4: Instructor Display QR code Use case description

In the Display QR code module, the instructor can display a unique QR code based on CRN and this QR code will be active for the time that he/she had chosen in the ‘set timer field’

1. Instructor: Manage Students

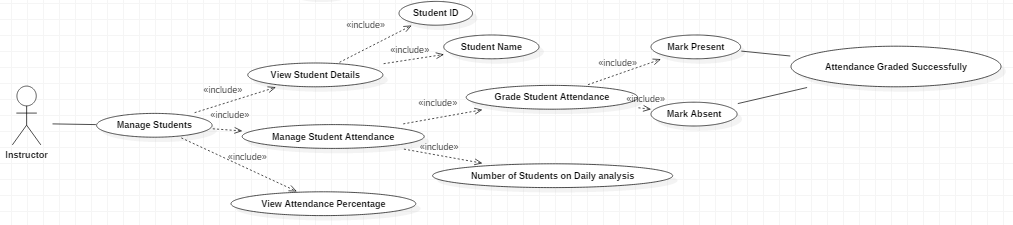


Figure5: Instructor Manage Students Use case description

In the Manage Students Module the instructor can view student details like student ID and student name, Manage student Attendance and View Attendance percentage of each student. In Manage student attendance the instructor can grade student attendance and he can view number of students present on daily analysis.

1. Instructor: Manage Courses

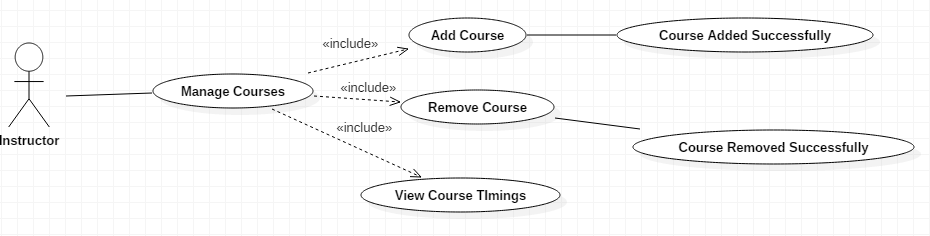


Figure6: Instructor Manage Courses Use case description

In the Manage Courses Module Instructor can add course, remove course and view course timings. If the instructor add course, he will get a popup as course added successfully and vice-versa.

## Module2: Student Use Case Diagram

This actor plays a key role in the system. He is responsible for login in to the system using his Student ID as his user name and using his unique password with all the constraints. He then scans the QR and captures an image of the QR and then sends it to the instructor for grading his attendance. He can also view the overall courses he is registered in to and also check the individual percentage in each of his course. He receives a feedback as soon as the code is captured and is sent to the instructor for grades.

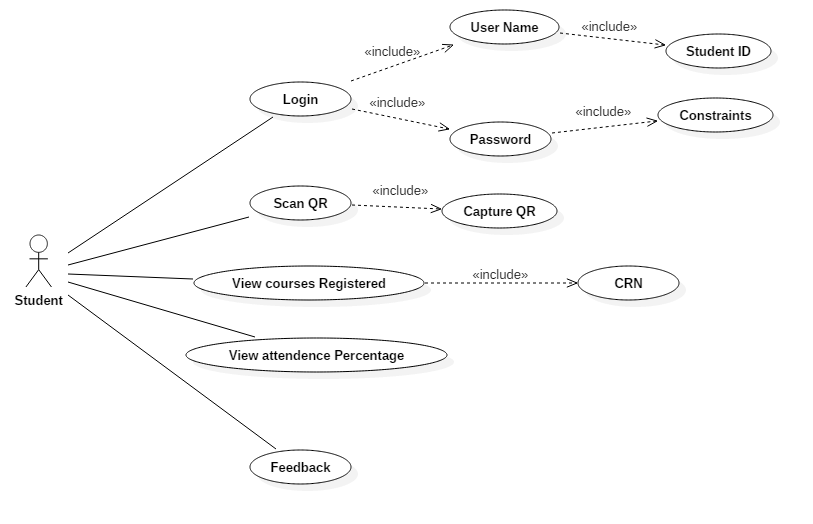


Figure7: Student Use case Diagram

1. Student: Login Successful

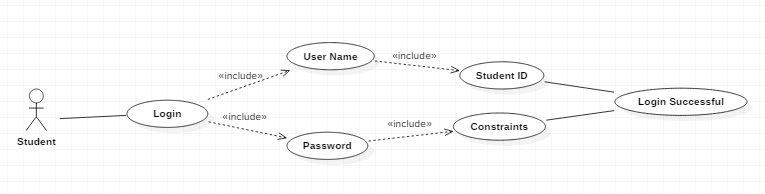


Figure8: Student Login Successful Use case Description

In the initial stage the student must login with his User Name and Password. Username of the student is his SID and password has few constraints. Once logged in the student will get a popup as login successful. In the student view once they logged in they can see three modules like Scan QR, View courses registered and view attendance percentage.

1. Student: Login unsuccessful

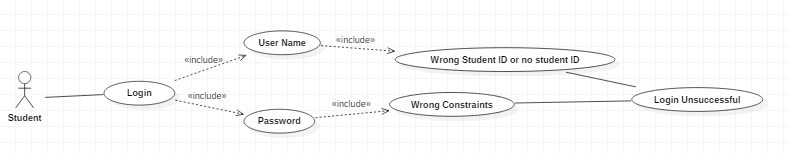


Figure9: Student Login Unsuccessful Use case Description

In the initial stage if the student enters wrong username or wrong password or if they have not entered any username or the password they will get a popup as login unsuccessful.

1. Student: Scanning QR

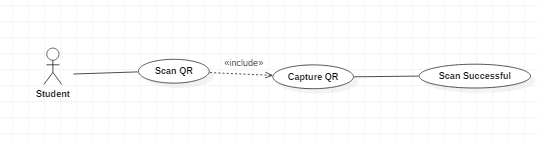


Figure10: Student Scan QR Use case Description

In the scan QR module, student can scan QR code with his mobile application and later he must capture the QR code for security purpose. After capturing QR the student gets a popup as scan successful.

1. Student: View Course Details

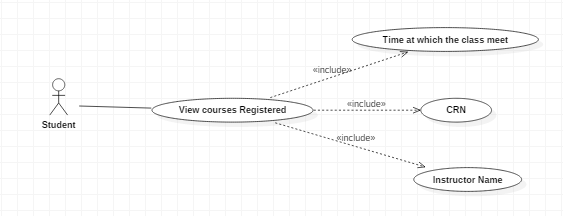


Figure11: Student View Course Details Use case Description

In the view courses registered module students can view their courses which they have been registered for that semester. Within that module they can view timings of the class, CRN and instructor name of that course.

1. Student: Attendance Percentage and feedback

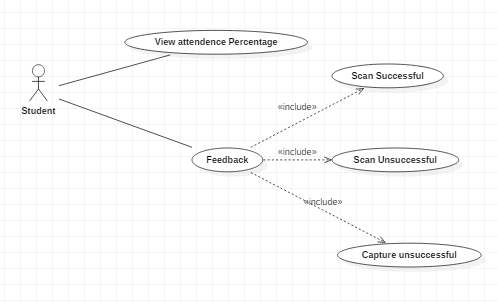


Figure12: Student Attendance Percentage and Feedback Use case Description

In the view attendance percentage module students can view their attendance percentage and students will receive feedback after scanning and capturing the QR like scan successful, scan unsuccessful and capture unsuccessful.