

**IT 314**  
**SOFTWARE ENGINEERING**  
**GROUP 6**  
**AUTO ATTENDANCE SYSTEM**

# INDEX

## 1. Problem Statement

1.1 Purpose

1.2 Intended Audience

1.3 Product Scope

## 2. Overall Description

2.1 Need for this system

2.2 Features of this system

## 3. Requirements

3.1 Functional Requirements

3.2 Non-Functional Requirements

## 4. Use case diagram

## 5. Use case description

# **1.INTRODUCTION**

## **PROBLEM STATEMENT**

Auto attendance system refers to a system that automates the process of taking attendance in a classroom or workplace, APP for automated attendance in Lab/Lectures. This project is a web application that provides the students an option to automatically mark attendance in the lecture and lab sessions. The instructor opens the sessions for a specific duration and students have to mark attendance for the same. AutoAttendanceSystem reduces the time of marking attendance. This system automates the process and reduces the time and effort required by an instructor.

### **1.1 PURPOSE**

The purpose of developing attendance management systems is to computerize the traditional way of taking attendance. Another purpose for developing this software is to generate the report automatically. This document contains detailed functional and non-functional requirements for the attendance management system. The purpose of this document is that the requirements mentioned in it should be utilized by software developers to implement the system and the end user should be able to utilize all the functionalities and features of the system.

This Software Requirements Specification document only covers the main system and does not describe the implementation of the database in which the main system interacts nor it describes any kind of backend programming. This documentation directly focuses on the features provided to the end-user.

### **1.2 INTENDED AUDIENCE**

The audience of this system will be:

1. Students
2. Instructor

### **1.3 PRODUCT SCOPE**

The instructor in charge will be able to keep track of the attendance of students in their various courses from a device by using the Auto Attendance System. Additionally, the system would allow the in-charge instructor to keep track of all the attendance information about all the students.

The system also offers many options like creating attendance, viewing statistics, and downloading reports of attendance. The objective is to offer an instructor a simple, portable solution for maintaining attendance records and attendance statistics.

This system allows the instructor to maintain attendance records of the classes to which he/she is teaching. The system provides a facility to download the report as per the requirement of the instructor.

## **2. OVERALL DESCRIPTION**

### **2.1 NEED FOR THIS SYSTEM**

The need for this system is as follows:

- 1) To save user time, cost, and institute resources.
- 2) The system is helpful as it generates a systematic overall report of every class attendance.
- 3) It maintains the record in a large database instead of the conventional method of maintaining a register or a biometric attendance, which further simplifies the process of searching for a particular record.
- 4) It provides accuracy.
- 5) This web application records data automatically.
- 6) To provide a good analysis for students as well as for professors to view the attendance data.
- 7) Helps the student to be punctual as there is a specific duration to mark the attendance.
- 8) Helps to lessen the paper-work and human efforts.

### **2.2 FEATURES OF THIS SYSTEM**

The features of this system are as follows:

- 1) Provides a separate login page for students and Instructors.
- 2) Instructors can create a new course and instructors can submit the student information in the portal to enroll that student.
- 3) During a lecture, Instructors can open an attendance for a specific period of time and students can mark their attendance in the same.
- 4) Instructors can enroll the students by uploading excel file consisting of two columns one "name" and other "email".
- 5) Students will get an email which consists of the name of the course he/she is newly enrolled in.
- 6) Students will be able to view which lecture he/she has marked attendance and in which he/she has not marked.
- 7) Instructors will be able to view the dashboard of the attendance of students of each lecture and in each course.
- 8) Instructor will be able to download the report of attendance at the end of the course.

- 9) Students will be notified in the form of mail for his/her attendance in a course along with its time frame for marking the attendance.
- 10) Instructor will also be able to delete the course.

## **3. REQUIREMENTS**

### **3.1 FUNCTIONAL REQUIREMENTS**

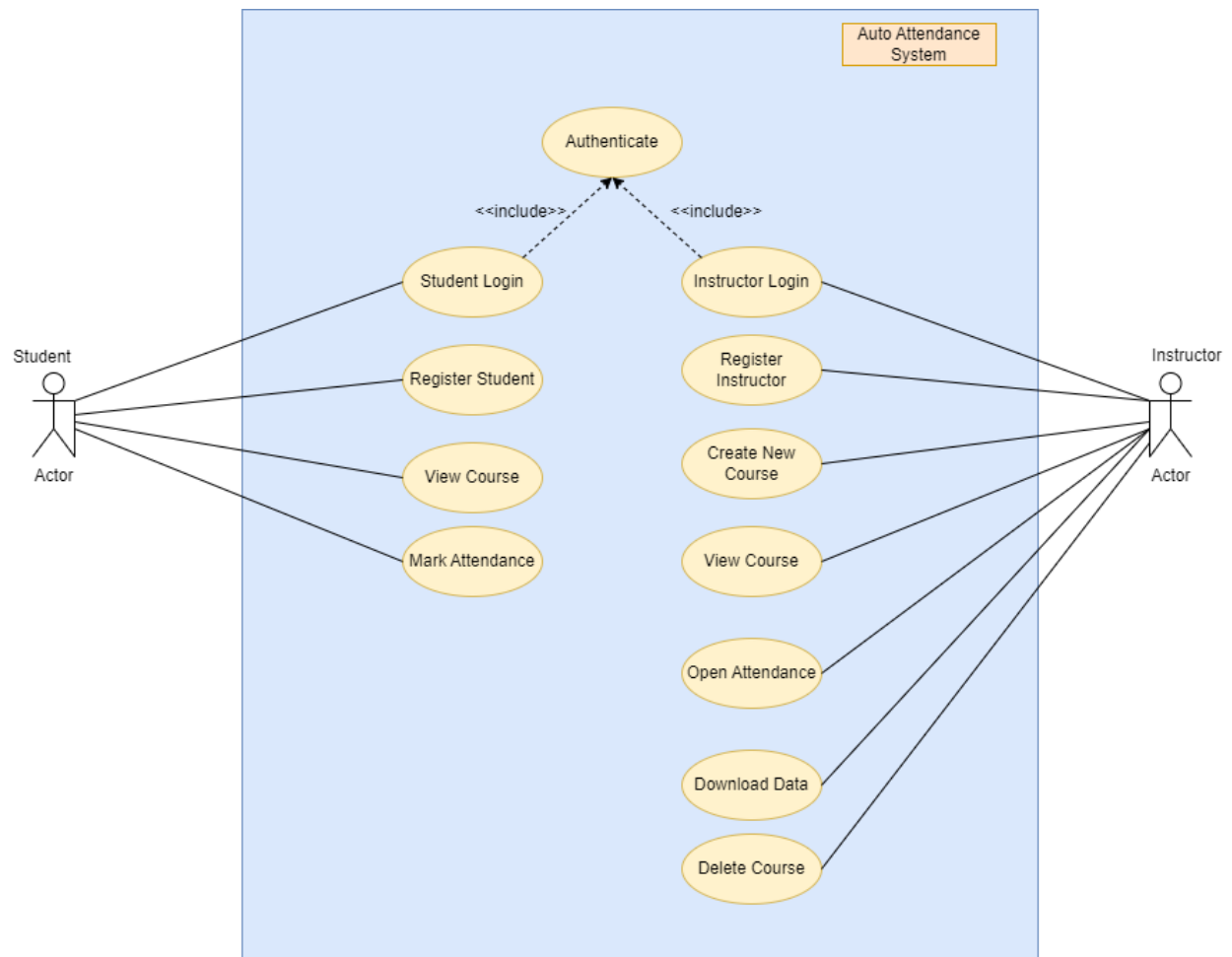
The functional requirements are as follows:

- 1) To provide a separate login feature for students and instructors.
- 2) Instructors will be able to create a new course and submit the student information in the portal to enroll that student in the form of excel file.
- 3) Students will get enrolment confirmation mail for that course.
- 4) Students will be able to see attendance graphs for each of his/her enrolled courses.
- 5) Instructor will be able to specify the time frame for attendance.
- 6) Students will get mail which consists of a link for marking his/her attendance.
- 7) Instructors will be able to make multiple courses and can see the attendance graph for each course.
- 8) Instructors will be able to see in which lecture which student had marked the attendance and which students have not marked attendance.

### **3.2 NON-FUNCTIONAL REQUIREMENTS:**

- 1) Security: The website should store the data securely and provide protection against threats.
- 2) Compatibility and Portability: The website should adapt according to the specifications of the user system. The website should support all the modern web browsers as well as older versions.
- 3) Reliability and Availability: The website should remain up 24\*7 and the databases and servers should be ready and available to establish the connection all the time.
- 4) Scalability and Performance: Multiple users should be able to access the website at a time and the process of authentication and validation should be fast even for a high number of users at the same time.
- 5) Localization: The website should fit into the local specifics of the user system. It should be responsive to the screen size.
- 6) Usability: The UI should be easy to use for all kinds of users.(i.e. Technical as well as Non-Technical users).

## 4. USE CASE DIAGRAM



## 5. USE CASE DESCRIPTION

### 1. Use Case: Student Registration

**Use Case Id:** UC\_1

**Goal:** This describes how a student registers the Attendance system.

**Actors:** Student

**Pre-condition:** User must have a valid email-address.

**Description:**

1. AAS asks for valid credentials like first name, last name, password.
2. User enters the credentials.
3. AAS creates a new student account.

**Post condition:** AAS displays login page.

**Exception:**

- 2.a If a student enters an invalid email address.
  - 2.a.1 System will again redirect to the registration page.
- 2.b if the password is not in the valid format.
  - 2.b.1 System will again redirect to the registration page.

### 2. Use Case: Student Login

**Use Case Id:** UC\_2

**Goal:** This describes how a student logs into the Attendance system.

**Actors:** Student

**Pre-condition:** User must create a valid account using UC\_1

**Description:**

1. AAS asks for valid credentials.
2. User enters the credentials.



3. AAS authenticates and allows the user to log in the system.

**Post condition:** AAS displays relevant dashboard.

**Exception:**

3.a. Authentication fails.

3.a.1 AAS displays the error message.

3.a.2. AAS again returns back to the login page.

### **3. Use Case:** View course

**Use Case Id:** UC\_3

**Goal:** This shows how to view the course.

**Actors:** Student

**Pre-condition:** Students should be successfully logged in the system by performing UC\_2.

**Description:**

- 1.AAS shows the list of all the courses that the student has enrolled.
- 2.Students click on the particular course that he/she wants to inspect.
- 3.AAS shows the valid course page which consists of attendance statistics.

**Post condition:** AAS displays course page.

**Exception:** None.

### **4. Use Case:** Mark Attendance

**Use Case ID:** UC\_3

**Goal:** To mark 'Present' for the student in particular course.

**Actors:** Student

**Description:**

1. The Student should get mail about the attendance in that particular course.

2. The Student clicks on the mark attendance button.

**Alternative:**

1. The Student logged in into the system using UC\_2.
2. The Student clicks on active attendance button.
3. The Student can mark the attendance of whichever course he/she wants to mark.

**Post-condition:**

Students will get a confirmation status about attendance of that course-lecture.

**Exception:**

- 2.a. The attendance portal expired.
  - 2.a.1 AAS shows an error message.

## **5. Use Case:** InstructorRegistration

**Use Case Id:** UC\_5

**Goal:** This describes how an instructor registers in the Attendance system.

**Actors:** Instructor

**Pre-condition:** Instructor must have a valid email-address.

**Description:**

1. AAS asks for valid credentials like first name, last name, password.
2. User enters the credentials.
3. AAS creates a new instructor account.

**Post condition:** AAS displays login page.

**Exception:**

- 2.a If the instructor enters an invalid email address.
  - 2.a.1 System will again redirect to the registration page.
- 2.b If the password entered is not in a correct format.
  - 2.b.1 Instructor is again redirected to the register page.

## **6. Use Case:** Instructor Login

**Use Case Id:** UC\_6

**Goal:** This describes how an instructor logs into the attendance system.

**Actors:** Instructor

**Pre-condition:** Instructor must create a valid account using UC\_5.

**Description:**

1. AAS asks for valid credentials.
2. Instructor enters the credentials.
3. AAS authenticates and allows the user to log in.

**Post condition:** AAS displays relevant dashboard.

**Exception:**

- 3.a. Authentication fails.
  - 3.a.1 AAS displays the error message.
  - 3.a.2. AAS again returns back to the login page.

## **7. Use Case:** Create new course

**Use Case Id:** UC\_7

**Goal:** This shows how to create a new course.

**Actors:** Instructor

**Pre-condition:** Instructor should be successfully logged in the system by performing UC\_6.

**Description:**

1. Instructor clicks on the “Create a new course” button.
2. AAS displays create a new course page.
3. Instructor enters all the course information like course code and course name that the AAS asks.

4. Instructor clicks on the submit button.
5. AAS shows the success message.

**Post condition:** AAS displays the valid course page that the instructor has newly created along with other courses created previously.

**Exception:**None.

## **8. Use Case:** View course

**Use Case Id:** UC\_7

**Goal:** This shows how to view the course.

**Actors:** Instructor

**Pre-condition:** Instructor should be successfully logged in the Auto-Attendance System by performing UC\_6.

### **Description:**

1. AAS shows the list of all the courses that he/she has created using UC\_7.
2. Instructor clicks on the particular course that he/she wants to inspect.
3. AAS shows the valid course page which consists of attendance statistics.

**Post condition:** AAS displays the valid course page.

**Exception:** None.

## **9. Use Case:** Open attendance

**Use Case Id:** UC\_9

**Goal:** This shows how to open a new attendance portal.

**Actors:** Instructor

**Pre-condition:** Instructor should be successfully logged in the system by performing UC\_6 and instructor should be able to view that course that he/she wants to open attendance using UC\_8.

### **Description:**

1. Instructor opens the course page.

2. Instructor clicks on the “Open new attendance” button.
3. AAS displays a form.
4. Instructor enters the lecture name and time frame.
5. AAS validates and shows the confirmation page.

**Exception:**

- 5.a. AAS validation fails.
  - 5.a.1 AAS shows the error message.
  - 5.a.2 AAS redirects the user to the same form page.

**Post condition:** AAS returns back to the same course page with the confirmation message.

**10. Use Case:** Download data

**Use Case Id:** UC\_10

**Goal:** This shows how to download attendance data.

**Actors:** Instructor

**Pre-condition:** Instructor should be successfully logged in the system by performing UC\_6 and instructor should be able to view that course using UC\_8.

**Description:**

1. Instructor opens the course page.
2. Instructor clicks on the “Download data” button.
3. AAS starts downloading the attendance data for that course.

**Post condition:** AAS returns back to the same course page along with download data.

**11. Use case:** Authentication

**Use case:** UC\_11

**Goal:** To authenticate user’s credentials.

**Actors:** AAS

**Pre-condition:** None

**Description:** None

**Post condition:** AAS shows success or error message.

## **12. Use Case:** Delete course

**Use Case Id:** UC\_12

**Goal:** This shows how to delete the course.

**Actors:** Instructor

**Pre-condition:** Instructor should be successfully logged in the system by performing UC\_6 and instructor should be able to view that course using UC\_8.

**Description:**

1. After logged into the system instructor can view the list of course created.
2. Instructor clicks on “delete” button besides the course.
3. AAS will delete the course and redirect to the dashboard.

**Post condition:** AAS will delete that particular course and show the remaining courses.