**COMSATS University Islamabad,**

**Abbottabad Campus**

**SOFTWARE DESIGN DOCUMENT  
(SDD)**

**for**

**<University Attendance System>**  
Version 1.0

***By***

**Fawad Iqbal CIIT/FA21-BSE-012/ATD**

**Faizan CIIT/FA21-BSE-011/ATD**

***Supervisor*Prof. Mukhtiar Zamin**

***Bachelor of Science in Software Engineering (2021-2025)***

**Table of Contents**

[Chapter 1 Project Proposal 6](#_Toc139999988)

[1.1 Introduction 6](#_Toc139999989)

[1.2 Vision 6](#_Toc139999990)

[1.3 Business Case 7](#_Toc139999991)

[1.4 Functional Requirements 7](#_Toc139999992)

[1.5 Supplementary Specification 9](#_Toc139999993)

[1.6 Glossary 11](#_Toc139999994)

[1.7 Risk List & Risk Management Plan 12](#_Toc139999995)

[Chapter 2 Use Cases 13](#_Toc139999996)

[2.1 Use Case Diagram 13](#_Toc139999997)

[2.2 Use Case Distribution 13](#_Toc139999998)

[2.3 Brief Level Use Case 13](#_Toc139999999)

[2.4 Fully Dressed Use Cases 13](#_Toc140000000)

[Chapter 3 UI Prototypes 14](#_Toc140000001)

[3.1 Login UI 14](#_Toc140000002)

[3.2 Dashboard 14](#_Toc140000003)

[3.3 Add Attendance 15](#_Toc140000004)

[Chapter 3 Domain Model 15](#_Toc140000005)

[3.1 Domain Model 15](#_Toc140000006)

[Chapter 4 System Sequence Diagram (SSD) 16](#_Toc140000007)

[4.1 SSD 16](#_Toc140000008)

[Chapter 5 Operation Contract (OC’s) 16](#_Toc140000009)

[5.1 Operation Contract 16](#_Toc140000010)

[Chapter 6 Package Diagram 17](#_Toc140000011)

[6.1 Package Diagram 17](#_Toc140000012)

[Chapter 7 Class Diagram 18](#_Toc140000013)

[7.1 Class Diagram: 18](#_Toc140000014)

[Chapter 8 Interaction Diagram 19](#_Toc140000015)

[8.1 Interaction Diagram 19](#_Toc140000016)

[8.2 Sequence Diagram 19](#_Toc140000017)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **sName** | **Date** | **Reason for changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

**Application Evaluation History**

|  |  |
| --- | --- |
| **Comments (by committee)**  **\*Include the ones given at scope time both in doc and presentation** | **Action Taken** |
|  |  |
|  |  |

**Supervised by**

**<Prof. Mukhtiar Zamin>**

Signature\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Chapter 1 Project Proposal

## 1.1 Introduction

Efficient management of attendance records is essential in the dynamic landscape of education. Traditional attendance tracking methods often involve laborious manual record-keeping and administrative efforts, leading to inefficiencies and potential errors. To address these challenges, this thesis presents a groundbreaking desktop application for a university attendance system. The application empowers teachers by providing them with a user-friendly interface to mark, modify, and delete attendance records, revolutionizing the way attendance is managed.

The core objective of this desktop application is to streamline the attendance management process within the university environment. By leveraging modern technology, the application simplifies and automates tasks that were once time-consuming and prone to mistakes. A key feature of the application is its fully functional login system, which ensures secure access, validates user credentials, and authenticates authorized users. This aspect guarantees the integrity and confidentiality of attendance data, providing peace of mind to both teachers and administrators.

With this application, teachers gain a powerful tool to mark attendance accurately and efficiently. They can easily modify attendance records to accommodate makeup classes, correct mistakes, or adapt to changing circumstances. The ability to delete attendance entries in certain situations adds flexibility and ensures accurate reporting. By digitizing the attendance tracking process, the application reduces the burden of manual paperwork and data entry, freeing up valuable time for teachers to focus on their core responsibilities of instruction and mentorship.

## 1.2 Vision

The vision for the desktop application for the university attendance system is to transform the way attendance management is conducted within educational institutions. By leveraging modern technology and intuitive design, the application aims to provide an efficient, accurate, and user-friendly solution for teachers and administrators.

The goal is to streamline attendance tracking processes, reduce administrative burdens, and enhance overall productivity within the academic environment.

## 1.3 Business Case

#### The development and implementation of the desktop application for the university attendance system presents several compelling business cases. These cases revolve around improving operational efficiency, enhancing data accuracy, and facilitating informed decision-making. The key business cases include:

#### Time and Cost Savings: The application reduces manual paperwork, streamlines attendance tracking, and automates administrative tasks. This leads to significant time and cost savings for educational institutions by minimizing the effort required for data entry, record-keeping, and report generation.

#### Improved Accuracy and Compliance: The application ensures accurate and reliable attendance data by eliminating the possibility of human errors inherent in manual processes. It also helps institutions meet regulatory requirements and maintain compliance by providing a systematic and auditable attendance management system.

#### Improved Student Experience: The application contributes to a positive student experience by ensuring accurate attendance records, minimizing disputes, and enabling prompt interventions when necessary. Students benefit from a fair and transparent attendance tracking system that supports their academic journey.

## 1.4 Functional Requirements

### Admin Requirements

* **Course and Class Management:** The administrator can manage courses and classes within the system. This involves creating and updating course information, assigning teachers to specific courses or classes, and managing class schedules.
* **Attendance Reporting:** The administrator has access to comprehensive attendance reports for analysis and decision-making. They can generate attendance reports for individual students, classes, or courses over specific time periods. The reports may include details such as attendance percentages, late arrivals, and absences.
* **Data Analytics:** The administrator can leverage data analytics functionalities to gain insights into attendance trends and patterns. They can view graphical representations and statistical summaries of attendance data, enabling them to identify areas of concern, track improvement initiatives, and make data-driven decisions.

### Teacher Requirements

* **Login and Authentication:** The teacher can securely log into the application using their credentials and undergo authentication to access the attendance system.
* **Course and Class Selection:** The teacher can select the relevant course and class for which they want to mark attendance. They can view their assigned courses, class schedules, and associated students.
* **Mark Attendance:** The teacher can mark attendance for students in their assigned classes. They can indicate whether students are present, absent, or late. The system should allow for batch marking of attendance if necessary.
* **Modify Attendance:** The teacher can modify attendance records if changes or corrections are required. They can update attendance statuses, adjust for makeup classes, or rectify any inaccuracies in the attendance data.
* **View Attendance Summary:** The teacher can view attendance summaries for individual students or entire classes. This includes the total number of classes attended, absences, late arrivals, and overall attendance percentages.

### Student Requirements

* **Login and Authentication:** The student can securely log into the application using their credentials and undergo authentication to access the attendance system.
* **View Attendance Records:** The student can view their own attendance records for the courses they are enrolled in. They can see the number of classes attended, absences, and late arrivals, providing them with a clear overview of their attendance performance.
* **Receive Notifications:** The student can receive notifications or alerts related to attendance, such as reminders about upcoming classes, changes in schedules, or important attendance-related announcements.
* **Access Attendance Reports:** The student can access attendance reports to view their attendance history for specific courses or classes. These reports may include detailed information about individual attendance records and percentages.

### System Requirements

* **User Authentication and Access Control:** 
  + The system should provide secure login functionality for admins, students, and teachers.
  + Only registered and authorized users should be able to access the attendance system.
  + User roles and permissions should be defined to control access to distinctive features and data within the system.
* **Attendance Marking and Modification:**
  + The system should allow teachers to mark attendance for students, specifying whether they are present, absent, or late.
  + Teachers should have the ability to modify attendance records when necessary, such as for makeup classes or corrections of errors.
* **Data Management and Storage:**
  + The system should securely store and manage attendance records for each student and class.
  + Attendance data should be associated with specific courses, classes, dates, and students.
  + The system should provide efficient methods for storing and retrieving attendance information.

## 1.5 Supplementary Specification

The Attendance System is an application designed to manage and track student attendance within a university environment. This Supplementary Specification outlines additional features and requirements for the system to enhance its functionality and usability.

### 1.5.1 User Interface:

The user interface should be intuitive and user-friendly, allowing teachers to easily mark attendance, modify records, and generate reports. The interface should provide clear instructions and guidance for efficient attendance management.

### 1.5.2 Performance:

The system should be capable of handling a large number of students, classes, and attendance records. It should be able to process attendance data quickly and generate reports in a timely manner. The system should maintain optimal performance even during peak usage periods.

### 1.5.3 Security:

The system should implement robust security measures to protect student attendance data. User authentication and authorization mechanisms should be in place to ensure that only authorized personnel can access and modify attendance records. All data transmissions should be encrypted to maintain data integrity and confidentiality.

### 1.5.4 Reports:

The system should generate comprehensive attendance reports for teachers, administrators, and students. The reports should include attendance percentages and absences for individual students, classes, and courses. The reports should be customizable and exportable in various formats for easy sharing and analysis.

### 1.5.5 Integration:

The attendance system should integrate with other university systems, such as the student information system and course management system. The integration should facilitate the exchange of relevant data, ensuring accuracy and consistency across different platforms.

### 1.5.6 Accessibility:

The system should adhere to accessibility standards and guidelines to ensure that it is accessible to all users, including those with disabilities. The user interface should be designed with considerations for different accessibility requirements, such as screen readers and keyboard navigation.

### 1.5.7 Support and Maintenance:

The system should be accompanied by comprehensive support and maintenance services. This includes user training, technical support, and regular system updates to address any issues or vulnerabilities that may arise.

### 1.5.6 Performance Metrics:

The system should be evaluated based on performance metrics, such as response time, system uptime, and user satisfaction. Regular performance testing and monitoring should be conducted to identify and address any performance bottlenecks or usability concerns.

### 1.5.7 Conclusion:

The Supplementary Specification highlights additional features and requirements for the Attendance System, ensuring its effectiveness and usability in managing student attendance within the university. By incorporating these specifications, the system can meet the specific needs and expectations of the institution, contributing to improved attendance tracking and management processes.

## 1.6 Glossary

1. Attendance System: The software application designed to track and manage student attendance within a university environment.
2. **Admin:** An administrative user with elevated privileges who has access to system configuration, user management, and overall system administration.
3. **Teacher:** A user role representing the faculty member responsible for marking and managing student attendance for their assigned classes.
4. **Student:** An enrolled individual attending classes who is subject to attendance tracking and record-keeping within the system.
5. **Attendance Record:** A specific instance of attendance for a student in a particular class on a given date, indicating whether the student was present or absent.
6. **Absence:** A status indicating that a student was not present for a specific class or session.
7. **Attendance Percentage:** The calculated percentage of classes attended by a student for a specific course or overall attendance.
8. **Makeup Class:** A rescheduled class to compensate for a missed session, typically arranged due to cancellations or special circumstances.
9. **User Authentication:** The process of verifying the identity and credentials of a user before granting access to the attendance system.
10. **Data Encryption:** The technique used to convert sensitive attendance data into a secure and unreadable format to protect it from unauthorized access.
11. **Attendance Report:** A document or presentation displaying attendance data and statistics, such as attendance percentages, absences, and late arrivals for students, classes, or courses.

## 1.7 Risk List & Risk Management Plan

### 1.7.1 Risk List

1. Data loss due to system failure or user error.
2. Inadequate system security leads to unauthorized access or data breaches.
3. Incorrect data input by users.
4. Insufficient capacity to handle large volumes of data.
5. Software bugs or errors leading to system crashes or malfunctions.
6. Inability to handle unexpected changes in scheduling or events.
7. Resistance from users to adopt the new system or change their habits.
8. Lack of resources (time, budget, personnel) to implement or maintain the system.
9. Lack of integration with other systems, such as student information systems or other systems.
10. Incomplete or inaccurate attendance data due to non-compliance or bypassing the system.
11. Insufficient user training and support resulting in user confusion or errors.

### 1.7.2 Risk Management Plan

1. Data loss: Implement regular backups and disaster recovery procedures.
2. System security: Use strong access controls, encryption, and regular security audits.
3. Incorrect data input: Implement data validation checks and provide user training.
4. Insufficient capacity: Architect the system to be scalable and conduct capacity planning.
5. Software bugs/errors: Conduct thorough testing and implement effective bug tracking.
6. Handling unexpected changes: Design the system to be flexible and have a change management process.
7. User resistance: Conduct extensive user training, involve users in design, and provide ongoing support.
8. Lack of resources: Thorough project planning and resource allocation.
9. Lack of integration: Establish collaboration, define protocols, and conduct thorough testing.
10. Incomplete/ inaccurate attendance data: Establish clear policies, implement auditing mechanisms, and provide training.
11. Insufficient user training/support: Develop comprehensive training materials, conduct sessions, and offer support channels.

# Chapter 2 Use Cases

## 2.1 Use Case Diagram

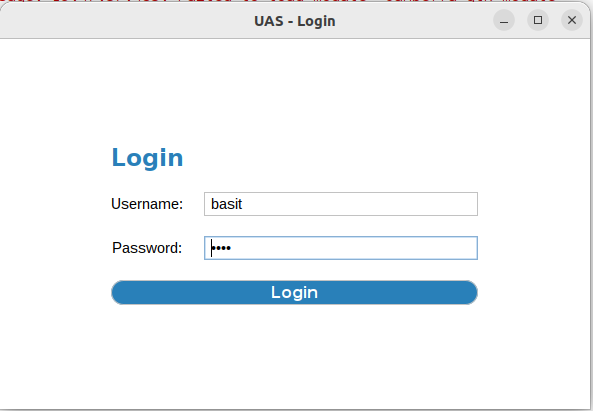
## 2.2 Use Case Distribution

## 2.3 Brief Level Use Case

## 2.4 Fully Dressed Use Cases

# Chapter 3 UI Prototypes

## 3.1 Login UI



## 3.2 Dashboard

## 3.3 Add Attendance

# Chapter 3 Domain Model

## 3.1 Domain Model

# Chapter 4 System Sequence Diagram (SSD)

## 4.1 SSD

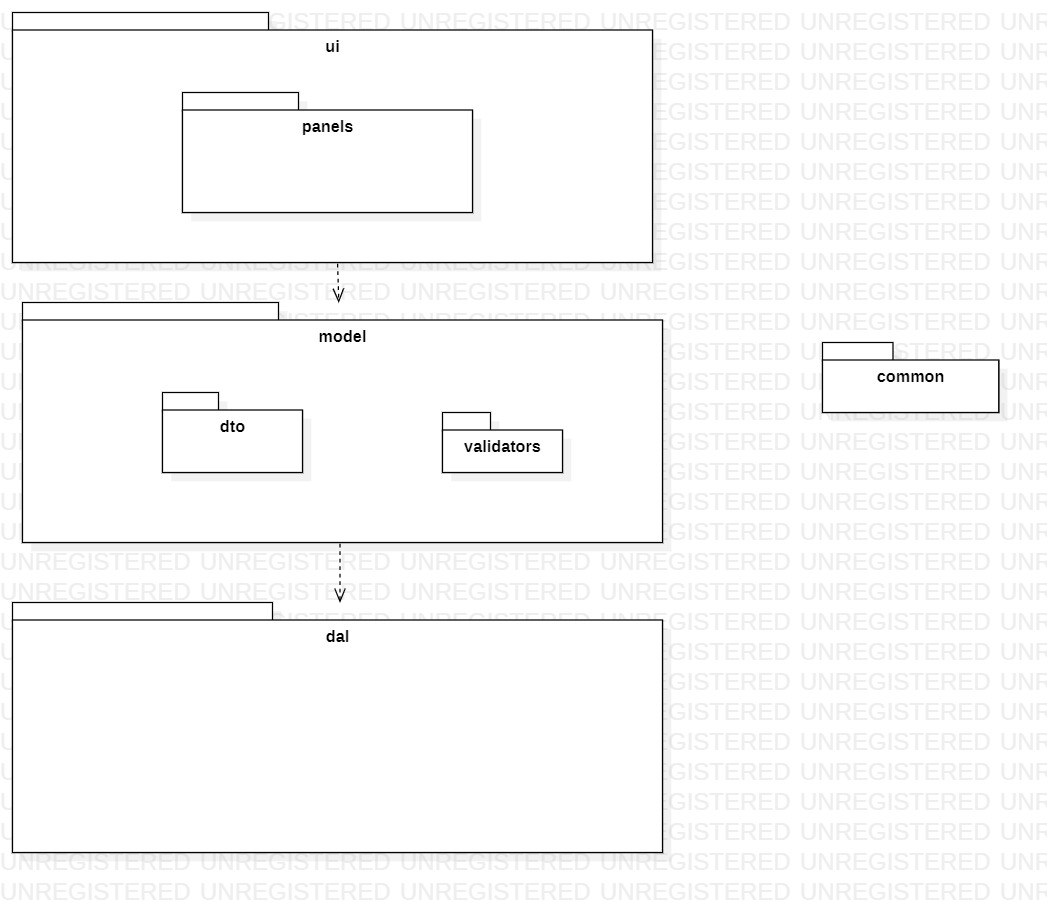


# Chapter 5 Operation Contract (OC’s)

## 5.1 Operation Contract

# Chapter 6 Package Diagram

## 6.1 Package Diagram



# Chapter 7 Class Diagram

## 7.1 Class Diagram:

# 

# Chapter 8 Interaction Diagram

## 8.1 Interaction Diagram



## 8.2 Sequence Diagram