SOFTWARE REQUIREMENTS SPECIFICATION

**For**

**Student Attendance System**

**Prepared by:-**

*Gayathiri K*

*Mangaiyarkarasi R*

*Sreemathi S*

*Vedha Kavya Sri P*

# Introduction

## Purpose

## In the educational landscape, the Student Attendance System serves as a pivotal tool designed to simplify and streamline the process of tracking student attendance. Its primary purpose is to replace traditional manual methods with an efficient and automated system, ensuring accuracy in attendance records. By leveraging technology, the system aims to promote transparency, enhance communication, and contribute to the overall effectiveness of educational institutions in monitoring and managing student participation.

## Document Conventions

* + - Entire document should be justified.
    - Convention for Main title

Font face: Times New Roman Font style: Bold

Font Size: 14

* + - Convention for Sub title

Font face: Times New Roman Font style: Bold

Font Size: 12

* + - Convention for body

Font face: Times New Roman Font Size: 12

## Scope of Development Project

Student Attendance System is extensive, encompassing various facets to ensure a precise and efficient tracking mechanism. This comprehensive project aims to transition from manual attendance processes to an internet-based application, enhancing accessibility and usability for teachers, administrators, students, and parents.

Stakeholders, including educators and guardians, will benefit from real-time reporting features, notifications for irregular attendance, and a user-friendly mobile interface for on-the-go accessibility. The system's architecture will be designed to seamlessly integrate with existing Student Information Systems, maintaining data consistency. Methods for attendance tracking, such as biometrics, card readers, and QR codes, will be implemented based on the institution's needs.

The project will prioritize security compliance to safeguard sensitive information and adhere to data protection regulations. Scalability will be a key consideration, allowing the system to accommodate future growth in student numbers. Post-implementation support and regular maintenance will ensure the longevity and effectiveness of the system. Through thorough testing, documentation, and user training, this project aims to deliver a robust and adaptable student attendance management solution tailored to the dynamic needs of educational institutions.

## Definitions, Acronyms and Abbreviations

JAVA -> platform independence SQL-> Structured query Language ER-> Entity Relationship

UML -> Unified Modeling Language

IDE-> Integrated Development Environment SRS-> Software Requirement Specification

## References

* + - Books
    - "Software Engineering: A Practitioner's Approach" by Roger S. Pressman
    - Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan
    - Java: The Complete Reference" by Herbert Schildt
* Website
* [https://www.iitms.co.in/blog/online-attendance-management-software-for- schools.html#:~:text=Attendance%20management%20system%20records%20and,with%20high%20accuracy%20 and%20efficiency](https://www.iitms.co.in/blog/online-attendance-management-software-for-%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20schools.html#:~:text=Attendance%20management%20system%20records%20and,with%20high%20accuracy%20 and%20efficiency)
* <https://www.linkedin.com/pulse/what-student-attendance-management-system-its-features-nitish-garg>

1. **Overall Descriptions**

## Product Perspective

Use Case Diagram of Student Attendance System

*searches*

1

1 *requests*

1

1

1..\*

\*

search\_book



1..\*

check\_limit

check\_availability

User 1

issue\_book

*request\_renew*

<<include>>

*monitors\_request*

1

*monitors\_renew* 1

1

*performs*

*give\_book*

<<include>>

0..\*

1..\*

renew\_book

verify\_member

<<include>>

*take\_book*

1

1 Librarian

Student

0..\*

1..\*

*adds\_new\_book*

*perform\_transaction\_updation*

Staff

\*

\*

return\_book

View\_logs

<<extend>>

add\_book

\*

calculate\_fine

update\_record

This is a broad level diagram of the project showing a basic overview. The users can be either staff or student.. This System will provide a search functionality to facilitate the search of resources. Further the staff personnel can add/update the resources and the resource users from the

system. The users of the system can request to view their attendance for which they would

have to follow certain criteria.

## Product Function

Entity Relationship Diagram of Student Attendance System



The operating environment of a student attendance management system typically includes hardware like computers or servers, software for data processing, a database system for storage, and network infrastructure for communication. Additionally, it may involve user interfaces for administrators, teachers, and students, ensuring compatibility with various devices and browsers. Security measures are crucial to protect sensitive attendance data, and regular updates may be needed to adapt to evolving technological standards.

## User Classes and Characteristics

The system provides different types of services based on the type of users [Faculty/Student]. The Faculty will be acting as the controller and he will have all the privileges of an administrator. The member is a student of the university who will be accessing the Attendance System.

The features that are available to the Administrator/Faculty are:-

* A Faculty can insert a student to the profile.
* Can view the different department of Students available in the university
* Can view the List of Students available in each department.
* Can take the datas from the student profiles
* Can add no of present and absent and their information to the database
* Edit the information of student attendance
* Can check the report of the day to day attendance
* Can check the report of the monthly attendance
* Can access all the accounts of the students

The features that are available to the Members are:-

* Users who are the primary subjects of the attendance system, needing to check in or out of the classes(students).
* Responsible for recording attendance, managing data, and accessing attendance records(teachers/instructors).
* Oversee the entire system, manage user accounts, generate comprehensive reports, and ensure system integrity(Adminstrators).

## Operating Environment

## Student attendance management systems are often designed to be platform-independent, meaning `they can run on various operating systems. Many of these systems are web-based and can be accessed through popular web browsers like Google Chrome, Mozilla Firefox, or Safari, making them compatible with different operating systems such as Windows, macOS, and Linux. The key is to ensure the system is accessible through a web browser to provide flexibility for users regardless of their operating system.

## Assumptions and Dependencies

The assumptions are:-

* The coding should be error free
* The system should be user-friendly so that it is easy to use for the users
* The information of all users, departments and the details of the student must be stored in a database that is accessible by the website
* The system should have more storage capacity and provide fast access to the database
* The system should provide search facility and support quick responses.
* The Student Attendance System is running 24 hours a day
* Users may access from any computer that has Internet browsing capabilities and an

Internet connection

* Users must have their correct usernames and passwords to enter into their online accounts and do actions

The dependencies are:-

* The specific software due to which the product will be run
* On the basis of listing requirements and specification the project will be developed and run
  + - The end users (admin) should have proper understanding of the product
    - The system should have the general report stored
    - The information of all the users must be stored in a database that is accessible by the Attendance System
    - Any update regarding the Student detail from the profile is to be recorded to the database and the data entered should be correct

## Requirement

Software Configuration:-

This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL Server as the back end to store the database.

Operating System: Windows NT, windows 98, Windows XP Language: Java Runtime Environment, Net beans 7.0.1 (front end) Database: MS SQL Server (back end)

## Data Requirement

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be the queries as fired by the users like create an account, selecting department and putting into account. Now the output will be visible when the user requests the server to get details of their account in the form of time, date and which department they are belonging in the account.

# External Interface Requirement

## GUI

The software provides good graphical interface for the user and the administrator can operate on the system, performing the required task such as create, update, viewing the details of the profile.

* It allows user to view quick reports like how many days the student have taken leave and etc.
* It provides stock verification and search facility based on different criteria.
* The user interface must be customizable by the administrator
* All the modules provided with the software must fit into this graphical user interface and accomplish to the standard defined
* The design should be simple and all the different interfaces should follow a standard

template

* The user interface should be able to interact with the user management module and a part of the interface must be dedicated to the login/logout module

Login Interface:-

In case the user is not yet registered, he can enter the details and register to create his account. Once his account is created he can ‘Login’ which asks the user to type his username and password. If the user entered either his username or password incorrectly then an error message appears.

Search:-

The member or a faculty can enter the student login and can check their activities .

Categories View:-

Categories view shows the department of the student they belongs to.

Librarian’s Control Panel:-

This control panel will allow faculty members or the administrator to add/remove users; add, edit, or remove a resource. And manage lending options.

# System Features

The users of the system should be provided the surety that their account is secure. This is possible by providing:-

* User authentication and validation of members using their unique member ID
* Proper monitoring by the administrator which includes updating account status, showing a popup if the member attempts to login into their respective profile
* Proper accountability which includes not allowing a member to see other member’s account. Only administrator will see and manage all member accounts

# Other Non-functional Requirements

## Performance Requirement

The proposed system that we are going to develop will be used as the Chief performance system within the different campuses of the university which interacts with the university staff and students. Therefore, it is expected that the database would perform functionally all the requirements that are specified by the university.

* The performance of the system should be fast and accurate
* Student Attendance Management System shall handle expected and non-expected errors in ways that prevent loss in information and long downtime period. Thus it should have inbuilt error testing to identify invalid username/password
* The system should be able to handle large amount of data. Thus it should accommodate high number of user information without any fault

## Safety Requirement

The system must adhere to strict data privacy regulations, employing encryption and access controls to safeguard student information. Additionally, it should implement measures to prevent any potential misuse or unauthorized access to sensitive attendance records. Furthermore, the system should undergo regular security audits and updates to address emerging threats and vulnerabilities, ensuring a robust and secure platform for student data management.

**5.3Security Requirement**

* System will use secured database
* Normal users can just read information but they cannot edit or modify anything except their personal and some other information.
* System will have different types of users and every user has access constraints
* Proper user authentication should be provided
* No one should be able to hack users’ password
* There should be separate accounts for admin and members such that no member can access the database and only admin has the rights to update the database.

## Requirement attributes

## Multiple Admin Access: Several administrators can modify the system, while other users have restricted access.

## Open Source Design: The project is open source, allowing for transparency and collaborative improvement.

## User-Friendly Database Quality: The database is designed for ease of use, ensuring all users navigate it comfortably.

## Easy Installation: Users should be able to effortlessly download and install the system.

## 5.5Business Rules

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, make a decision, or infer new data from existing data.This includes the rules and regulations that the System users should abide by. This includes the cost of the project and the discount offers provided. The users should avoid illegal rules and protocols. Neither admin nor member should cross the rules and regulations.

## User Requirement

## The student attendance system must offer distinct access levels for administrators, teachers, and students, ensuring tailored permissions for data entry, viewing records, and system management. Its interface should be intuitive, facilitating effortless attendance tracking across various devices. Real-time updates are critical, allowing immediate recording of attendance statuses like present, absent, or late, ensuring accuracy. Secure authentication measures, such as password protection and strict access controls, safeguard user data. Customizable reporting features enable the generation of tailored attendance reports and in-depth trend analysis. Finally, comprehensive support through guides and training resources ensures seamless system integration and user assistance for optimal utilization.

The admin provides certain facilities to the users in the form of:-

* Backup and Recovery
* Forgot Password
* Data migration i.e. whenever user registers for the first time then the data is stored in the server
* Data replication i.e. if the data is lost in one branch, it is still stored with the server
* Auto Recovery i.e. frequently auto saving the information
* Maintaining files i.e. File Organization

# 6 Other Requirements

## Data and Category Requirement

## There are different categories of users namely teaching staff, Admin, students etc. Depending upon the category of user the access rights are decided. It means if the user is an administrator then he can be able to modify the data, delete, append etc.

Full system access with management privileges.

Teachers/Faculty: Ability to input attendance and manage their assigned classes.

Students: Access to view their own attendance records for various courses.

Attendance Categories: Types of Attendance: Differentiate between regular classes, exams,

extracurricular activities, etc. Attendance Status: Categories like present, absent, late, excused

absences for precise tracking.

## Appendix

A: Admin, Abbreviation, Acronym, Assumptions, Attendance; B:Business rules; C: Class, Client, Conventions; D: Data requirement, Dependencies; G: GUI; K: Key; M: Member; N: Non-functional Requirement; O: Operating environment; P: Performance, Perspective, Purpose; R: Requirement, Requirement attributes, Report; S: Safety, Scope, Security, System features, Student, Staff; U: User, User class and characteristics, User requirement;

## Glossary

The following are the list of conventions and acronyms used in this document and the project as well:

* Administrator: A login id representing a user with user administration privileges to the software
* User: A general login id assigned to most users
* Client: Intended users for the software
* SQL: Structured Query Language; used to retrieve information from a database
* SQL Server: A server used to store data in an organized format
* Layer: Represents a section of the project
* User Interface Layer: The section of the assignment referring to what the user interacts with directly
* Application Logic Layer: The section of the assignment referring to the Web Server. This is where all computations are completed
* Data Storage Layer: The section of the assignment referring to where all data is recorded
* Use Case: A broad level diagram of the project showing a basic overview
* Class diagram: It is a type of static structure diagram that describes the structure of a system by showing the system’s cases, their attributes, and the relationships between the classes
* Interface: Something used to communicate across different mediums
* Unique Key: Used to differentiate entries in a database

## Class Diagram

A class is an abstract, user-defined description of a type of data. It identifies the attributes of the data and the operations that can be performed on instances (i.e. objects) of the data. A class of data has a name, a set of attributes that describes its characteristics, and a set of operations that can be performed on the objects of that class. The classes’ structure and their relationships to each other frozen in time represent the static model. In this project there are certain main classes

which are related to other classes required for their working. There are different kinds of relationships between the classes as shown in the diagram like normal association, aggregation, and generalization. The relationships are depicted using a role name and multiplicities. Here ‘Students’, ‘Parents’ ,‘Staffs’, and ‘Courses’ are the most important classes which are related to other classes.

