**SOFTWARE REQUIREMENT SPECIFICATION**

**FOR**

**STUDENT ATTENDANCE SYSTEM**

**PREPARED BY:**

RAJA SELVIN A (22EC117)

RAGUL G (22EC114)

ROHITH A M (22EC127)

**1.INTRODUCTION**

**1.1 PURPOSE**

The main objective of this document is to illustrate the requirements of the project STUDENT ATTENDANCE SYSTEM. The document gives the detailed description of the both functional and non-functional requirements proposed by the client. The purpose of this project is to provide a friendly environment to maintain the attendance records of students and teachers and the courses. The main purpose of this project is to maintain easy circulation system using computers and to provide different reports. This project describes the hardware and software interface requirements using ER diagrams and UML diagrams.

**1.2 DOCUMENT CONVENTIONS**

* Entire document should be justified
* Convention for Main Title
* Font face: Times New Roman
* Font style: Bold
* Font size: 14
* Convention of Sub Title
* Font face: Times New Roman
* Font style: Bold
* Font size: 12

* Convention for Body
* Font face: Times New Roman
* Font size: 12

**1.3 SCOPE OF DEVELOPMENT PROJECT:**

The student attendance system is used for marking attendance of students and teachers and the the information of the course taken . this particularly helps in generation of the status criteria of a person and can send these information to the person individually.

It is especially useful for any educational institute where modifications in the content can be done easily according to requirements.

The project can be easily implemented under various situations. We can add new features as and when we require, making reusability possible as there is flexibility in all the modules.

The language used for developing the project is Java as it is quite advantageous than other languages in terms of performance, tools available, cross platform compatibility, libraries, cost (freely available), and development process.

**1.4 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

**1.5 REFERENCES**

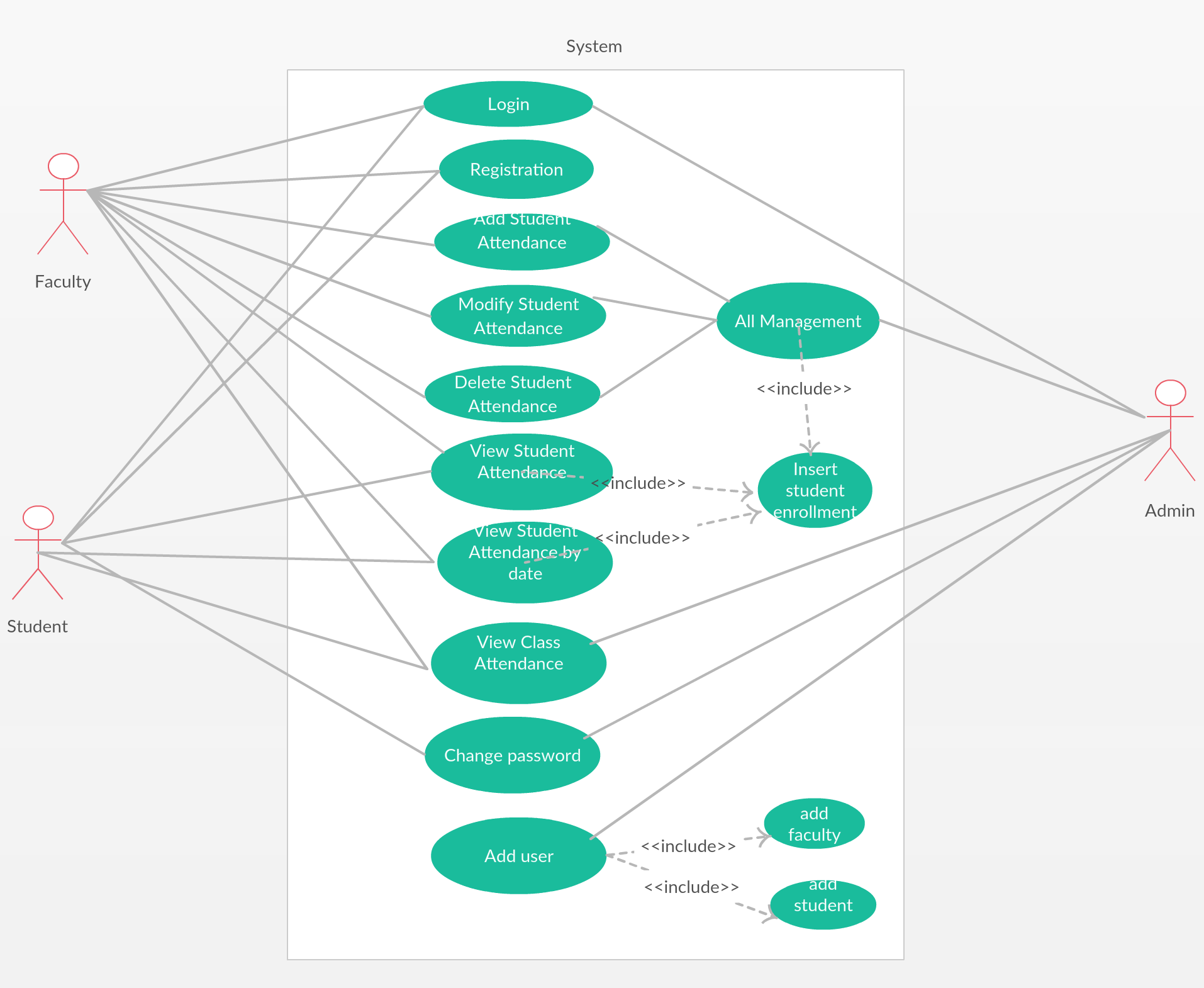
**WEBSITES**

* USVBIBVHBFVHVBHIFVBVIBE
* NJVNDJNFJVDBBDBV

**2. OVERALL DESCRIPTIONS**

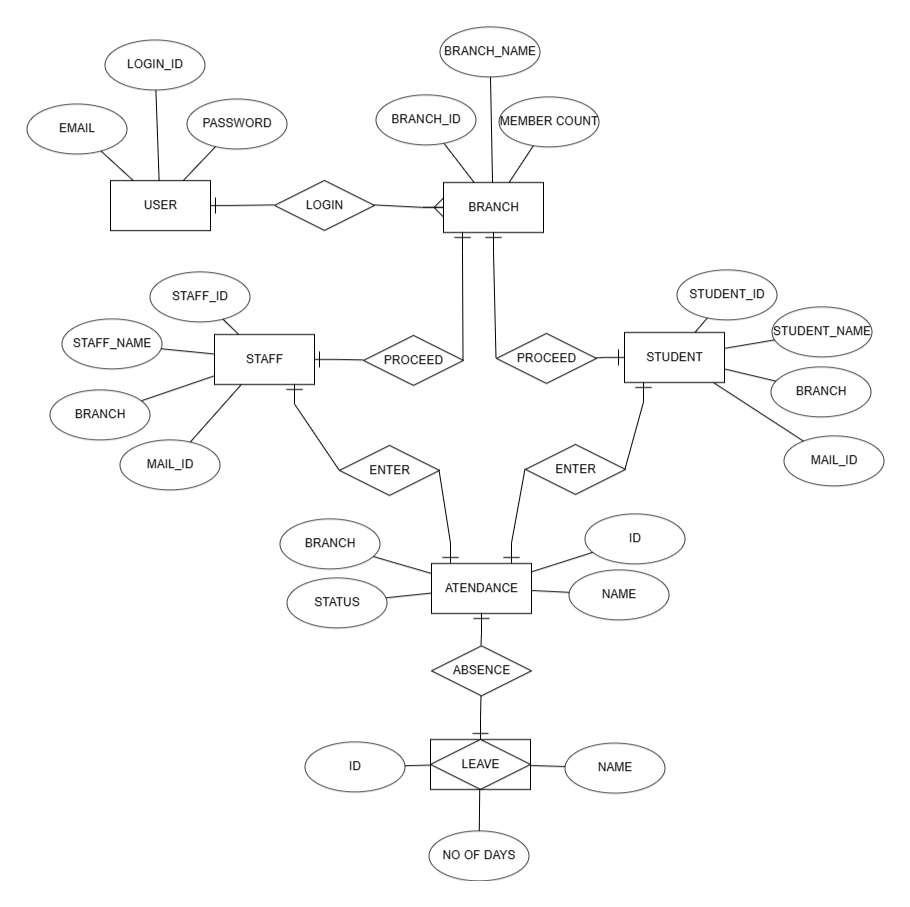
**2.1 PRODUCT PERSPECTIVE**

Use case diagram of student attendance system

****

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

**2.2 PRODUCT FUNCTION**

Entity relationship diagram of student attendance system

The attendance system provides an login criteria to the database in order to obtain the attendance hiistory of the students and staffs of different classes and branch .this criterea goes for the staffs too that it provides an information that is stored in the database inorder to retrive the value for the status providance of the individual attendance summary. The project provides an students and staffs input to vote their atendance and their percentile status to the user who tends to retrive the data.

**2.3 USER CLASSES AND CHARACTERISTICS**

The system provides different types of services based on the type of users [STUDENT/STAFF/ADMIN]. The admin will be acting as the controller and he will have all the privileges of an administrator. The member can be either a student or staff of the university who will be voting their attendance.

The features that are available to the admin are:

* The admin can login into the whole database.
* The admin can change the attendance status of a student or a staff.
* Can view the list of total attendance in each and every branch.
* Can block students and staff from an database.
* Can retrieve the data of the whole database.
* Can access the acts of all students and staff.

The features available to the staffs are:

* Can vote their attendance.
* Can moniter the attendance of the students.
* Can make a change in the attendance status of the students.
* Can retrieve the data of the students attendance.
* Can retrieve their attendance status.

The features available to the students are:

* Can vote their attendace.
* Can veiw their attendace status from the database.

**2.4 OPERATING ENVIROINMENT**

The product will be operating in windows environment. The Student Attendance System is a website and shall operate in all famous browsers, for a model we are taking Microsoft Internet Explorer, Google Chrome and Mozilla Firefox. Also it will be compatible with the IE 6.0. Most of the features will be compatible with the Mozilla Firefox & Opera 7.0 or higher version. The only requirement to use this online product would be the internet connection.

The hardware configuration include Hard Disk: 40 GB, Monitor: 15” Color monitor, Keyboard: 122 keys. The basic input devices required are keyboard, mouse and output devices are monitor, printer etc.

**2.5 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 the users.
* The info regarding the attendance of members should be accessible.
* The system should have more storage capacity and provide fast access to the database.
* User may access the database at anytime from anywhere with internet browsing capabilities.

The dependencies are:

* The specific hardware and software due to which the product will be run.
* On the basis of listing requirements and specifications the project will be developed and run.
* The end user (Admin) should have proper understanding of the project.
* The system should have general report stored.
* Any updates in the attendance of an individua login should be stored simultaneously in the database.

**2.6 REQUIRMENT**

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)

Hardware configuration:

Processor: Pentium(R)-dual core CPU

Hard Disk: 40GB

RAM: 256 MB or more

**2.7 DATA REQUIRMENT**

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 books 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 books are currently in the account.

**3.EXTERNAL INTERFACE REQUIRMENT**

**3.1 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 book.

* It allows user to view quick reports like present and absent in between particular time.
* 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.

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.

View:

View shows the retieved data of the attendance of a single student or staff as per their login id.

**4.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 issue the data of a particular individuals attendance sheet.
* Proper accountability which includes allowing a staff to see other students account. And administrator will see and manage all member accounts

**5.OTHER NON-FUCTIONAL REQUIREMENTS**

**5.1 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 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

**5.2 SAFETY REQUIREMENT**

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup so that the database is not lost. Proper UPS/inverter facility should be there in case of power supply failure.

**5.3 SECURITY REQUIREMENT**

* + - System will use secured database.
    - Normal users can just read information but they cannot edit or modify anything.
    - 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.

**5.4 REQUIREMENT ATTTRIBUTES**

* + - There may be multiple admins creating the project, all of them will have the right to create changes to the system. But the members or other users cannot do changes.
    - The project should be open source.
    - The Quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database.
    - The user be able to easily download and install the system.

**5.5 BUSINESS 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.

**5.6 USER REQUIREMENT**

The users of the system are members and staffs of the university who act as administrator to maintain the system. The members are assumed to have basic knowledge of the computers and internet browsing. The administrators of the system should have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, user manual, online help and the guide to install and maintain the system must be sufficient to educate the users on how to use the system without any problems.

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**

**6.1 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. All other users except the admin and staff only have the rights to retrieve the information about database.

**6.2 APPENDIX**

A: Admin, Abbreviation, Acronym, Assumptions; 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; S: Safety, Scope, Security, System features; U: User, User class and characteristics, User requirement;

**6.3 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

**6.4 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.

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.