Software Requirements Specification

for

University Management System

Version 1.0 approved

Prepared by
Mamta Yadav
Vanshika Bais
Aditya Yadav
Kapil Pawar

Shri Ramdeobaba College of Engineering and Management,Nagpur

08-05-2023

Copyright © 1999 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document.

Table of Contents

Revision History

Name	Date	Reason For Changes	Version	

1. Introduction

University Management System is created with the purpose of providing users an efficient system which can follow the business model of the University and provide competent services, and augment new functionalities to ameliorate the current system.

1.1 Purpose

This document describes the software requirements specification (SRS) for the University Management System that provides the access and management of information of different modules in a like Students,Teachers/Faculty, Fees, Marks, Leave. Our project is based on a database, which stores and maintains the information of different modules within the system. The advantage of the management system is to avoid entries in hard copies and it saves the burden of hard copies of data. The system is a Desktop Application and GUI for this system is developed in java. The Database for this management system is created in SQL. There are three users for this system 1. Admin (have full access to read and write of all modules in management system) 2. Teacher (have access limited to write and manage the student's marks, grant leave, etc.) 3. Student(have access to view marks , apply leave).This is version 1.0 of the software requirements specification.

The purpose of this document is to retrieve and analyze the ideas that define the product and requirements that the user needs. This document describes the details of our product, its parameter, and its goals. This SRS document describes the target, audience, user interface of product and Software/Hardware requirements of our product. This document also describes the problem we have faced during the designing and implementation of the product and also describes how we have solved this problem and make our product more efficient. The management system saves the human power and time cost to perform the same task. The data in the database can be saved for a long time and can be used for different purposes in the future. In management systems, there is a minor chance of losing the data. This document also defines how customers and users see our product and understand the functionality of the product. This document will help the developers/designers in case of maintenance of the software product.

1.2 Document Conventions

This document follows the IEEE standard for software requirements specification. The priority for higher-level requirements is assumed to be inherited by detailed requirements.

1.3 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, marketing staff, users, testers, and documentation writers.

Developers: The SRS is a technical specification document that specifies the software requirements, design restrictions, and interfaces that will be implemented by the development team. The SRS can be used by developers to comprehend the SIMS's functional and non-functional needs and transform them into software code.

Users: The SRS gives a clear and short description of the SIMS features and functionalities, which can assist users in understanding how the system works and how it

can assist them in inventory management. Users can also utilize the SRS to provide feedback on the usability, accessibility, and performance of the system.

Project managers: Project managers can use the SRS to plan and monitor project progress, allocate resources, and manage risks. They can also utilize the SRS to guarantee that the project satisfies the needs and specifications of the client.

Marketing Staff: The SRS contains information about the SIMS's features, benefits, and target users, which can assist marketing staff in understanding the product and developing marketing strategies. They can also utilize the SRS to describe the value proposition of the product to potential buyers.

Testers: The SRS provides the SIMS's acceptance criteria, test cases, and test scenarios, which can assist testers in developing test plans and validating the system's functionality, dependability, and performance. The SRS can also be used to guarantee that the system fulfils quality standards and user expectations.

Technical writers use the SRS as a key source of information to create user manuals, help files, and other documentation items for the SIMS. They can utilize the SRS to check that the documentation is correct, complete, and meets the system requirements.

The SRS is organized as follows:
Section 1 provides an introduction to the document
Section 2 provides an overall description of the product
Section 3 describes the external interface requirements
Section 4 describes the functional requirements,
Section 5 other non-functional requirements.

1.4 Product Scope

As Universities are growing day by day more and more, and also increasing the complexity of storing information of students and related to the University system, they face many related issues: fee of students, details of employees, details of students,leave etc.

This project is based on the educational institute system where this application gives maximum services in a single software product that is used by student,teacher and system administration. This project is based on a desktop application that is sharing information on different departments in a college.

In this project that includes Java and SQL. Java is used to design the GUI for the application by which the user can interact with software applications. The SQL Server is used for creating the database in which different information will store. The main focus of this project is to give the best GUI for the users and provide the many modules in a single product. Admin can view all of the information that is stored in the database through application and admin also can modify this information because the admin has full access to the system.

The teacher can view and modify their information as well as information related to students, teachers have limited access. The student can modify their information only and can see,marks, apply for leave.This project can adjust any additional module at any time.

The University Management System will cover the following functionalities:

- Login/Registration
- Fees
- Marks
- Leave

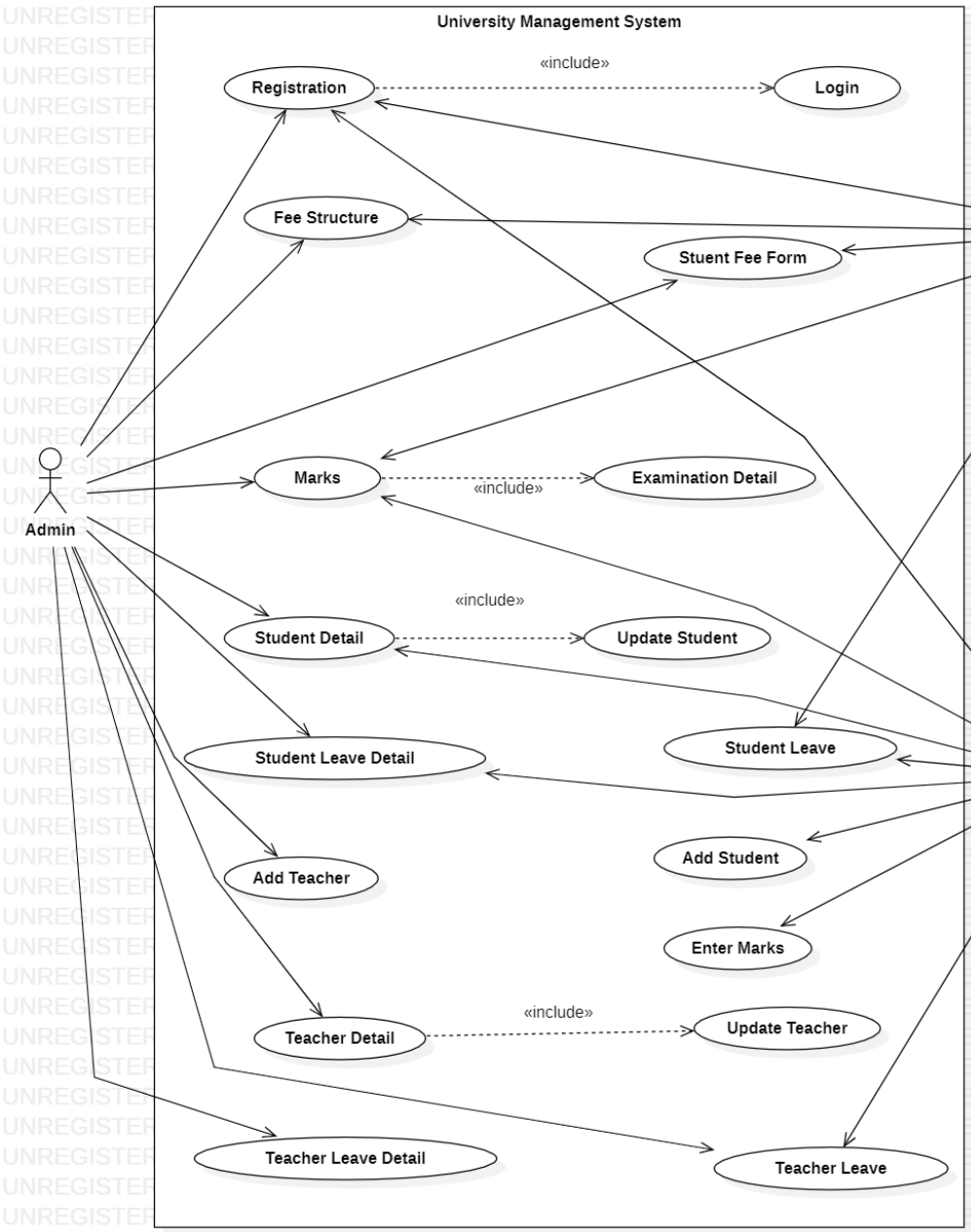
1.5 References

IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.
Atlassian

2. Overall Description

2.1 Product Perspective

A university management system is the software solution designed to streamline and automate various administrative tasks and processes within a university or educational institution. It serves as a centralized platform that integrates and manages multiple functions, departments, and stakeholders within the university ecosystem.



USE CASES:

1. Registration: entry of important details.
2. Fee Structure: Fee structure for particular course.
3. Marks: Marks obtained by students in exam.
4. Student Detail: information about a student.
5. Student Leave Detail: detail of the day student on leave.
6. Add Teacher: adding new teacher.
7. Teacher Detail: information about the teacher.
8. Teacher Leave Detail: detail of the day teacher on leave.
9. Login: login credentials for admin.
10. Student Fee Form: pending dues data of student.
11. Examination Detail: result of semester examinations.
12. Update Student: can change the personal information of student.
13. Student Leave: student can apply for leave.
14. Add Student: adding new student.
15. Enter Marks: can enter the marks of particular subject.

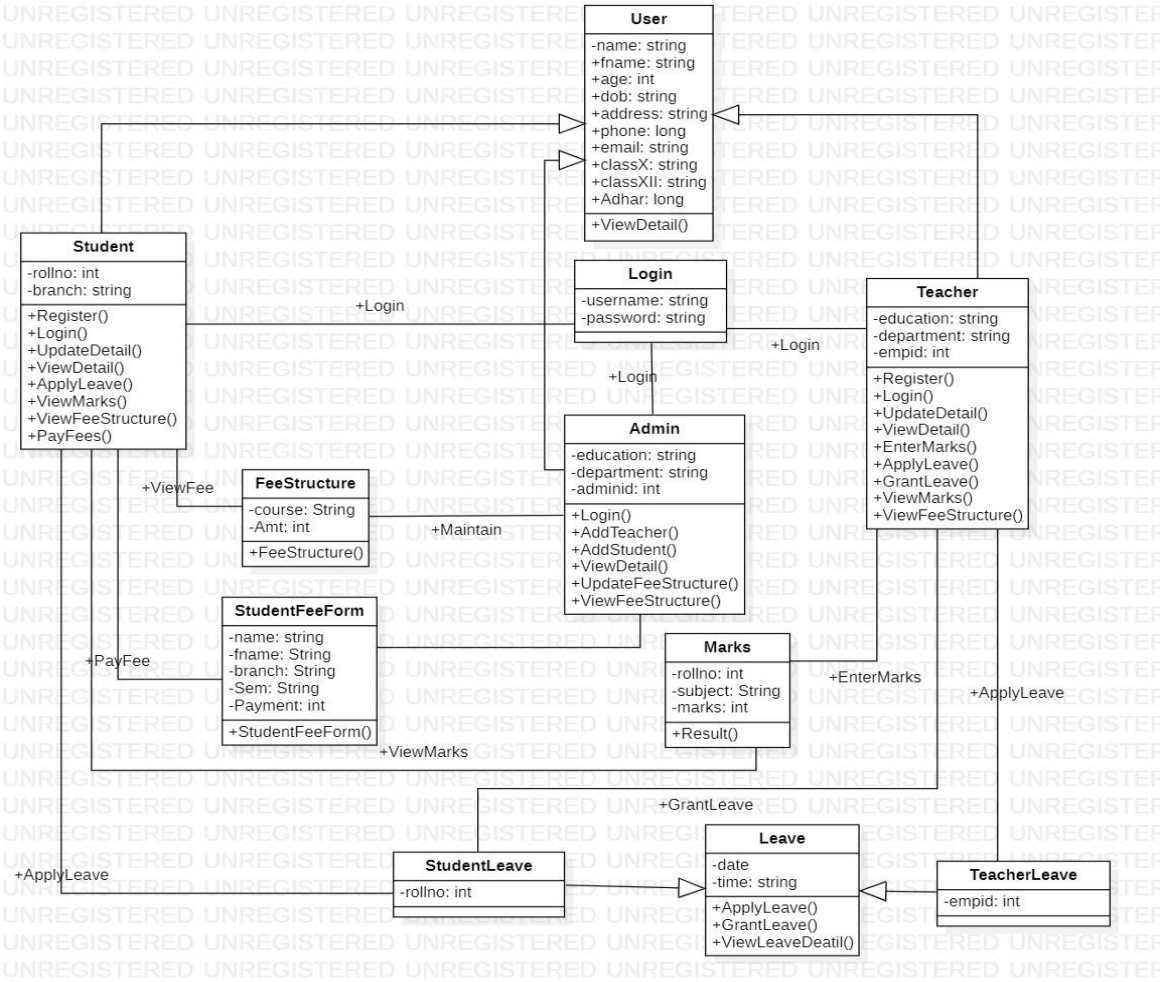
16. Update Teacher: can change the personal information of teacher.
17. Teacher Leave: Teacher can apply for leave.

ACTORS:

1. Admin: the admin of university is responsible for roles of registration, fee structure, marks, student detail, student leave detail, add teacher, teacher detail, teacher leave detail, student fee form, teacher leave.
2. Student: the student is responsible for registration, fee structure, student fee form, marks, student leave.
3. Teacher: the teacher is responsible for registration, marks, student details, student leave, student leave detail, add student, enter marks, teacher leave.

2.2 Product Functions

The University Management System will perform the functions describe in the following diagram :



Step by step description:

1. The users will initially login into the system with their login credentials.
2. The admin can add new students and faculties into the system database and can also view and update the details of already existing students and faculties.
3. The faculty as a user can view their details and the details of the students.
4. The faculty can also enter and update marks of the students for the respective subjects.
5. Also, the teachers can approve the leave applied by the students.
6. The students can view their details, view their marks and apply for leave.
7. All the users can view the fee structure of the university and admin can update the fee structure.
8. The students have the access to the fee form through which they can pay the university fees.

2.3 User Classes and Characteristics

The user classes of the University Management System will be Admin,Teacher and Student.

Admin can view all of the information that is stored in the database through application and admin also can modify this information because the admin has full access to the system.
The teacher can view and modify their information as well as information related to students, teachers have limited access.
The student can modify their information only and can see,marks, apply for leave.

Classes and their attributes and methods:

1. User
   Attributes: name, fname, age,dob,address,phone,email,classX,classXII,adhar
   Methods: ViewDetails()
2. Student
   Attributes: rollno,branch
   Methods: Register(),Login(),UpdateDetails(),ViewDetail(),ApplyLeave(),ViewMarks()
   ViewFeeStructure(),PayFees()
3. Teacher
   Attributes: education,department,empid
   Methods: Register(),Login(),UpdateDetails(),ViewDetail(),ApplyLeave(),ViewMarks()
   ViewFeeStructure(),GrantLeave(),EnterMarks()
4. Admin
   Attributes: education,department,adminid
   Methods: Login(),AddTeacher(),AddStudent(),ViewDetail(),ViewMarks()
   ViewFeeStructure(),UpdateFeeStructure()
5. Login
   Attributes: username,password
6. FeeStructure
   Attributes: course,amt
   Methods: FeeStructure()
6. StudentFeeForm
   Attributes: name,fname,branch,sem,payment
   Methods: StudentFeeForm()
7. Marks
   Attributes: roolvo,subject,marks
   Methods: Result()
8. Leave
   Attributes: date,time
   Methods: ApplyLeave(),GrantLeave(),ViewLeaveDetails()
9. StudentLeave
   Attributes: rollno
9. TeacherLeave
   Attributes: empid

2.4 Operating Environment

The UMS is expected to be deployed in a real environment to manage the DBMS inside the college. The centralized database is used to store the information. The user only within the university (members of university staff) can use this management system. Users outside from the university cannot access the management system. This application is developed for windows operating system that can be run on Windows.

The database is used in different departments within a branch of the university. The database used to store the information is the centralized database. The software we have developed will be installed on different computer systems within a university and software will be connected to a centralized database through LAN within a university and then the user can interact with the system and can store the data and other users can get access the stored through a centralized database.

Requirements:

1. Hardware Requirements:
A server or cloud-based environment with sufficient processing power and memory to handle the inventory management software.
2. Operating System Requirements:
The system should be compatible with the latest version of popular operating systems such as Windows.
3. Database Management System Requirements:
The system should support the latest versions of popular database management systems such as SQL Server.
4. Security Requirements:
The system must have appropriate security measures such as user authentication, data encryption, and access control to protect data and prevent unauthorized access.
7. Backup and Recovery Requirements:
The system must have a backup and recovery plan to ensure that data can be restored in the event of a system failure or data loss. The backup and recovery plan should be regularly tested to ensure that it is effective.

2.5 Design and Implementation Constraints

During the implementation of the product, different challenges are faced. Choosing the interface for the management system was a paramount issue. Connecting the database with the application was a major problem.

For connecting the database we had to create our account in ORACLE and then we had to download the driver(software). The connection of the database that is created in ORACLE with java is not very simple as like SQL. So the installation of ORACLE driver(software) is necessary to create a connection between ORACLE and java. But after installing the required driver it creates a problem in installing and connecting with a server in the oracle server, so we decided to leave the oracle and then we choose the SQL server to create the database.

The SQL Server is easy to install and connect with a server in SQL it is very easy to understand the implementation of the database and also easy to create a new database and connect with the GUI application.

2.6 User Documentation

The user documentation for the University Management System will include user manuals, on-line help, and reports. The documentation will be available in electronic form.

2.7 Assumptions and Dependencies

The University Management System assumes that University has reliable internet.It also assumes that users have the capability to use database .It also depends on accuracy and reliability of data, hardware and software infrastructure.

3. External Interface Requirements

3.1 User Interfaces

The user interface (UI) for the University Management System should be intuitive, user-friendly, and visually appealing. It should provide easy navigation and efficient access to the system's various functionalities. Here is a description of the key components and screens that should be included in the UI:

1. Login Screen:
   The first screen displayed to users, prompting them to enter their login credentials.
   Includes fields for username and password.
   Option to select the user type (e.g., Administrator, Faculty, Student).

2. Dashboard:
   Upon successful login, users are directed to the dashboard.
   Provides an overview of important information and quick access to frequently used features.

3. Navigation Menu:
   A prominent and easily accessible menu bar or sidebar for navigation.
   Categorizes system functionalities into
   logicasection(e.g.,Master,Details,Leave,Examination
   Update Details,Fee Details,Utility).
   Each section expands to display sub-menu options for specific tasks.

4. Student Management:
   Student Profile: Allows students to view and update their personal information, contact details, and profile picture.
   Leave: Enables students to apply for leave.
   Grades: Displays a comprehensive view of a student's grades, attendance records, and academic progress.
   Fee: Enables students to view Fee Structure and fee form for payment.

5. Faculty Management:
   Faculty Profile: Provides faculty members with the ability to view and update their personal information, expertise, and contact details.
   Performance Evaluation: Allows administrators or designated personnel to evaluate faculty performance based on predefined criteria.
   Mraks: Enables teachers to enter marks and view marks .

6. Administrative Functions:
   User Management: Enables administrators to manage user accounts, including registration, activation, deactivation, and password reset.
   System Configuration: Provides options to configure system settings.
   Reporting: Generates various reports, such as student enrollment reports, faculty performance reports, etc.

7. Responsive Design:
   The UI should be designed to be responsive and compatible with different screen sizes including desktops.
   Use responsive layout techniques to ensure optimal viewing and interaction across devices.

3.2 Hardware Interfaces

The University Management System will require a standard desktop computer running Windows with an internet connection. The software should use communication protocols that are compatible with the hardware components of the system. This may include standard protocols such as TCP/IP, HTTP, and other protocols used by specific hardware devices. The software product should be able to interface with the physical connectors and interfaces used by the hardware components of the system.

3.3 Software Interfaces

The University Management System will need to be connected to several software components in order to function properly. Required connection will be:

1. Database: The Smart Inventory Management System will be connected to a database that stores all of the inventory data. The database will be SQL and the system will use Structured Query Language to retrieve data from the database.
2. Operating System: System will be designed to work on Windows XP and above operating system.
3. Programming Language: It will be developed using Java programming.
4. Libraries: The system will use several libraries .

4.System Features

4.1System Feature 1(Functional Requirements)

4.1 Registration/Login

4.1.1 Description and Priority

This feature enables users to register for accounts and log into the system securely.

4.1.2 Stimulus/Response Sequences

User Actions:

1. Enter necessary information to create a new account.
2. Enter login credentials to access the system.

System Responses:

1. Display error message if invalid login credentials.
2. Display success message upon successful login.

4.1.3 Functional Requirements

REQ-1: The system must allow users to create unique username and password.

REQ-2: The system must verify that passwords meet complexity requirements and are encrypted for storage.

REQ-3: The system must provide a secure login mechanism and prevent unauthorized access.

4.2 Student Management

4.2.1 Description and Priority

This feature enables students to view and update their profile information, view marks,
Fees structure and apply leave.

4.2.2 Stimulus/Response Sequences

User Actions:

1. View and Update Profile.
2. View Marks.
3. Fees Structure
4. Apply Leave

System Responses:

1. Display students profile and update option for profile update.
2. Display marks of student.
3. Display fees structure and fees form for payment.
4. Display apply leave section

4.2.3 Functional Requirements

REQ-1: The system must allow students to view and update their profile.

REQ-2: The system must display marks of students

REQ-3: The system should display fees structure and provide fees form to students.

REQ-4: The system should display apply leave section.

4.3 Faculty Management

4.2.1 Description and Priority

This feature enables teachers to view and update their profile information, view marks,

Enter marks,apply leave and grant leave.

4.2.2 Stimulus/Response Sequences

User Actions:

1. View and Update Profile.
2. View Marks and Enter Marks
3.Apply Leave and Grant Leave

System Responses:

1. Display teachers profile and update option for profile update.
2. Display marks of student and enter marks section.
3. Display apply leave and grant leave section.

4.2.3 Functional Requirements

REQ-1: The system must allow teachers to view and update their profile.

REQ-2: The system must display marks of students and enter marks section for entering marks of student.

REQ-3: The system should display grant leave section.

REQ-4: The system should display apply leave section.

4.4 Administrative Functions

4.4.1 Description and Priority

This feature provides all administrative functions.

4.4.2 Stimulus/Response Sequences

User Actions:

User Roles and Permissions

System Responses:

1. Display associated role with the user.

4.4.3 Functional Requirements

REQ-1: The system should properly display according to users role.

REQ-2: Define different user roles (e.g., administrator, faculty, student) with specific permissions and access levels.

REQ-3: Manage user roles and their associated permissions.

5.Other Nonfunctional Requirements

5.1Performance Requirements

The system should be able to handle a large volume of data and provide real-time data. The system should have the following performance requirements:

- The system should be able to handle data entry.
- The System should be reliable and easy to use.
- The system should provide real-time data with a maximum latency.

5.2Safety Requirements

1. Authentication: The system must require users to authenticate themselves before accessing any sensitive data or performing any sensitive operations.
2. Authorization: The system must enforce role-based access control, ensuring that users can only access the data and functionality that they are authorized to use.
3. Data Encryption: The system must use encryption to protect sensitive data both in transit and at rest.

6. Other Requirements

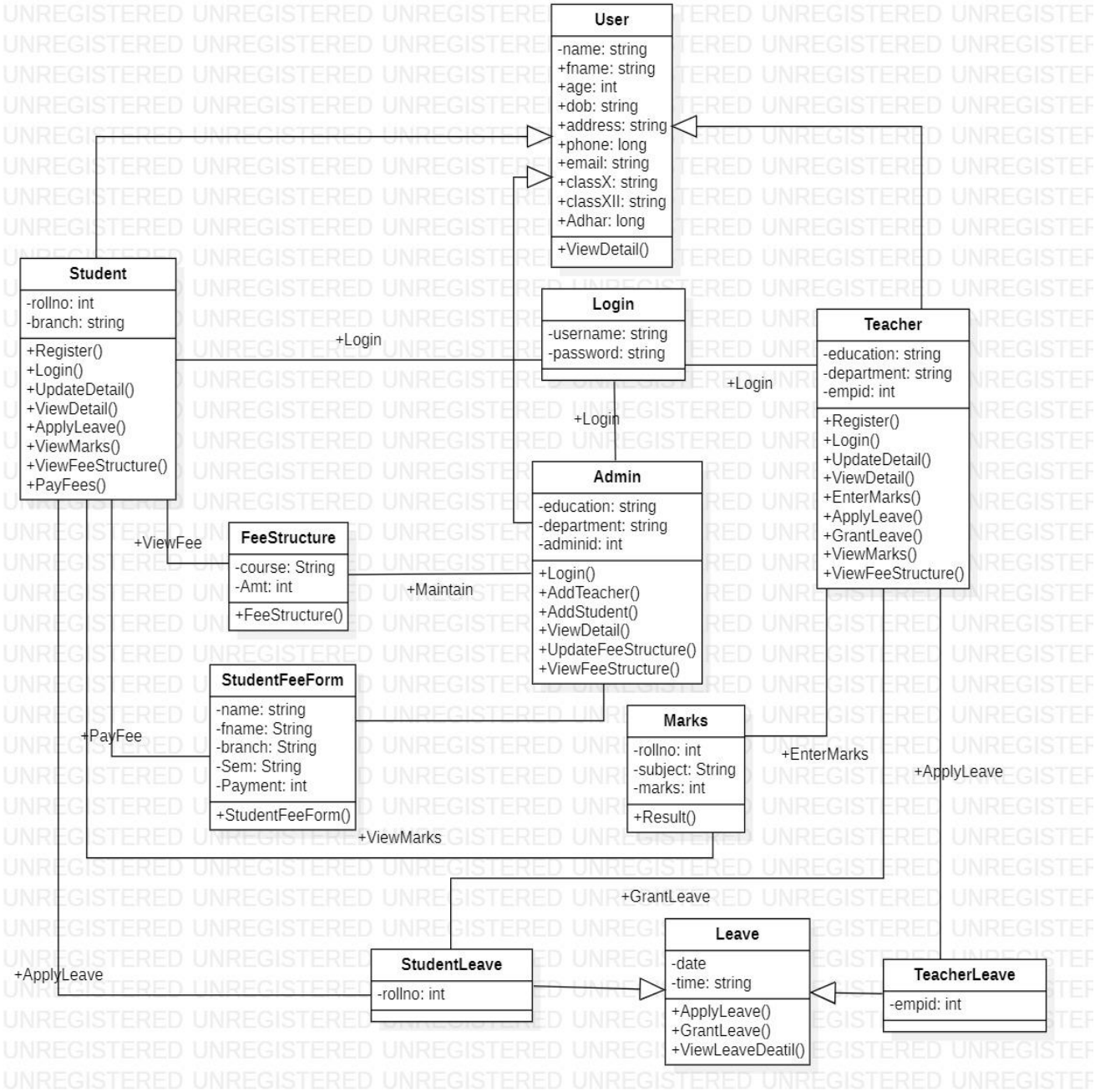
Here are some additional requirements that can be considered for a University management system:
1. Mobile support: The system should provide mobile-friendly interfaces or mobile apps for users who need to access inventory information or perform transactions on the go.
2. Customization: The system should support customization and configuration options, such as user roles, permissions, workflows, or branding, to meet the specific needs of different users, businesses, or industries.
3. Localization: The system should support localization requirements, such as language translations, regional settings, or currency conversions, to enable global or multilingual use

Appendix A: Glossary

Glossary for the Smart Inventory Management System:
1. SRS: Software Requirements Specification
2. DBMS: Database Management System.
3. GUI: Graphical User Interface
4. UMS: University Management System.
5. SQL: Structured Query Language
6. SQL Server: SQL Server is a database management system in which database is created and manages the database like “update”, “Delete” and insert a new record in the database.

Appendix B: Analysis Models

Class Diagram:



Appendix C: To Be Determined List

Mobile access refers to both the question of whether or not a system should be usable on mobile devices as well as the availability of particular functionalities.
System updates and maintenance: How the system will be maintained over time, including how often updates are released and how users are informed about them.
How users will be trained on the system, the resources that will be made available for continuous support, and the channels that will be open for user feedback and help requests are all
covered under training and support.