Department of Information and Communication Technology

Faculty of Technology

University of Ruhuna

Object Oriented Programming Practicum

ICT 2132

Software Requirement Specification

GROUP 04

SUBMITTED BY

TG/2021/1017 M.M.N.D.Seneviratne

TG/2021/1048 A.R.M.Geethanga

TG/2021/1081 M.D.V.Dias

TG/2021/1054 W.M.D.U.Wanasingha

Table of Contents

3. Introduction

3.1 Purpose of the document

The document represents complete and comprehensive specification and description of requirements of the software that is needed to be accomplished for the successful deployment of Management Information System for the Faculty of Technology, university of Ruhuna.

3.2 Scope of the document

Requirements covered by the document include functional and as well as non - functional depending upon the type of requirements. Specific users that facilitate by the proposed system and how they interact with the GUI environment is explained briefly in the document. Furthermore, interface and performance requirements are defined generally along with other non-functional attributes.

3.3 Overview

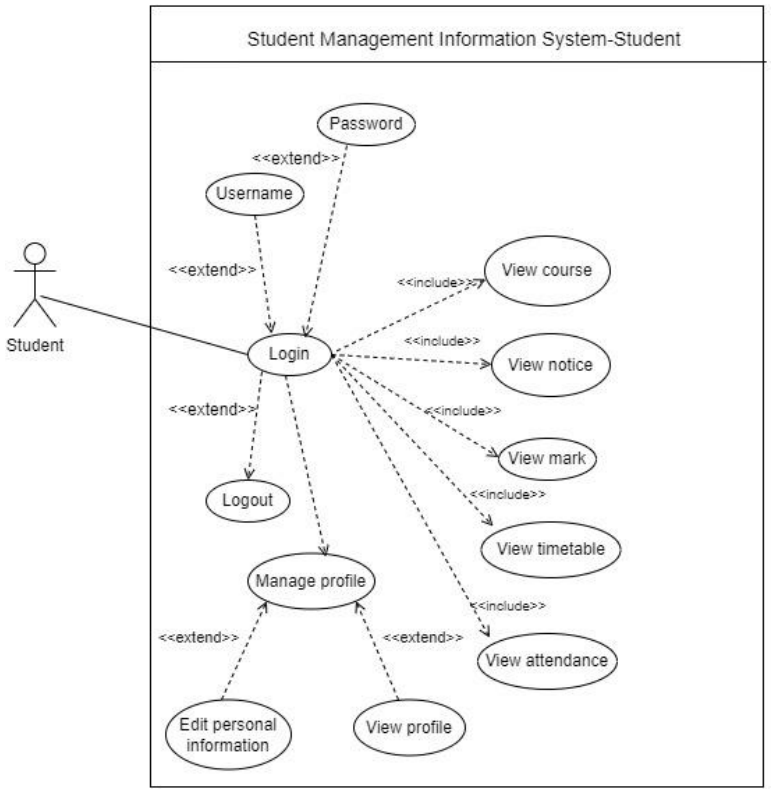
The overall system is built to handle mainly user profiles, course details, student marks, student attendance, notices, timetables and medicals interacting with the following users' admins, technical officers, lectures and students through integrated interfaces. And the system is engaged with a database and specified functionalities are provided by the system for each user scope accordingly. The MIS can be operate as a standalone system for the faculty by automating processes and information.

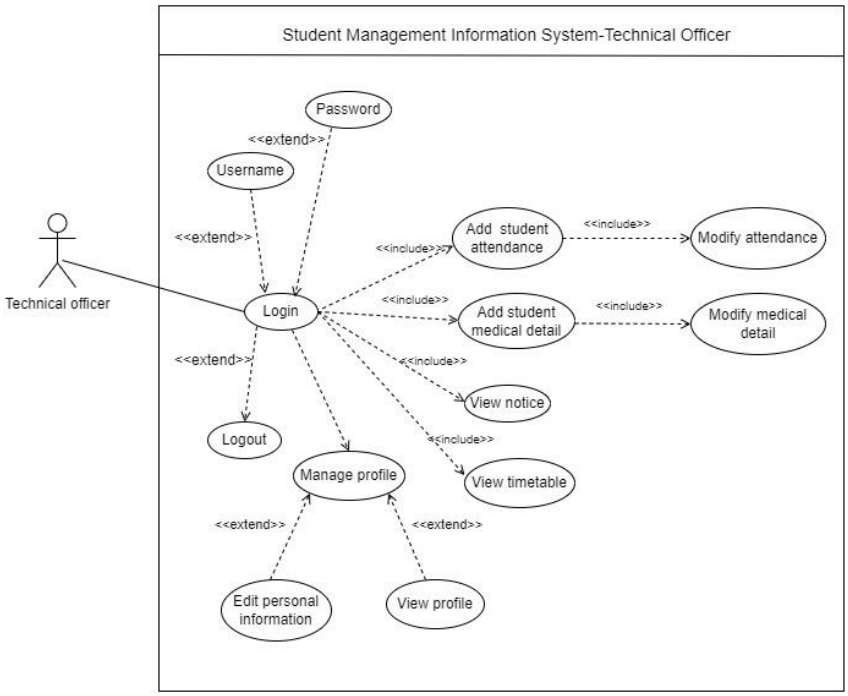
4. General Description

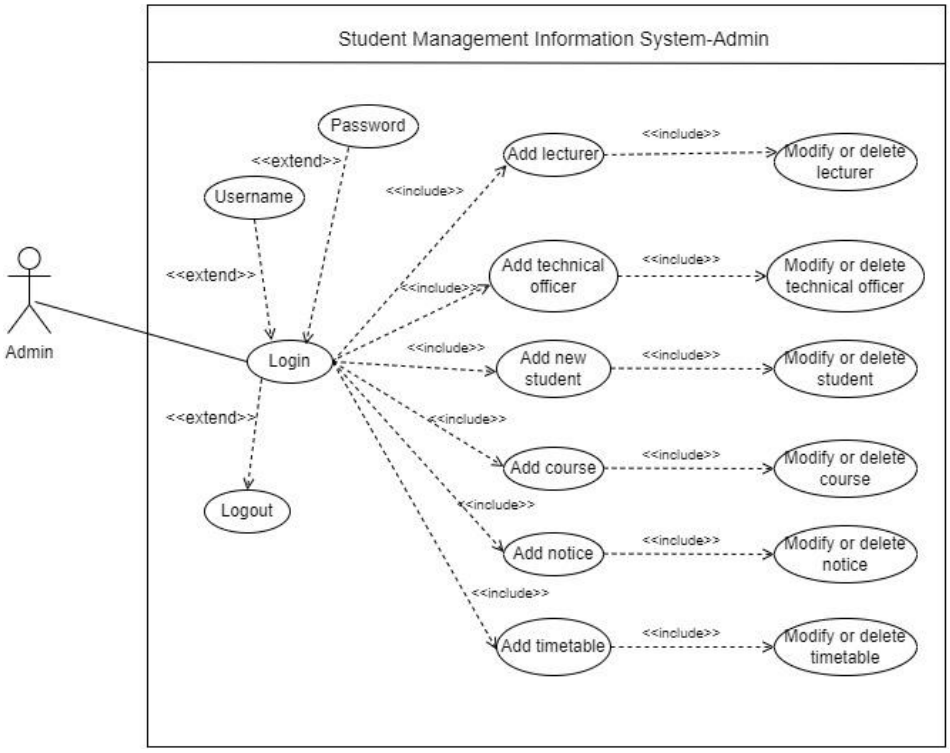
The system contains a GUI login to the system which require login credentials from users and by entering details users navigate to own interfaces that made up by their scope. By maintaining database, system can handle user accounts, and keep course details, attendance details, exam marks etc. Each user is given functionalities according to their specific requirements. Moreover, Interfaces are build responsive and features are added visualizing user-friendly attributes.

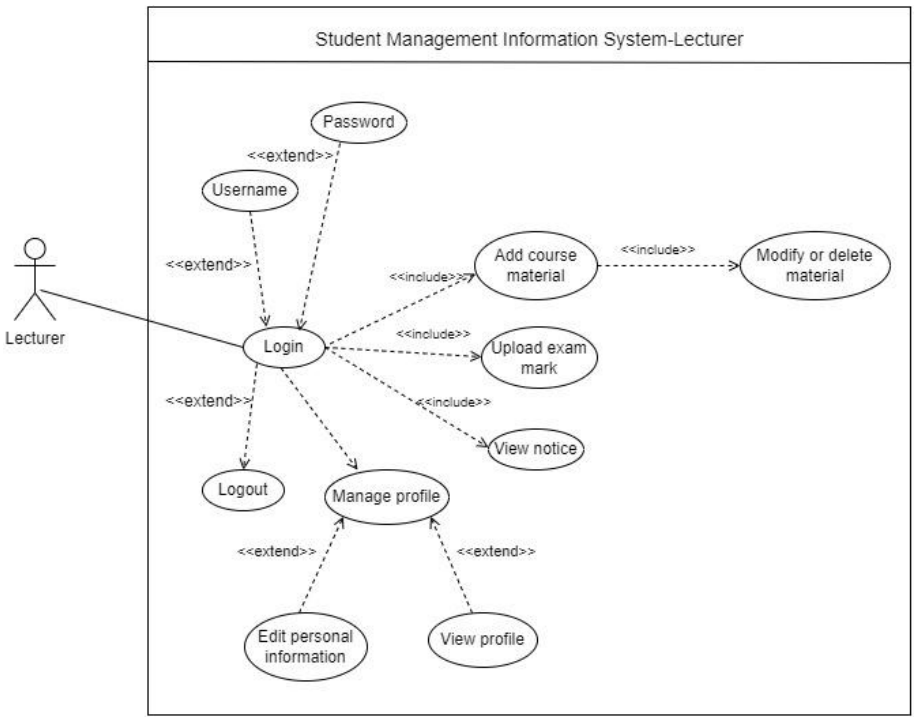
4.2 Use Case diagrams

Use case diagram are used to illustrate use cases, actors and the relationships between them. Here are isolated views of use case diagrams for each scenario.




Student Management Information System-Technical Officer
Password
<<extend>>
Username
<<extend>>
Technical officer
Login
<<include>>
Add student attendance
<<include>>
Modify attendance
<<include>>
Add student medical detail
<<include>>
Modify medical detail
<<include>>
View notice
<<include>>
View timetable
<<extend>>
Logout
Manage profile
<<extend>>
<<extend>>
Edit personal information
View profile


Student Management Information System-Admin
Password
<<extend>>
Username
<<extend>>
Admin
Login
<<include>>
Add lecturer
<<include>>
Modify or delete lecturer
<<include>>
Add technical officer
<<include>>
Modify or delete technical officer
<<include>>
Add new student
<<include>>
Modify or delete student
<<include>>
Add course
<<include>>
Modify or delete course
<<include>>
Add notice
<<include>>
Modify or delete notice
<<include>>
Add timetable
<<include>>
Modify or delete timetable
<<extend>>
Logout



4.3 User Objectives

- Admin

The user admin should be able to

1. Create and maintain all user profiles within the system.
2. Create and maintain courses.
3. Create and maintain notices.
4. Create and maintain timetables

- Technical officer

The user technical should be able to

1. Update their profile except user name and password
2. Add and maintain attendance details of students.
3. Add and maintain medical details of students.
4. See notices.
5. See timetables of their department.

- Lecture

The user lecture should be able to

1. Update their profile except user name and password
2. Modify and add materials to courses.
3. Upload marks for all kind of exams.
4. See student details.
5. See student eligibility for exams.
6. See student marks, grades and GPA
7. See attendance and medical records of students
8. See notices.

- Student

The user student should be able to

1. Update only contact details and profile picture of their profile
2. See attendance details
3. See medical details
4. See their course details
5. See their grades and GPA
6. See their timetables
7. See notices

4.3 User characteristics

- Admin: Only admins can create and maintain all user profiles, notices and timetables in the system.

- Student: Students log in to system for mainly academic purposes. Students are not allowed to update their account's username and password.

- Lectures: Lectures can add and modify materials for courses, and upload marks for all kind of exams in the system.

- Technical officer: Only Technical officer can add maintain attendance details and medical details of student in the system.

5. Functional requirements

Functional requirements specify the expected behavior of the system-which outputs should be produced from the given inputs. They describe the relationship between the input and output of the system .Here is a list of functional requirements the purposed system might have,

1. The system should have a graphical login interface which prompts users to enter their username and password as login credentials.
2. The system should identify each user uniquely by their usernames and passwords.
3. Every user should be able interact with a graphical user environment after logged in.
4. The system should have the ability to connect with the database anytime.
5. Every user should have the ability to logout from the system anytime.
6. Some privileged users should be able to change their password and username.
7. Some users should be able to create user profile, courses, timetables and notices
8. The system should maintain timetables, notices, and profile pictures for all users.
9. The system should have the ability serve many users at the same time.
10. The system should have the ability to inform users if there is a database error.

6. Interface Requirements

6.1 User interfaces

The user interface for the Student Management Information System will provide a user-friendly and intuitive experience.

6.1.1 Admin Interface

1. Should provide options for maintaining user accounts, courses, notices, timetables, and other administrative tasks.
2. User-friendly interface for easy navigation and data manipulation.
3. Access control features to restrict unauthorized access to administrative functionalities

6.1.2 Lecturer Interface

1. Ability to manage and upload course materials and marks for all kind of exams.
2. Should provide options for see student details, student eligibilities, student marks, attendance and medical records of students.
3. Also provide options for see notices and update their profile. (except username and password)

6.1.3 Student Interface

1. Access to course materials, grades and academic notices.
2. Ability to view attendance details, medical details, course details, exam marks, and their timetables.
3. Also provide option for update their profile. (only contact details and profile picture)

6.1.4 Technical Officer Interface

1. Should provide options for maintaining attendance details and medical details of students.
2. Ability to view notices and timetables of their department.
3. Also provide option for update their profile. (except username and password)

6.2 Hardware Interfaces

1. The system should be compatible with standard hardware configurations.

2. It should support various hardware specifications such as different CPU architectures, RAM capacities, and storage options to accommodate different user needs and device capabilities.

3. Minimum hardware requirements (CPU, RAM, storage) for optimal performance. It depicts below table. (This is not essential)

CPU	Intel Core i5 or equivalent
RAM	Minimum 4 GB
Storage	a minimum disk space requirement for storing application files and data (<100 GB)

6.3 Communication Interfaces

None

6.4 Software Interfaces

6.4.1 Java Interface

1. The system should be developed using Java programming language following Object-Oriented Programming (OOP) concepts.
2. Compatibility with Java development frameworks and libraries for efficient development.

6.4.2 MYSQL Database Interface

1. Interaction with MySQL database for storing and retrieving data related to users, courses, grades, etc.
2. Implementation of secure database transactions to ensure data integrity and confidentiality.

7. Performance Requirements

Performance requirements defines how proposed system performs desired functions under specific condition is explained as well as the performance constraints on the software system. All the requirements relating to the performance characteristics of the system are clearly specified here.

- Response time: The acceptable response time for operations within the system should be less as it doesn't take more than few seconds. For instances, user authentication should take no longer than one or two seconds, loading user profiles should take no longer that one or two seconds.

- Throughput: The system should handle more concurrent users without significant degradation of the performance.

- Concurrency: The system should support for many requests from users without contentions and congestions. This includes concurrent access to shared resources like notices and timetables.

- Accessibility: The system should implement interfaces with high accessibility to specified information and resources with ease for users.

- Reliability: The system should maintain information in the database following data integrity parameters along with backup and recovery facilities.

- User experience: The system should provide it's users with responsive, interactive and user friendly graphical environment.

- Resources utilization: The system should maintain its resources at low implementation and operational costs for the faculty.

8. Non-functional attributes

Non-functional attributes are explained that are required by system for better performance.

- Security: The system includes user authentication and authorization mechanisms to ensure security measures and only some users can allowed to privileged actions.

- Usability: User interfaces are designed should ensure ease of use, consistency and responsiveness.

- Maintainability: The system is flexible to and easy for modifications according to the faculty regulations and policies.

9. Conclusion

In conclusion, the SRS document for the proposed MIS system provides a comprehensive overview of the functional, non-functional requirements along with interface requirements and Performance requirements. Other key areas are also explained in descriptive and informative manner providing overall details for designing and implementing a successful management information system for the faculty of technology, university of Ruhuna.

********** **End of the Document** **********