SOFTWARE LABORATORY MANAGEMENT SYSTEM

Group Number : 19
Group Members :
1) Alok Ranjan - 2022CSB091
2) Ranveer Kumar - 2022CSB092
3) Karan Kumar - 2022CSB093
4) Rajesh Kumar - 2022CSB094

PROBLEM UNDERSTANDING:

The Software Laboratory Management System facilitates students, lab-in-charge, faculty members, and lab assistants in managing lab resources, scheduling, issue reporting, and monitoring.

INTRODUCTION TO SYSTEM

- Effective management of software laboratories is critical for smooth academic operations.
- Challenges such as resource conflicts, manual scheduling, lack of transparency, and poor communication hinder productivity.
- The **Management of Software Laboratories** system is designed to streamline lab scheduling, resource management, and attendance tracking.
- Key Users : **students**, **faculty members**, **lab-in-charges**, **lab assistants**.
- The goal is to improve lab resource allocation, reduce scheduling conflicts, and enhance overall lab management.

KEY ACTION AND USERS IN THE SYSTEM

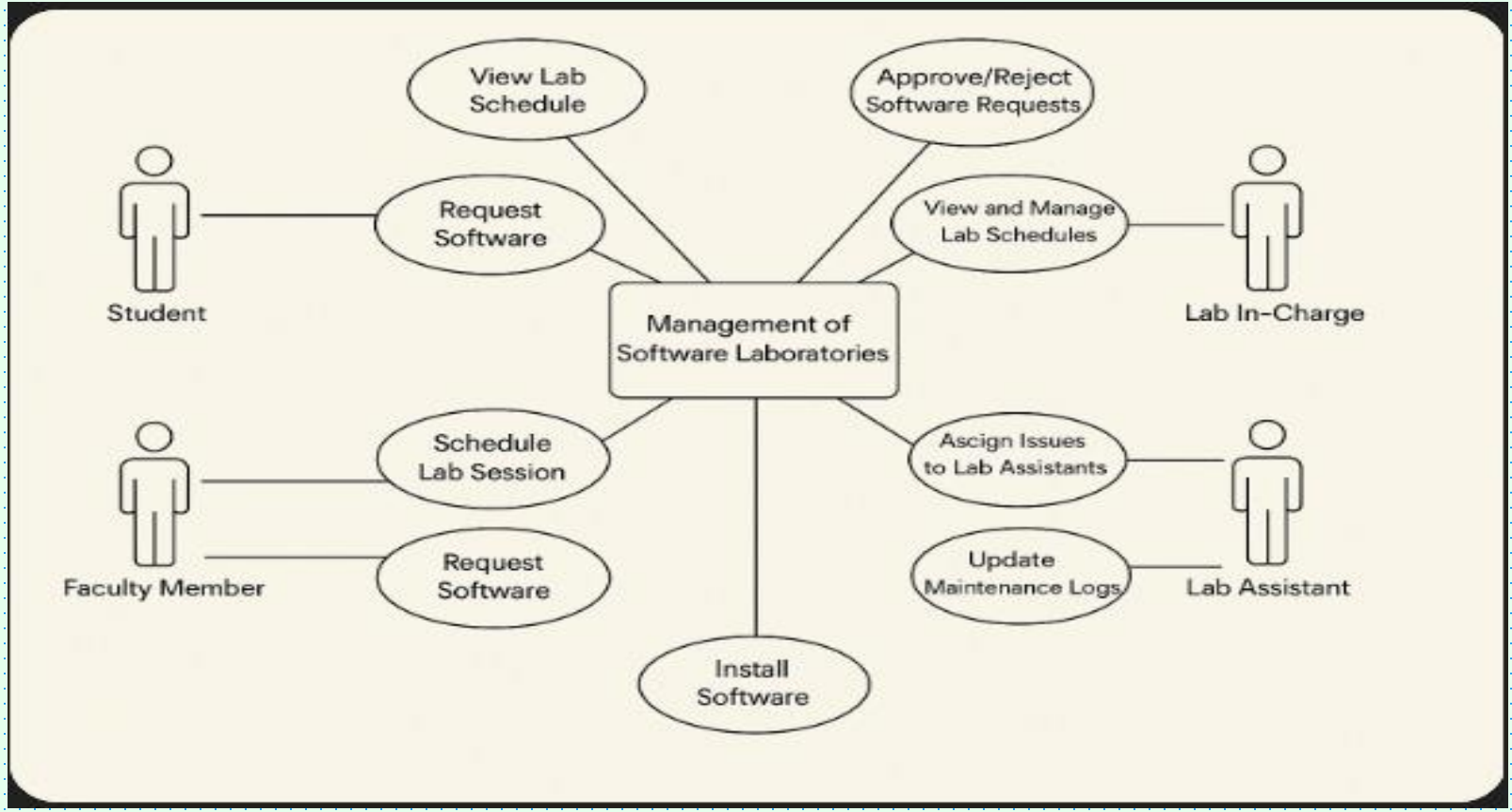
- **Student**: Can book, cancel, and view bookings.
- **Faculty Member**: Can book labs for classes, request resources, and track attendance.
- **Lab-In-Charge**: Oversees lab schedule, approves/rejects bookings, and manages resources.
- **Lab Assistant**: Assists students and faculty and tracks attendance.

USE-CASES

A Use-Case Diagram represents the interactions between users (actors) and the system functions.

Users	Use Cases
Student (Requests lab access, reports issues)	Lab Reservation: Faculty reserves labs; lab-in-charge approves.
Faculty Member (Reserves labs, monitors student activities)	Resource Allocation: Students request software/hardware; lab-in-charge processes requests.
Lab In-Charge (Manages schedules, approvals, and reports)	Monitoring & Logs: Lab-in-charge tracks usage logs and generates reports.
Lab Assistant (Maintains equipment, assists users)	Issue Reporting: Students and faculty report issues; lab assistants fix them.

USE-CASE DIAGRAM REPRESENTATION



OVERVIEW OF FUNCTIONAL REQUIREMENTS

- The system includes lab booking, resource management, attendance tracking, notifications, and reporting functionalities.
- Each user role has specific tasks and permissions.

LAB BOOKING AND SCHEDULING SYSTEM

- **Students** can request for booking individual or group lab slots.
- **Faculty Members** can schedule labs for classes and practical.
- **Lab-In-Charge** can approve or reject booking requests, ensuring no conflicts.
- **Booking Notifications** are sent to users upon confirmation.

LAB RESOURCE MANAGEMENT

- **Resources** such as computers, software, and projectors are tracked.
- **Faculty Members** can request specific resources during their booking process.
- **Lab-In-Charge** updates resource availability status.

ATTENDANCE AND USAGE MONITORING

- **Lab Assistants** track student attendance for booked slots.
- The system prevents unbooked students from using lab resources.
- **Reports** can be generated for lab usage and attendance.

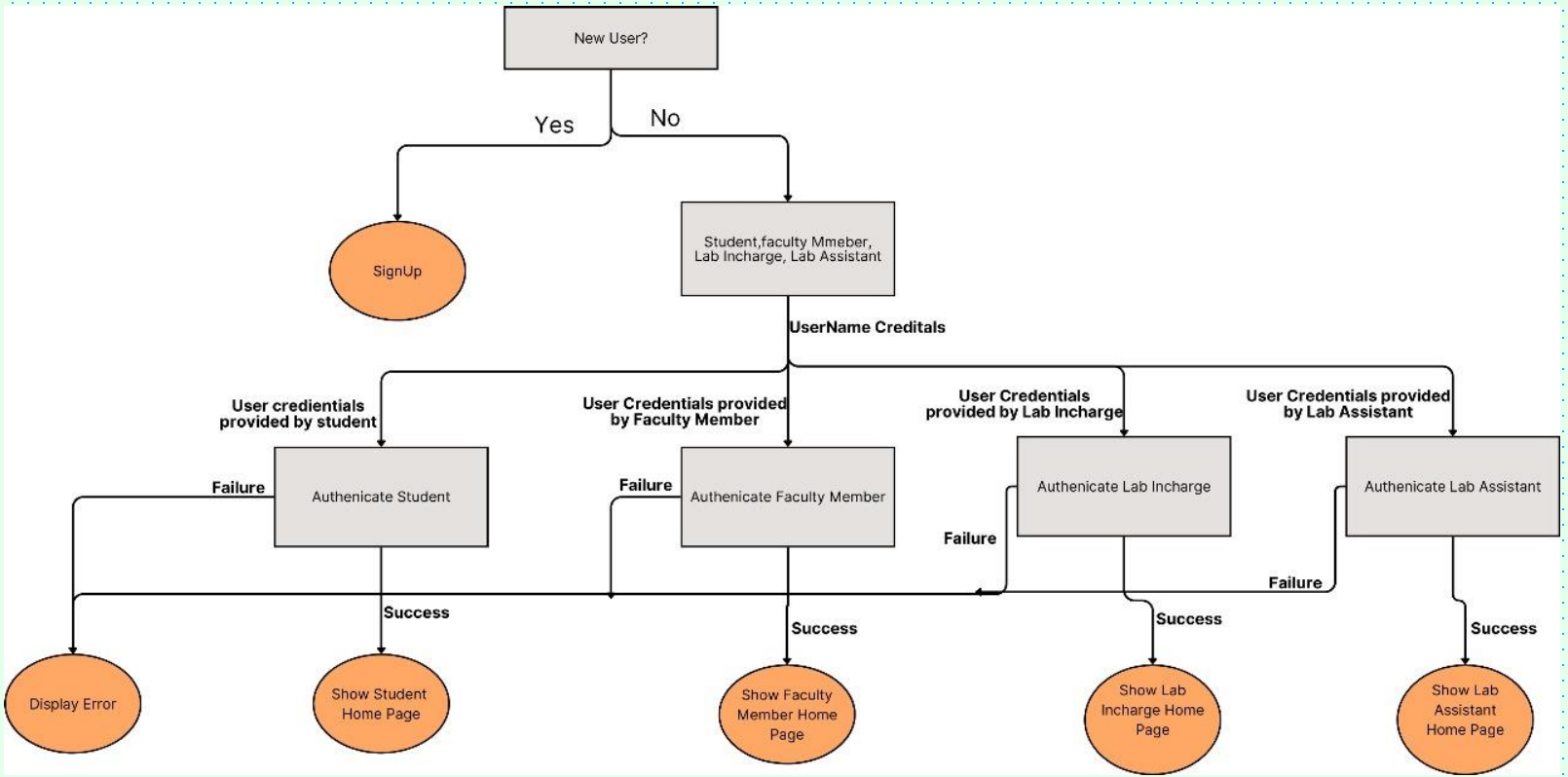
NOTIFICATIONS AND ALERTS

- **Students** receive notifications on booking confirmations and cancellations.
- **Faculty Members** are alerted if their booking is unavailable or if changes occur.
- **Lab-In-Charge** is notified about booking conflicts, resource issues, and maintenance needs.

SECURITY AND ACCESS CONTROL

- The system uses **user authentication** with usernames and passwords.
- **Role-based access control** ensures that users only access relevant features.
- **Password security** and **data encryption** are enforced.

AUTHENTICATION FLOW



REPORTING AND ANALYTICS

- **Lab-In-Charges** can generate usage reports and track resource allocation.
- **Faculty Members** can view student attendance and session participation.
- **Reports** can be exported for administrative purposes.

NON-FUNCTIONAL REQUIREMENTS

- **Scalability**: The system should handle up to 1000 users without significant performance loss.
- **Reliability**: Ensure 99% uptime, with regular backups.
- **Security**: Encrypted data storage, secure login, and access control mechanisms.
- **Performance**: Quick response time (booking process within 3 seconds).

CONCLUSION

The **Management of Software Laboratories** system will optimize lab scheduling, resource allocation, and student attendance tracking.

- By fulfilling the outlined functional and non-functional requirements, the system aims to improve the overall management of the software lab environment.

THANKS!