

**DATABASE MANAGEMENT SYSTEM**  
**PROJECT FINAL REPORT**

**SANAS M.M.**  
**2022/E099**  
**EC5070**

# **TITLE: LABORATORY BOOKING SYSTEM FOR UNIVERSITY ACADEMIC SESSIONS**

## **INTRODUCTION:**

In modern educational institutions, practical sessions play a crucial role in reinforcing theoretical knowledge, especially in science and engineering programs. Laboratories are critical environments where students gain hands-on experience with tools, software, and equipment. However, managing the scheduling and usage of these labs presents various challenges. Traditional manual booking systems often result in mismanagement, scheduling conflicts, poor equipment tracking, and inefficient use of resources.

To address these challenges, a Laboratory Booking System is essential. It enables streamlined coordination between instructors, lab technicians, and students by automating the entire booking and management process. Such a system ensures that laboratories are reserved in a structured manner, resources are allocated efficiently, and the equipment availability is verified prior to bookings. Additionally, logging usage data allows institutions to generate reports, analyse trends, and make informed decisions on lab maintenance, expansion, or upgrades.

This project aims to design and implement a database-driven Laboratory Booking System that supports real-time availability checks, instructor-led bookings, equipment management, and usage reporting. The system supports multiple user roles (e.g., Students, Instructors, Lab Technical Officers, and Lecturers in Charge), each with tailored permissions. Through the integration of web technologies and relational database systems, the platform not only reduces manual workload but also enhances the transparency, traceability, and efficiency of lab operations.

By introducing this system, institutions can better manage their physical infrastructure and improve the overall quality of academic practical sessions.

## **PROBLEM STATEMENT**

Universities and educational institutions heavily rely on laboratories to conduct practical sessions that complement theoretical learning. These labs are shared resources used by multiple departments and courses. However, managing their availability, ensuring fair allocation, and handling equipment logistics remain significant challenges, especially when done manually.

In the current manual system, lab bookings are often recorded on paper or via informal communication, such as emails or messages. This leads to several recurring issues:

- **Double-booking** of labs for the same time slots by different instructors.
- **Lack of visibility** for instructors and staff to view real-time lab availability.
- **Poor tracking** of lab equipment, which can result in overuse, loss, or unavailability during practical.
- **Inconsistent approval processes**, where requests may be delayed or lost due to communication gaps.
- **No centralized record** of lab usage history, making it difficult to generate reports or analyse patterns.

These inefficiencies impact the quality of academic delivery, disrupt scheduled practical, and increase the administrative workload for staff managing the labs.

Therefore, there is a need for a **centralized, automated Laboratory Booking System** that:

- Allows instructors to check real-time lab availability and book accordingly.
- Ensures each booking is verified against lab capacity and equipment requirements.
- Maintains proper logs for all bookings and equipment usage.
- Provides role-based access control for students, instructors, and administrators.
- Generates accurate reports for academic and administrative review.

By addressing these issues through a structured, database-driven approach, institutions can ensure better lab utilization, reduce conflicts, and improve the overall experience for both students and faculty.

## **AIM**

The aim of this project is to develop a role-based Laboratory Booking and Management System that streamlines lab usage for students, instructors, and technical officers by offering tailored functionalities for each user type. The system is built to improve coordination, reduce scheduling conflicts, and manage lab equipment efficiently.

More specifically, the system aims to:

### **For Students:**

- Allow students to enrol in specific courses through the system.
- Enable students to view only the lab schedules related to their enrolled courses.
- Restrict students from accessing unrelated lab sessions or attending unscheduled practicals.
- Prevent students from accessing unrelated lab sessions or attending unscheduled practicals.
- Enforce that students can only attend labs when the session is officially scheduled.

### **For Instructors:**

- Enable instructors to initiate a lab booking by entering a Lab Code.
- Upon selecting a course practical, allow instructors to:
  - View available labs that are compatible with the selected practical.
  - View detailed equipment information available in each lab.
- Allow instructors to book free time slots for the selected lab based on real-time availability.

### **For Technical Officers (TOs):**

- Allow TOs to view upcoming lab sessions only in the labs they are assigned to.
- Enable them to view log details of all activities and bookings in their labs.
- Provide a feature to generate lab usage and equipment logs as downloadable PDF reports.
- Give them oversight over equipment usage and lab readiness.

# PROBLEM SOLUTION

To address the limitations of traditional lab management and booking systems, we have developed a **web-based Laboratory Booking System** that integrates database management, role-based access control, and dynamic scheduling. The solution is implemented using **PHP (backend)**, **MySQL (database)**, **HTML/CSS/JavaScript (frontend)** and is designed to serve multiple user roles with distinct capabilities.

## Key Features of the Solution

### 1. Role-Based User Access

- The system supports **four types of users**: Students, Instructors, Technical Officers (TOs), and Lecturers in Charge.
- Each role has specific permissions, ensuring users only interact with relevant parts of the system (e.g., students can't make bookings; only instructors can).

### 2. Lab Booking Workflow for Instructors

- Instructors initiate bookings by selecting a course and practical session.
- The system displays only the labs **suitable for the selected practical**.
- Available labs are filtered based on **real-time schedule data** and **equipment availability**.
- Instructors can then book free slots, which are stored in the lab\_booking table.

### 3. Student Access Control

- Students can **only access lab sessions** associated with the courses they are enrolled in.
- If a student attempts to join a lab session not scheduled or assigned to them, the system prevents access, ensuring integrity in lab usage.

### 4. Lab Availability and Equipment Management

- The system maintains a lab\_schedule table to track lab reservations.
- Equipment details are stored in the lab\_equipment table, which ensures that each lab booked includes the required tools for the session.
- This prevents sessions from being booked in labs that lack necessary resources.

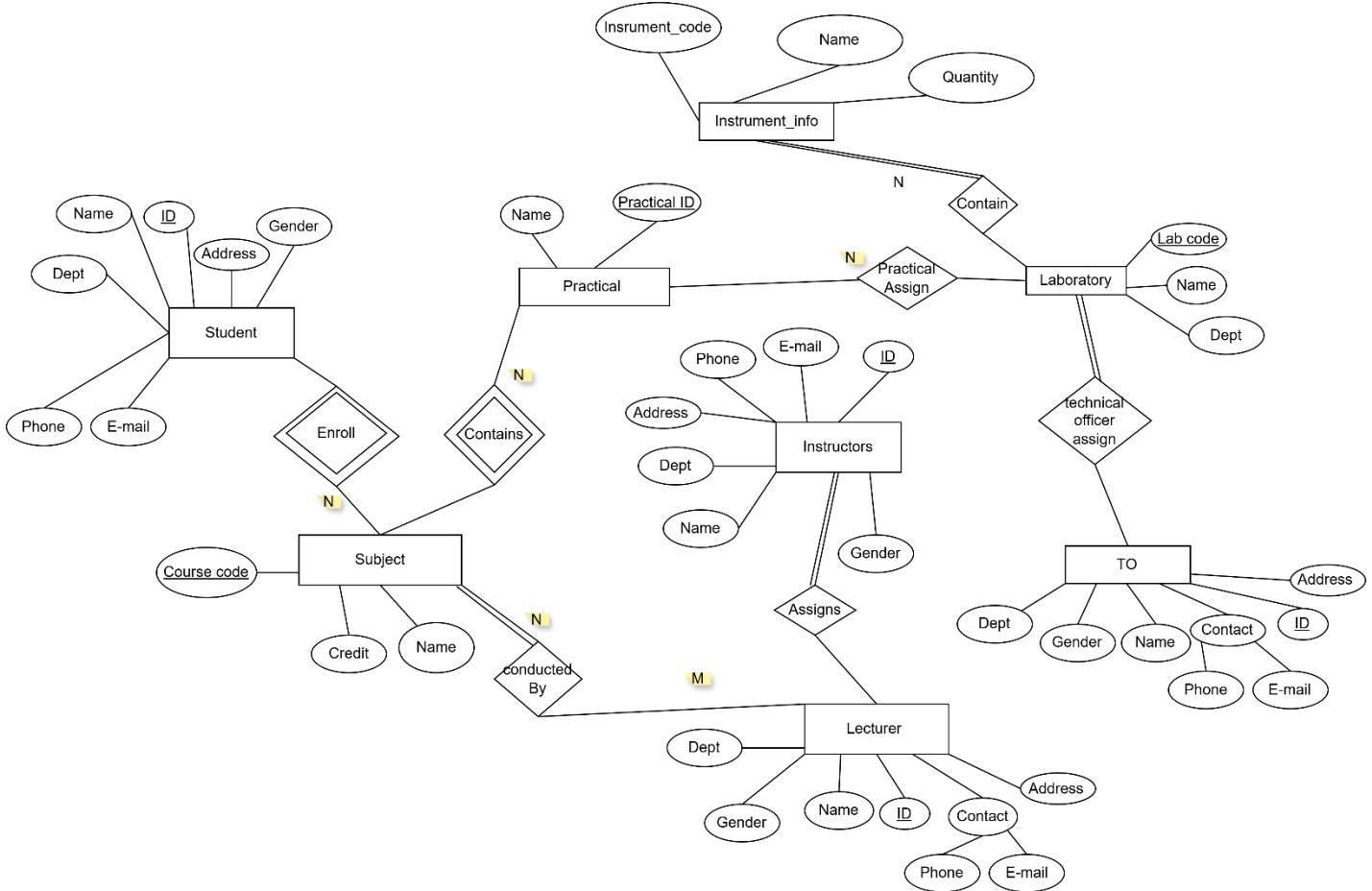
### 5. Technical Officer Interface

- TOs can view lab schedules **limited to their assigned labs**.
- They can monitor equipment usage and generate lab usage reports.
- A PDF generation feature allows them to download lab activity logs for offline documentation or auditing.

## 6. Logging and Reporting

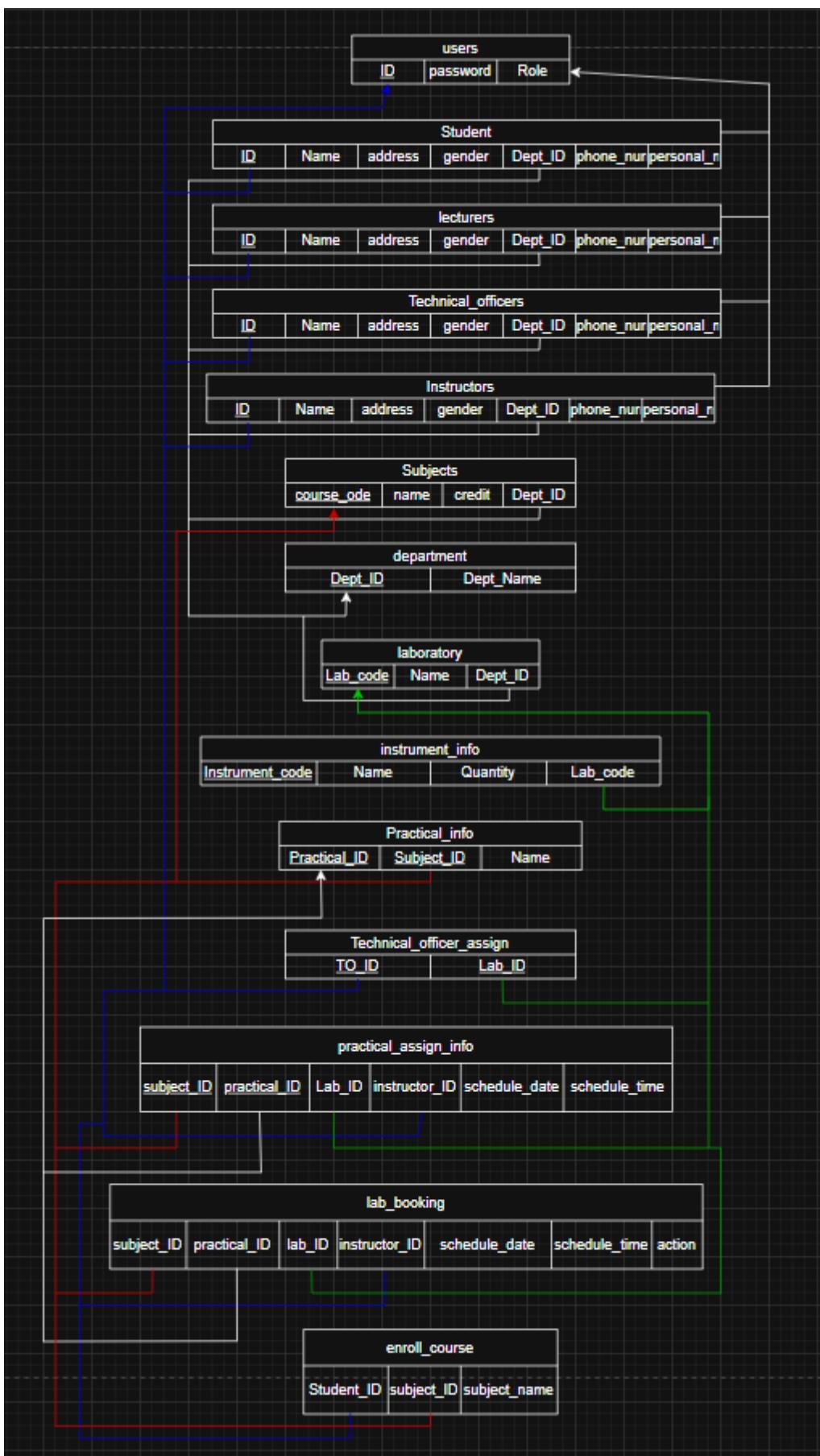
- All bookings and equipment usage are **stored in the database**.
- Logs are used to track history and generate **real-time or periodic reports**.
- Reports can be exported in PDF format, ensuring transparency and accountability.

## REVISED ER DIAGRAM

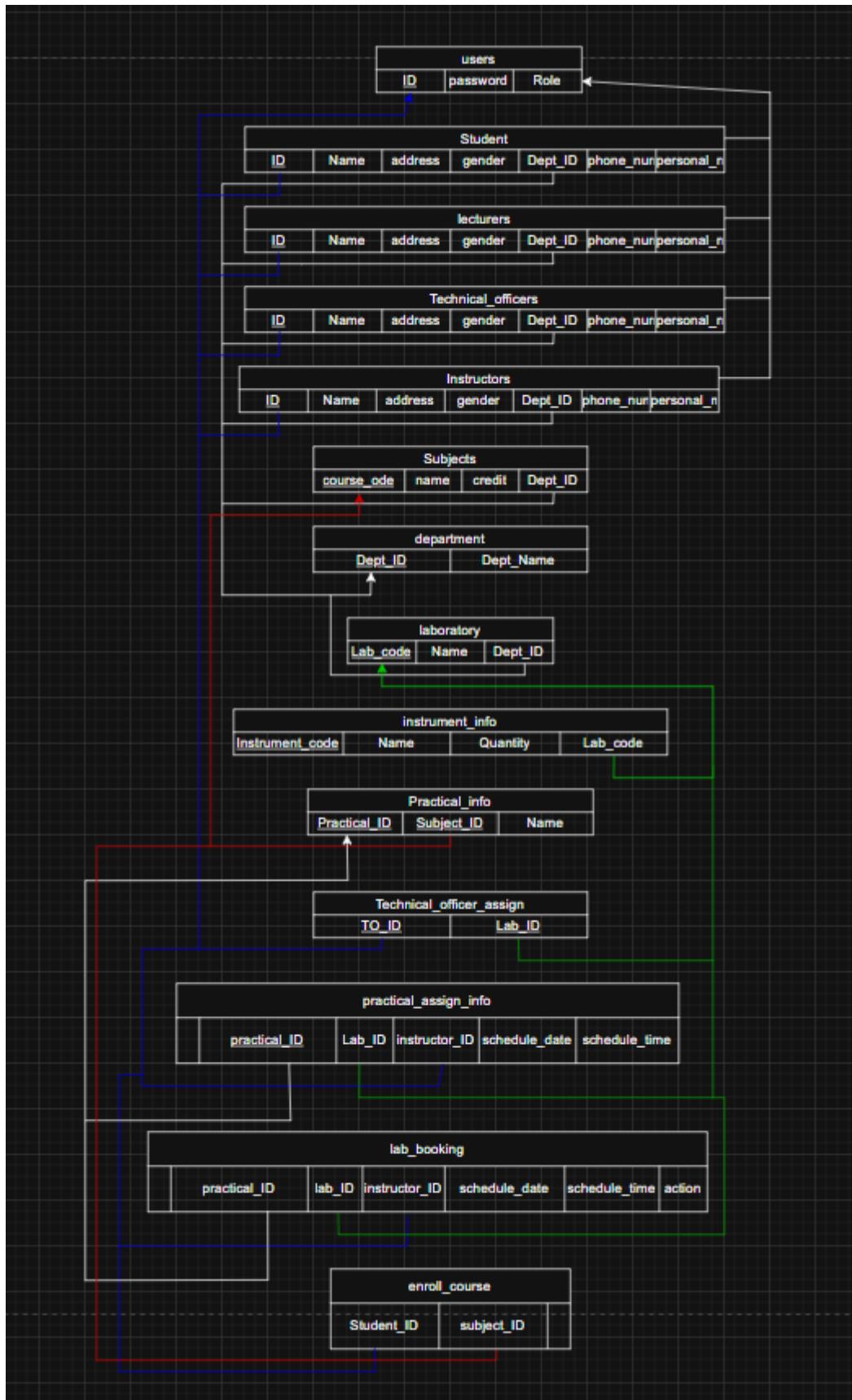


## REVOICED RELATIONAL SCHEMA DIAGRAM

In 1NF



## In 2NF



And its already in 3NF also.

## **KEY CHALLENGES**

### **1) Learning Web Development from Scratch**

One of the major challenges was the need to study and understand multiple web development technologies from the ground up:

- HTML – For structuring web pages
- CSS – For styling and layout design
- JavaScript (JS) – For dynamic interactions and client-side behavior
- PHP – For server-side scripting and handling database logic
- SQL – For managing and querying the MySQL database

Balancing learning and implementation was time-consuming and required a lot of trial and error, especially since each language had different syntax and logic structures.

### **2. Dynamic Value Handling in PHP**

A key issue faced during development was handling dynamic data transfer between frontend inputs and PHP processing files. In some cases, the expected values passed through forms or JavaScript were either not captured correctly in PHP or were displayed incorrectly (e.g., showing “ssss” or empty values). This made:

- Form validation difficult
- Data processing buggy
- Troubleshooting confusing due to the lack of meaningful error output

### **3. File and Folder Structure Confusion**

There were difficulties in understanding how file paths work across folders (e.g., assets, backend, database, etc.). When calling files like generate\_lab\_Booking.php or connecting to databases, issues occurred due to:

- Misreferenced relative paths (e.g., ../, ./)
- Assets not loading properly
- PHP scripts not finding included files
- No visual indication of current working directory in browser output

This made debugging and code linking between HTML, CSS, JS, and PHP files more complicated.

### **4. Ensuring Secure and Correct Booking Logic**

Implementing correct logic to:

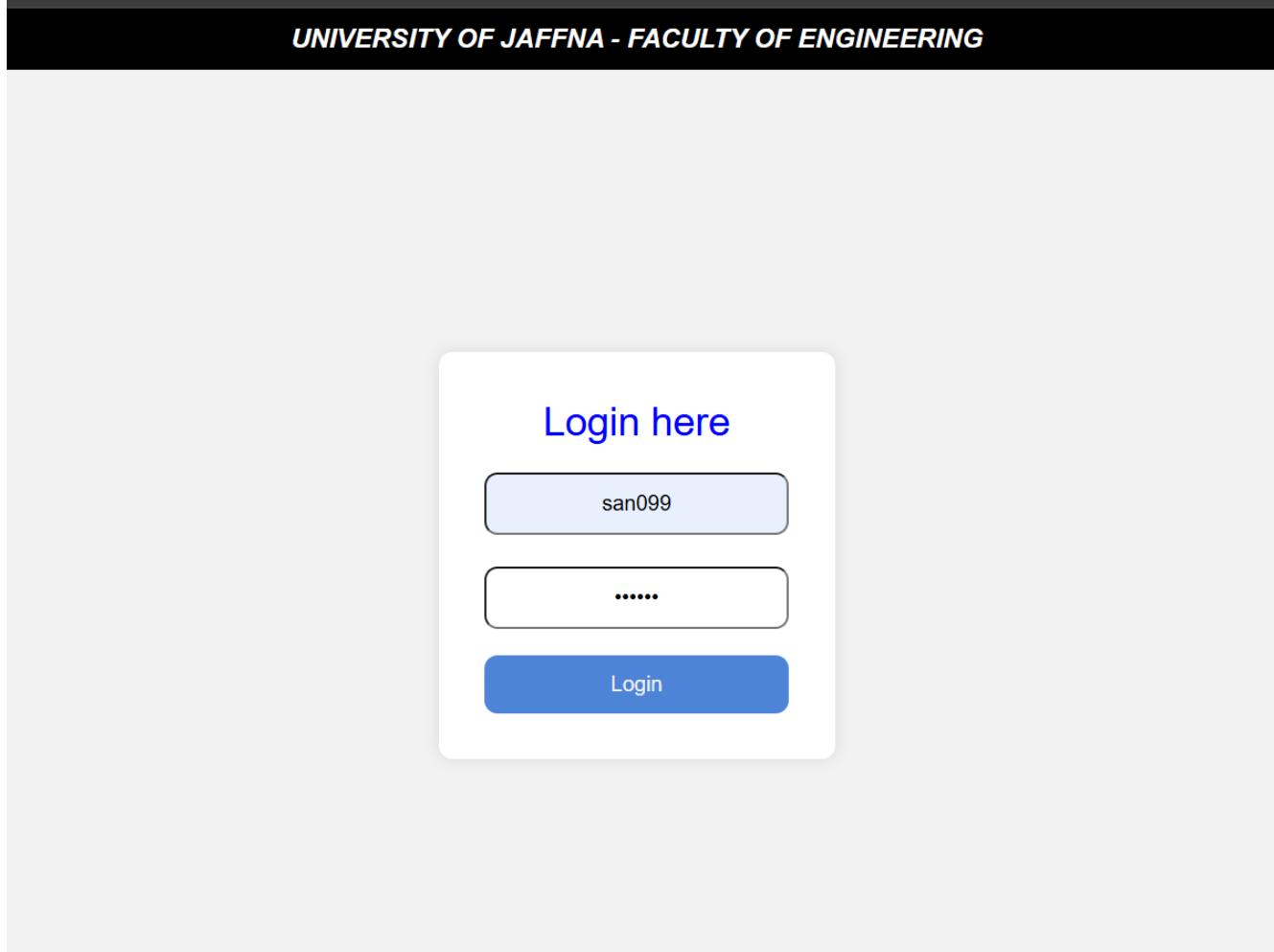
- Prevent double-booking
- Match labs with practical requirements
- Filter students to only access valid labs  
required deep understanding of conditional logic, SQL joins, and real-time validations.

### **5. Generating PDF Reports in PHP**

Creating downloadable lab usage reports using PHP required learning how to use FPDF or similar libraries. Positioning elements correctly on the PDF, formatting tables, and inserting data dynamically from the database involved careful planning and testing

## INTERFACES

### 1. Login interface



## 2. Student interface

### Add course page

UNIVERSITY OF JAFFNA - FACULTY OF ENGINEERING

[View Courses](#) [View Labs](#)

**Course Details**

Course Code	Name	Credits	Hours	Action
No Course added yet				

[Add New Course](#)

### View Labs page

UNIVERSITY OF JAFFNA - FACULTY OF ENGINEERING

[View Courses](#) [View Labs](#)

**Lab Details**

Course Code	Course Name	Lab Name	Laboratory code	Date	Time	Status
EC5110	Computer Architecture	Memory Hierarchy Simulation	COM04	2025-06-29	08:00:00	<span>Attended</span> <span>Not Attended</span>

## 3. Instructor Interface

### Book New Lab page

UNIVERSITY OF JAFFNA - FACULTY OF ENGINEERING

[Book Labs](#) [View Lab Schedule](#)

**Book your Labs**

Course Code	Practical Code	Practical Name	Laboratory Code	Date	Time	Action
No Course added yet						

[Add New Lab](#)

## View Lab Schedule page

UNIVERSITY OF JAFFNA - FACULTY OF ENGINEERING

[Book Labs](#) [View Lab Schedule](#)

### View Lab Schedule

Course Code	Practical Code	Practical Name	Laboratory Code	Date	Time	Action
EC5110	PR014	Memory Hierarchy Simulation	COM04	2025-06-29	08:00:00	<span>Finished</span> <span>Postponed</span> <span>Canceled</span>

## 4. TO Interface

UNIVERSITY OF JAFFNA - FACULTY OF ENGINEERING

### Laboratory Booking Details

Subject ID	Practical ID	Lab ID	Instructor ID	Date	Time
No booking records found					

[Generate PDF](#)

### Laboratory Log

Subject ID	Practical ID	Lab ID	Instructor ID	Date	Time	Action
No lab logs found						

[Generate PDF](#)

## FUNCTION

### 1) User Authentication Function

The User Authentication Function in the Laboratory Booking System is designed to verify the identity of users and determine their access level based on their role. When a user attempts to log in by entering their username and password, the system checks the credentials against the records stored in the users table within the database. If the credentials match an existing user, the system retrieves the associated role — such as Student, Instructor, Lab Technical Officer (TO), or Lecturer In Charge.

Once authenticated, the system initiates a secure session and assigns the user's role and username to session variables. This enables the system to recognize the user throughout their visit and control access accordingly. For example, a student can only view assigned lab schedules, whereas an instructor can request bookings. Similarly, lab technical officers have access to additional feature report generation.

The function ensures that each user is directed to the appropriate dashboard based on their role, preventing unauthorized access to features outside their scope. This role-based access control is crucial for maintaining the integrity, security, and usability of the system.

## 2) Student Page Functions

The **Student Dashboard** in the Laboratory Booking System provides functionalities tailored specifically for students, ensuring they can effectively engage with their assigned practical sessions. One of the key features is the “**Enroll New Course**” function, which allows students to register for available academic courses. This enrolment is essential, as it determines which lab sessions a student is eligible to attend. Upon enrolling, the system maps the student to relevant practical sessions associated with the selected course.

Another important feature is the “**View Scheduled Labs**” function. This allows students to view all upcoming lab sessions linked to the courses they have enrolled in. The system filters the lab schedule to display only the sessions that are relevant to the student's registered courses, thereby ensuring clarity and avoiding confusion. This targeted view enhances the user experience and prevents unauthorized access to labs not intended for the student. Together, these features help maintain a structured and controlled lab access system while giving students clear visibility into their academic responsibilities.

## 3) Instructor Page Functions

The **Instructor Dashboard** in the Laboratory Booking System is designed to facilitate practical session management by enabling instructors to handle lab bookings with ease and accuracy. One of the core features is the “**Book Lab**” function, which includes several automated workflows to streamline the booking process.

When an instructor begins the booking process, they first enter a **course code**. Based on this input, the system automatically retrieves and displays a list of **practical codes** associated with the selected course, eliminating manual errors and ensuring that only valid options are shown. After selecting the relevant practical, the instructor is presented with a list of **available laboratories** that meet two key criteria:

1. The lab must be available during the selected time slot.
2. The lab must belong to the **same department** that offers the selected course.

Once the instructor selects a laboratory, the system automatically displays the **equipment and instruments** available in that lab, helping the instructor confirm whether the lab is suitable for the practical session. The instructor can then choose a **booking date**, and the system checks the lab's availability in real-time. If the slot is free, the system **automatically books the lab** and stores the details in the database, updating the lab schedule accordingly.

In addition to booking, the instructor has access to a “**View Scheduled Labs**” page. This feature displays all previously booked sessions by the instructor. From this page, the instructor can manage session statuses by marking them as **Finished**, **Canceled**, or **Postponed**, ensuring that lab schedules remain up-to-date and accurate.

This comprehensive automation not only reduces the instructor’s workload but also ensures accurate lab usage, prevents booking conflicts, and enhances overall system efficiency.

## **Lab Technical Officer (TO) Page Functions**

The Lab **Technical Officer (TO)** Dashboard in the Laboratory Booking System provides essential tools for managing and monitoring lab activities under their responsibility. Upon logging in, each TO is granted access only to the **laboratories they are officially assigned to**, ensuring focused and role-specific interaction with the system.

A key function of the TO dashboard is the ability to **view the complete lab schedule** for their designated laboratories. This includes upcoming bookings, practical sessions, time slots, and the instructors who booked them. In addition to schedule visibility, TOs can also access log details that **record all historical activities** and equipment usage associated with each lab. These logs provide transparency and allow the TO to track how often and by whom the lab has been used.

To support reporting and documentation, the system includes an integrated feature that allows TOs to **download both the lab schedule and the log data as PDF files**. This functionality helps in maintaining physical or digital records, preparing reports for academic review, or ensuring compliance with equipment maintenance and usage policies.

Overall, the TO functions support efficient lab supervision, promote accountability, and ensure that labs are prepared and managed effectively for each scheduled session.

## **GITHUB link**

[Git hub Repository Link](#)