SOFTWARE REQUIREMEMENT SPECIFICATION

NAME	DHARANI M
ROLL NUMBER	7376221CS141
SEAT NUMBER	45
PROJECT ID	5
PROBLEM STATEMENT	RESOURCE BOOKING SYSTEM

PROBLEM STATEMENT:

- **Resource Overload:** With different departments managing bookings independently, there is a risk of resource overloading, resulting in conflicts and inefficiencies in resource utilization.
- **Booking Duplication:** Lack of centralized oversight leads to duplicate bookings for the same facility, causing confusion and wasting valuable resources.
- **Inefficient Communication:** Important booking-related communications are dispersed across various channels, making it difficult for users to track and manage their bookings effectively.
- Administrative Burden: Administrators face the challenge of manually coordinating bookings, resolving conflicts, and ensuring fair allocation of resources, which is timeconsuming and prone to errors.
- Lack of Transparency: Users may struggle to access real-time availability information, leading to frustration and delays in booking processes.

PROJECT FLOW:

PURPOSE:

The purpose of the Resource Booking System is to streamline the reservation process for facilities like classrooms, labs, auditoriums, and seminar halls. It enables users to effortlessly check availability, submit booking requests, and receive prompt confirmation notifications after the approval of admins. By providing a user-friendly platform for managing bookings, it optimizes resource utilization and enhances organizational efficiency.

SCOPE OF THE PROJECT:

The scope of the project is to develop a resource booking system that allows faculty members to reserve facilities such as classrooms, labs, auditoriums, and seminar halls. The system will include user authentication, a booking request form, conflict checking, and a real-time dashboard for managing schedules. It will integrate with existing email systems to send notification emails for booking approvals and rejections. The system will provide admins with the ability to review and approve booking requests, and generate reports for usage statistics.

CONSIDERATION:

- All users possess Google account for authentication
- Ensuring compliance with data protection regulations and implementing robust security measures to safeguard users' personal information and booking data.

USER PERSONAS:

- Admin: Manages system operations ,resolves conflicts ,approves or reject users request and sent the confirmation status through mail
- **Faculty:** Checks the resource availability and book the venue and time. After booking he will get the confirmation status through mail

FUNCTIONAL REQUIREMENTS:

• User Authentication:

The system should provide user authentication for faculty and admin staff. Faculty and admin staff should have separate login credentials. User authentication should be performed using a username and password.

• Venue and Schedule Management:

Faculty should be able to view available venues (e.g., classrooms, labs, auditoriums) and their corresponding schedules. Faculty should be able to check the

availability of venues for specific dates and times. Admin staff should have access to a comprehensive view of all venues and their schedules.

• Booking Requests:

Faculty should be able to submit booking requests for desired venues, dates, and times. Faculty should receive confirmation emails upon successful submission of their booking requests.

• Conflict Resolution:

The system should perform conflict checks to prevent double bookings or overlapping reservations. It should check if the selected venue is already booked for the requested time slot. If a conflict is detected, the system should notify the faculty and suggest alternative venues or time slots.

• Approval Workflow:

Admin staff should have the ability to review and approve/reject booking requests. Admin staff should have access to a dashboard displaying pending booking requests. Admin staff should be able to send notifications to faculty regarding the status of their booking requests (approved or rejected).

• Schedule Updates and Notices:

Faculty should have the ability to send schedule updates and notices to students or colleagues. The system should provide a communication feature (e.g., email integration) for faculty to send notifications efficientl.

• Reporting and Analytics:

The system should provide reporting capabilities to generate usage statistics, such as the number of bookings per venue, department, or time period. Analytics should be available to identify popular venues, frequently booked time slots, and any booking conflicts that require attention.

NON-FUNCTIONAL REQUIREMENTS:

1.Performance:

- The system should provide fast response times, with a maximum page load time of 2 seconds.
- The system should be able to handle concurrent requests from at least 500 users without significant performance degradation.

2.Scalability:

- The system should be able to handle a 50% increase in user load within a 6-month period without requiring major infrastructure changes.
- The system should support the addition of new venues and resources without impacting performance.

3. Reliability:

- The system should have an uptime of at least 99.9%.
- It should be able to recover from system failures within 5 minutes without data loss.

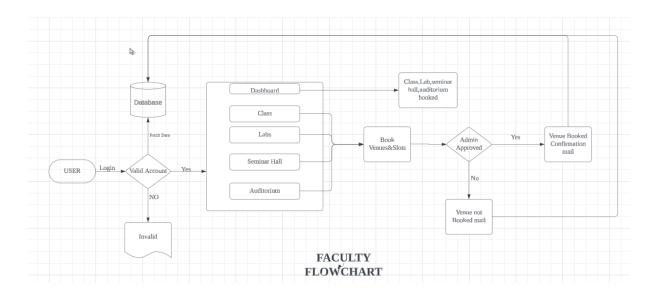
4. Security:

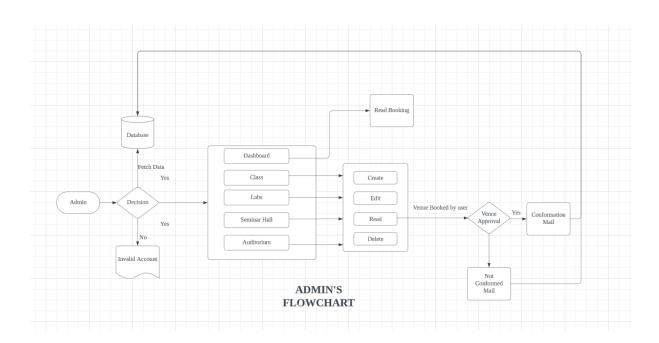
- User passwords should be stored securely using encryption algorithms.
- The system should implement role-based access control to ensure that only authorized users can perform specific actions.

5.Usability:

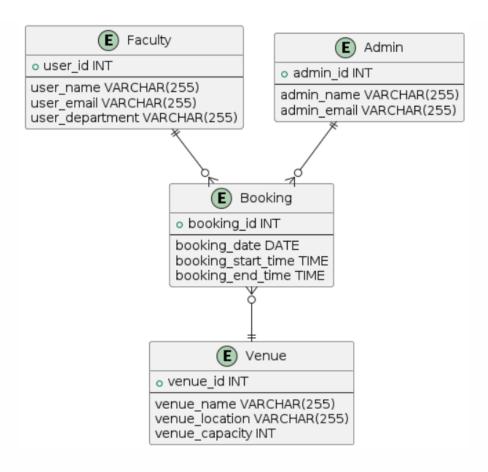
- The system should have a responsive and intuitive user interface that works well on different devices and screen sizes.
- It should support accessibility standards, including keyboard navigation and screen reader compatibility.

FLOW CHART:





ER-DIAGRAM:

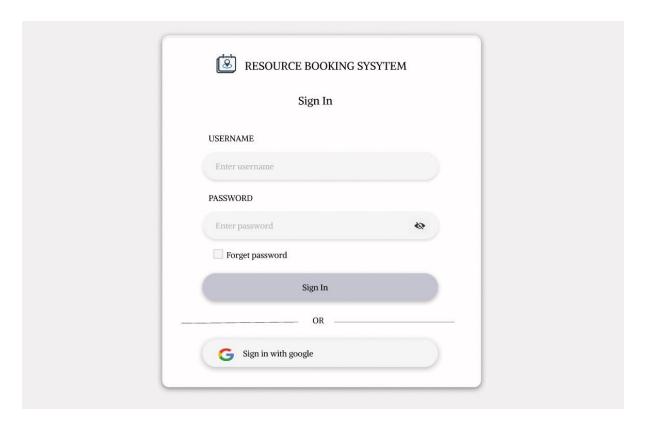


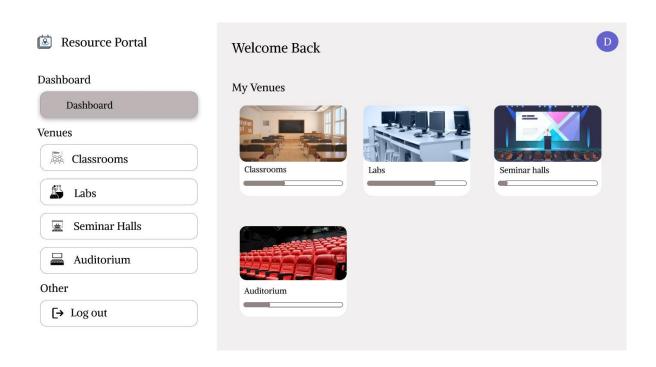
STACK:

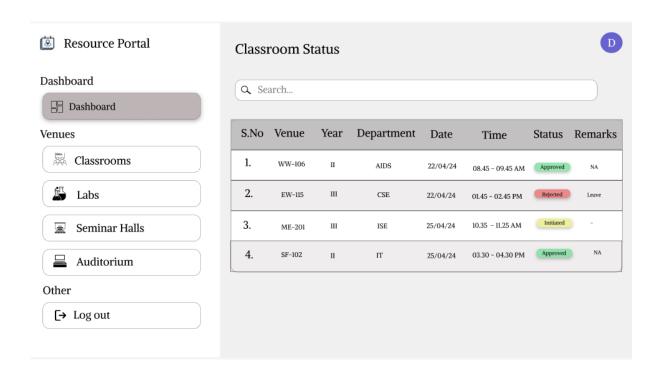
FRONTEND	• HTML
	• CSS
	• JS
BACKEND	
	• Python
	• Django(Python Web)
DATABASE	PostgreSQL
	• MySQL

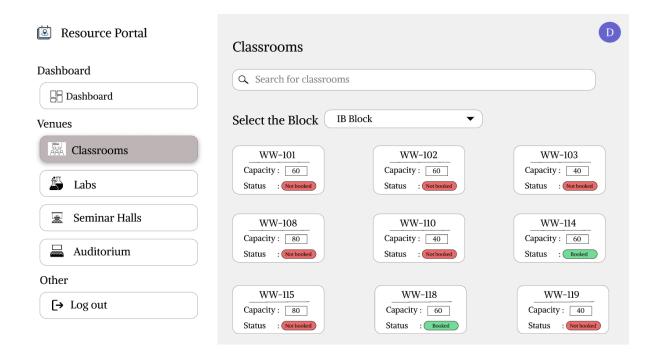
PROTOTYPE:

1.FACULTY VIEW:

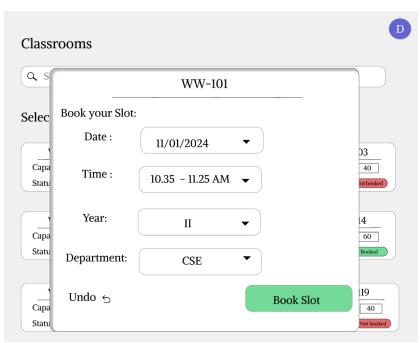




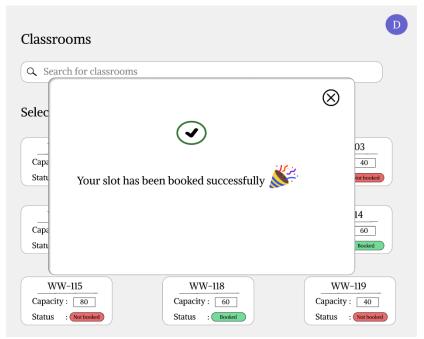












2.ADMIN'S VIEW:

