
Requirements and Design Document

for

PanelHub

Version 1.0

Prepared by Team 8:

Austin George	M250288CS	austin_m250288cs@nitc.ac.in
Rishabh Gadpal	M250981CS	rishabh_m250981cs@nitc.ac.in
Yendluri Pravalika	M251129CS	yendluri_m251129cs@nitc.ac.in

Project Manager: Tilok Doley

Project Client: National Institute Of Technology Calicut

Course: CS6103E - Software Systems lab

Date: 27 Oct 2025

Glossary

System	The PanelHub application
API	Application programming Interface, from data is received or sent to backend service.
Scholar	Students of NITC
Faculty	Faculty of NITC
F#	Functional requirement
UI	User Interface - the frontend of the application where users will interact with the application.

Table of contents

Glossary	2
Table of contents	3
MERN Application Design Document: PanelHub	4
1. Introduction	4
Project Vision	4
Problem Statement	4
Target Audience	4
Key Objectives	4
2. Functional Requirements	5
3. UI Design	7
3.1 Scholar Registration Page	7
3.2 Faculty Registration Page	7
3.3 Login Screen	8
3.4 Manage Free Slot for faculty	8
3.5 View Confirmed Booking for faculty	9
3.6 Manage Approved Booking for faculty	10
3.7 View Unapproved Booking for faculty	10
3.8 Book Slot by scholar	11
3.9 Manage Booking by scholar	12
4. System Architecture	13
4.1. High-Level Diagram	13
4.2. Technology Stack	13
4.3. Database Schema (MongoDB Collections)	14
4.3.1 Role Collection	14
4.3.2 Department Collection	14
4.3.3 Course Category Collection	14
4.3.5 Booking Collection	15
4.3.6 Faculty Free Slots Collection	15
4.3.7 Notification Collection	16

MERN Application Design Document: PanelHub

1. Introduction

Project Vision

PanelHub is a web-based scheduling platform that simplifies and automates the process of booking and managing project evaluation slots in academic institutions.

Problem Statement

Academic scheduling for project evaluations is often inefficient due to manual coordination, lack of real-time visibility of faculty availability, and overlapping bookings. This leads to scheduling conflicts, wasted time, and frustration for both students and faculties. PanelHub addresses these issues by providing a centralized, automated system to streamline scheduling and ensure transparency.

Target Audience

The primary users of PanelHub are:

- **Faculty:** Responsible for adding and managing faculty availability, approving scholar requests for booking slots.
- **Scholars (Students/Researchers):** Undergraduate, postgraduate, and doctoral students who need to view, book, and manage evaluation slots with their respective faculty panels.

Key Objectives

The key objectives of PanelHub are to:

- Provide an intuitive platform for slot booking and cancellation.
- Reduce scheduling conflicts through real-time updates and conflict prevention.
- Improve transparency by allowing scholars to view available slots.
- Enhance efficiency for administrators by centralizing faculty and venue management.

2. Functional Requirements

Key Features & Scope: A high-level list of the main features that will be built.

F1: Scholar should be able to create an account in the system using name, valid NITC email ID, password, department (CSE, ME, ECE, EEE, CE), course category (UG/PG/PhD).

F2: Faculty should be able to create an account in the system using name, valid NITC email ID, password, department.

F3: Scholar and Faculty should be able to login into the system.

F4: The system should be able to do role based authorization and direct them to role specific dashboards. (Scholar Role, Faculty Role).

F5: In the faculty dashboard, the system shall display a table like calendar up to 1 week ahead where the faculty can add/delete the free time for each course category. The calendar will be a table like structure where dates are the rows of the calendar and time as the columns of 1Hr periods. The time is limited between 8:00 AM to 8:00 PM. No free slots can be added on Sundays.

F6: In the Faculty dashboard, the system should display the confirmed status bookings in the order of date and time.

F7: In the Faculty dashboard, the system shall display to the faculty the booking requests approved by the faculty which are in pending status (Partial approval - waiting for other faculties approval).

F8: In the Faculty dashboard, the system shall display the booking requests which are yet to be approved by him/her.

F9: The system shall delete the bookings requests from scholars and other faculties of the panel when a faculty rejects the request for booking.

F10: The system shall provide the filtering functionality to the faculty to filter based date, time and course category.

F11: A faculty may cancel a confirmed booking from their dashboard. This action cancels the booking for all participants and notifies the scholar and other panel members.

F12: In the Scholar Dashboard, Scholar should be able to find a suitable slot for presentation by giving the details like list of his/her department faculties (up to 3 faculties in the same department of the Scholar) required in the panel and date. Each slot is of 1Hr duration.

F13: When the scholar applies a search option for a slot, the system shall display the all slots corresponding to his/her department where all the panel faculties have common free time on that specific date. If no slots are available, the Scholar has to select some other date for getting the slot.

F14: The system shall allow the Scholar to book the slot. One active booking is allowed.

F15: System should display status of the booking whether pending (waiting for approval from each panel faculties) or confirmed to the Scholar. The status turns from pending to confirmed when all the faculties in the list approves.

F16: The system shall allow the Scholar to cancel the confirmed/pending booking.

F17: The system shall send notification to Scholar and panel faculties regarding every state change in the booking.

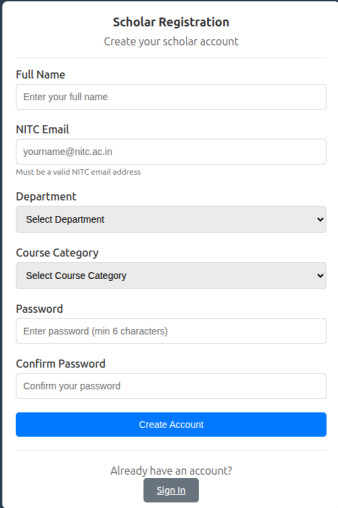
F18: The system shall provide a logout option for Scholar and Faculty.

3. UI Design

Below are the sample UI designs which will almost resemble the final product which will be developed. Dummy data is given in the UI for the purpose of presentation of the UI.

3.1 Scholar Registration Page

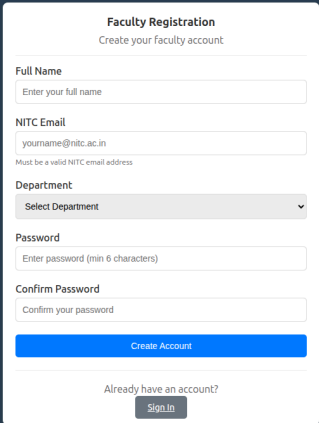
This page is for a scholar to create an account in the system.



The Scholar Registration form is centered on a dark blue background. It has a title 'Scholar Registration' and a subtitle 'Create your scholar account'. The form contains the following fields: 'Full Name' with a placeholder 'Enter your full name'; 'NITC Email' with a placeholder 'yourname@nitc.ac.in' and a note 'Must be a valid NITC email address'; 'Department' and 'Course Category' as dropdown menus with 'Select Department' and 'Select Course Category' respectively; 'Password' with a placeholder 'Enter password (min 6 characters)'; and 'Confirm Password' with a placeholder 'Confirm your password'. A blue 'Create Account' button is at the bottom of the form. Below the button, there is a link 'Already have an account?' and a 'Sign In' button.

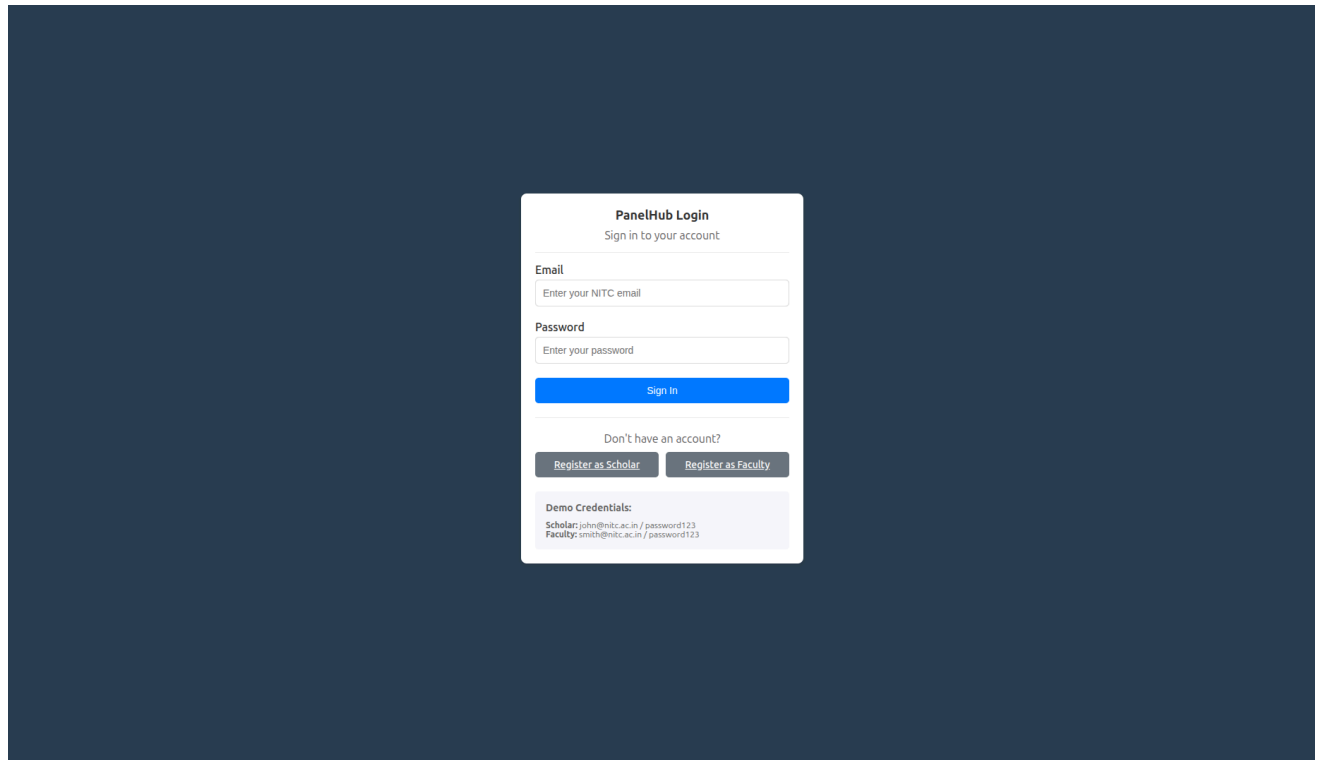
3.2 Faculty Registration Page

This page is for a faculty to create account the in system



The Faculty Registration form is centered on a dark blue background. It has a title 'Faculty Registration' and a subtitle 'Create your faculty account'. The form contains the following fields: 'Full Name' with a placeholder 'Enter your full name'; 'NITC Email' with a placeholder 'yourname@nitc.ac.in' and a note 'Must be a valid NITC email address'; 'Department' as a dropdown menu with 'Select Department'; 'Password' with a placeholder 'Enter password (min 6 characters)'; and 'Confirm Password' with a placeholder 'Confirm your password'. A blue 'Create Account' button is at the bottom of the form. Below the button, there is a link 'Already have an account?' and a 'Sign In' button.

3.3 Login Screen



3.4 Manage Free Slot for faculty

Faculty can add his/her free slots in this table.

PanelHub

Manage Free Slots

View Confirmed Booking

Manage Approved Booking

View Unapproved Booking

Welcome, Dr. Smith (Faculty)Logout

Manage Free Slots

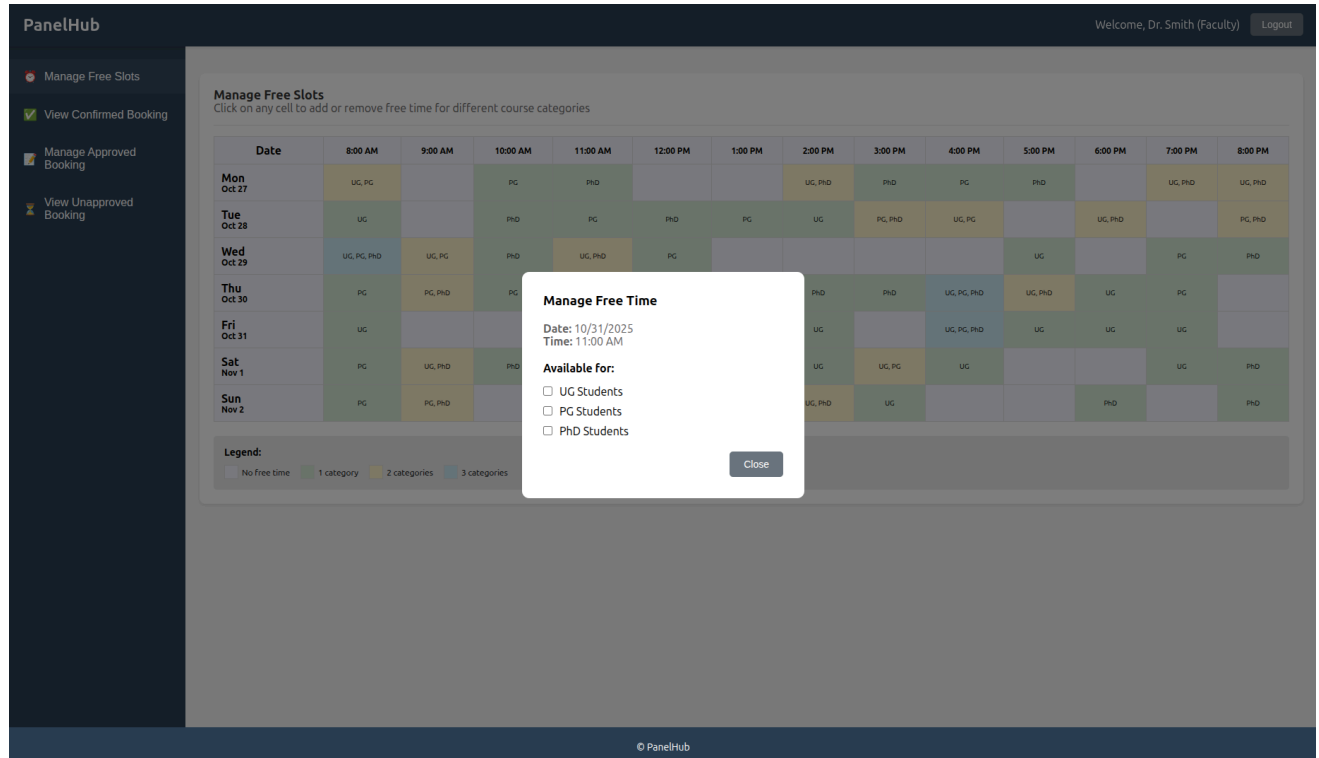
Click on any cell to add or remove free time for different course categories

Date	8:00 AM	9:00 AM	10:00 AM	11:00 AM	12:00 PM	1:00 PM	2:00 PM	3:00 PM	4:00 PM	5:00 PM	6:00 PM	7:00 PM	8:00 PM
Mon Oct 27	UG, PG		PG	PHD			UG, PHD	PHD	PG	PHD		UG, PHD	UG, PHD
Tue Oct 28	UG		PHD	PG	PHD	PG	UG	PG, PHD	UG, PG		UG, PHD		PG, PHD
Wed Oct 29	UG, PG, PHD	UG, PG	PHD	UG, PHD	PG					UG		PG	PHD
Thu Oct 30	PG	PG, PHD	PG	PHD	PG		PHD	PHD	UG, PG, PHD	UG, PHD	UG	PG	
Fri Oct 31	UG				UG	UG, PG	UG		UG, PG, PHD	UG	UG	UG	
Sat Nov 1	PG	UG, PHD	PHD		UG, PG, PHD	UG	UG	UG, PG	UG			UG	PHD
Sun Nov 2	PG	PG, PHD			PG		UG, PHD	UG			PHD		PHD

Legend:

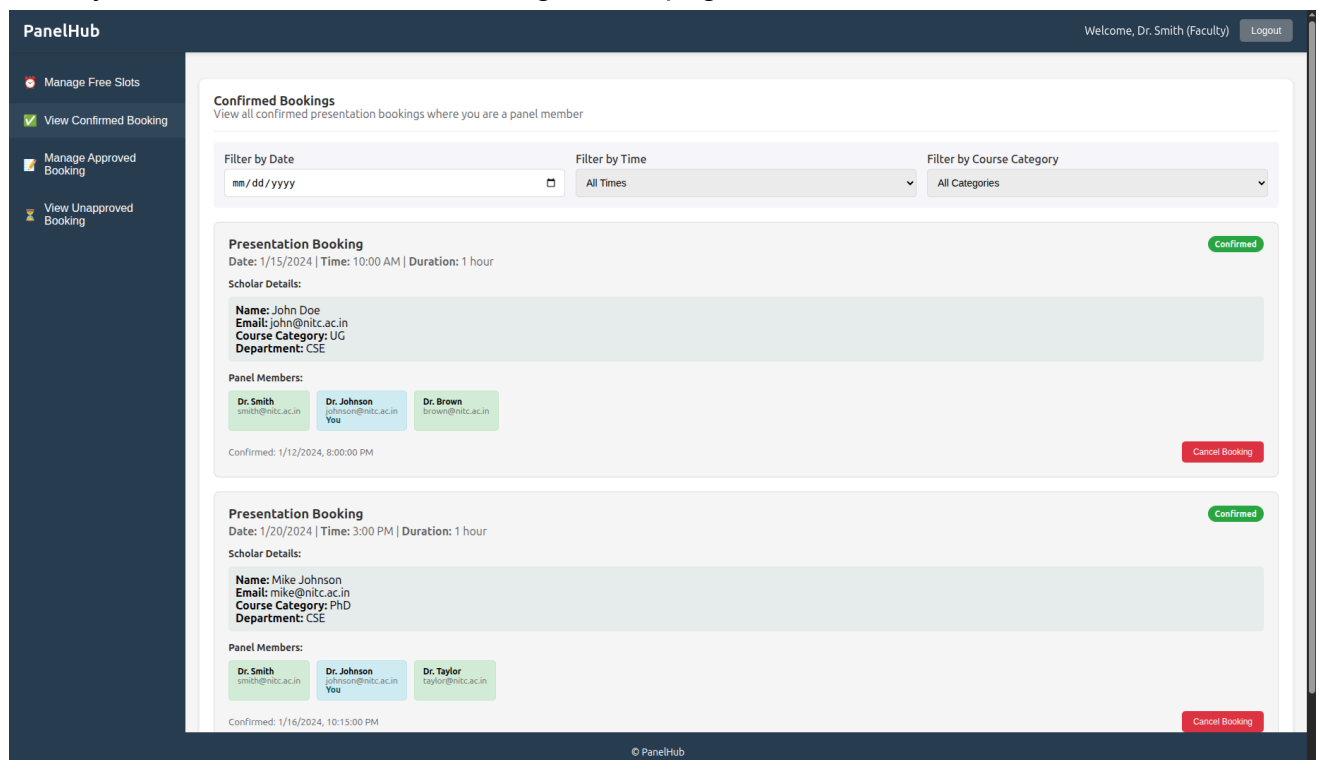
No free time1 category2 categories3 categories

© PanelHub



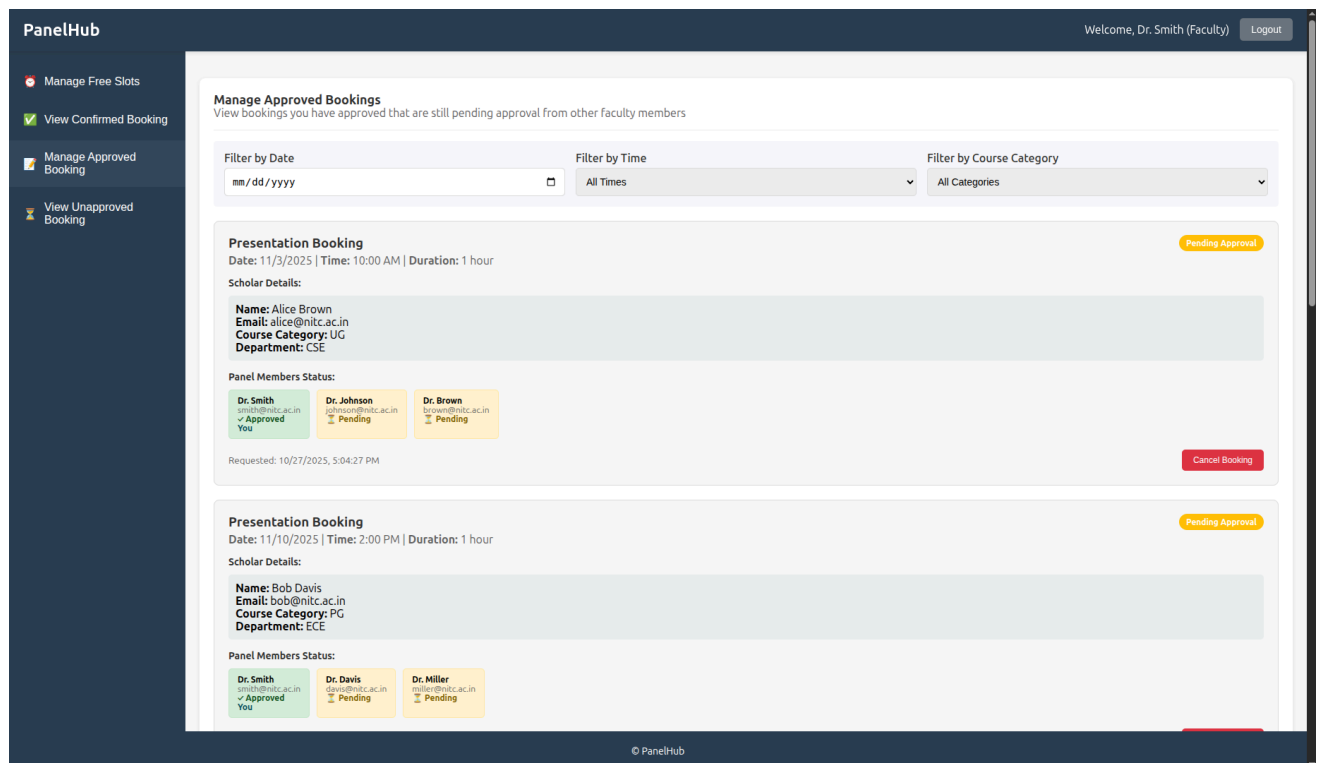
3.5 View Confirmed Booking for faculty

Faculty can see the confirmed booking on this page.



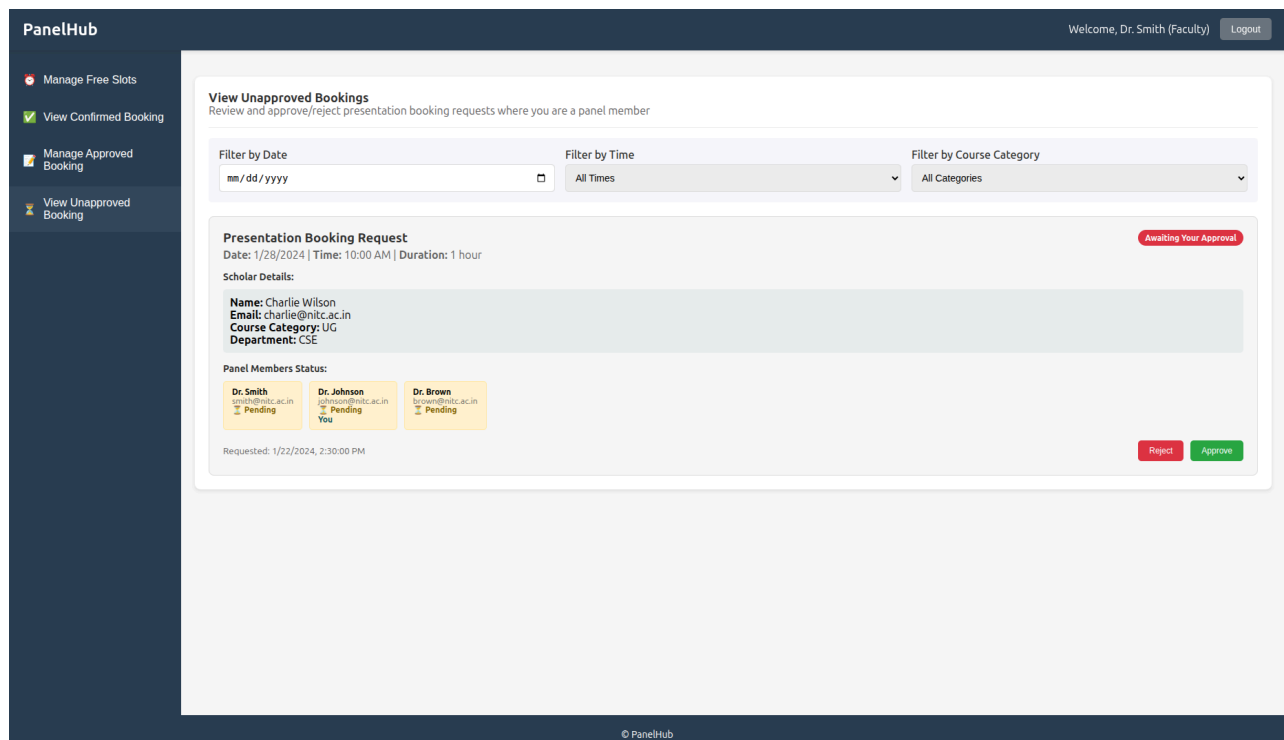
3.6 Manage Approved Booking for faculty

Faculty can manage the partially approved bookings.



3.7 View Unapproved Booking for faculty

Faculty can approve/reject the booking request from a scholar.



3.8 Book Slot by scholar

Scholars can search the free slots available for the faculties in the required in the panel.

PanelHub

Welcome, John Doe (Scholar)

Logout

Book Slot

Manage Booking

Book Presentation Slot

Select faculty members and find available slots for your presentation

Select Faculty Members (Max 3)

Select up to 3 faculty members

Selected: 0/3

Preferred Date

mm/dd/yyyy

Search Available Slots

© PanelHub

Available Slots

Select a slot to book your presentation

11:00 AM

Duration: 1 hour

Book This Slot

11:00 AM

Duration: 1 hour

Book This Slot

3:00 PM

Duration: 1 hour

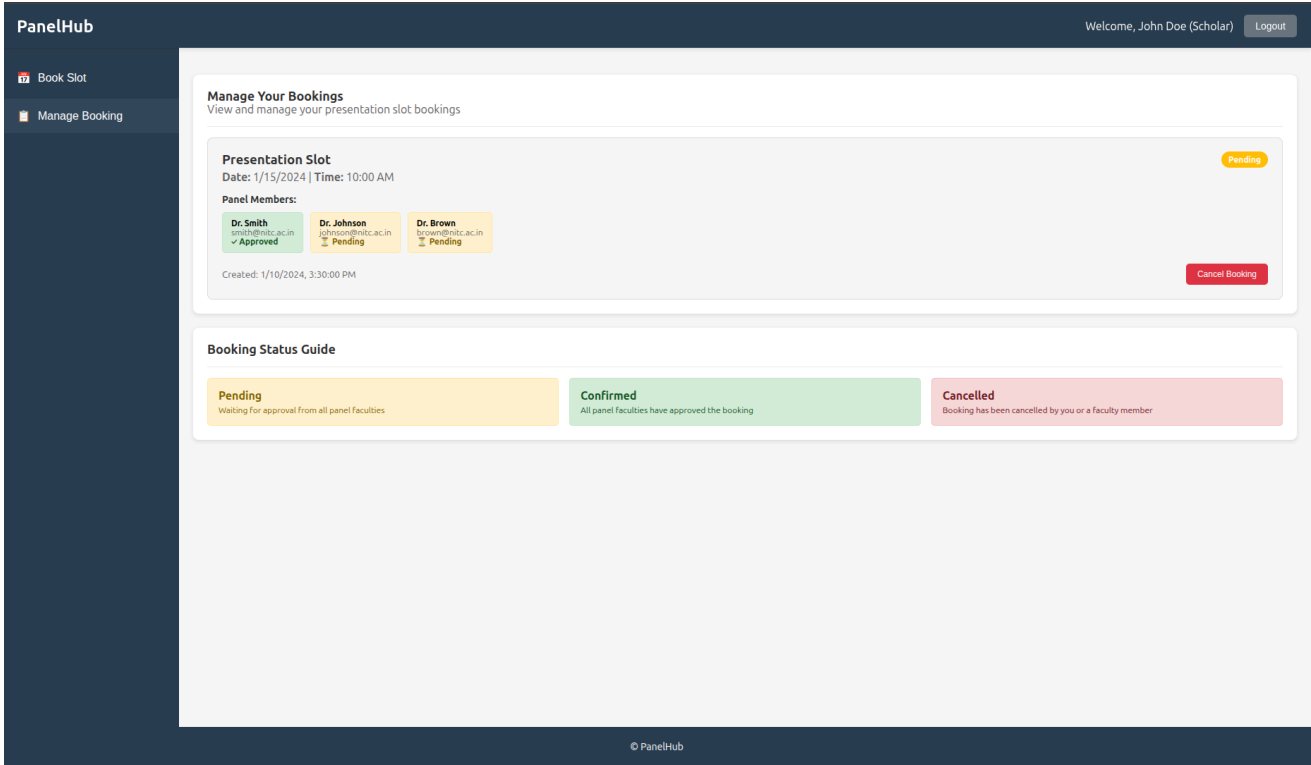
Book This Slot

Selected Panel:

- Dr. Lee
- Dr. Johnson
- Dr. Taylor

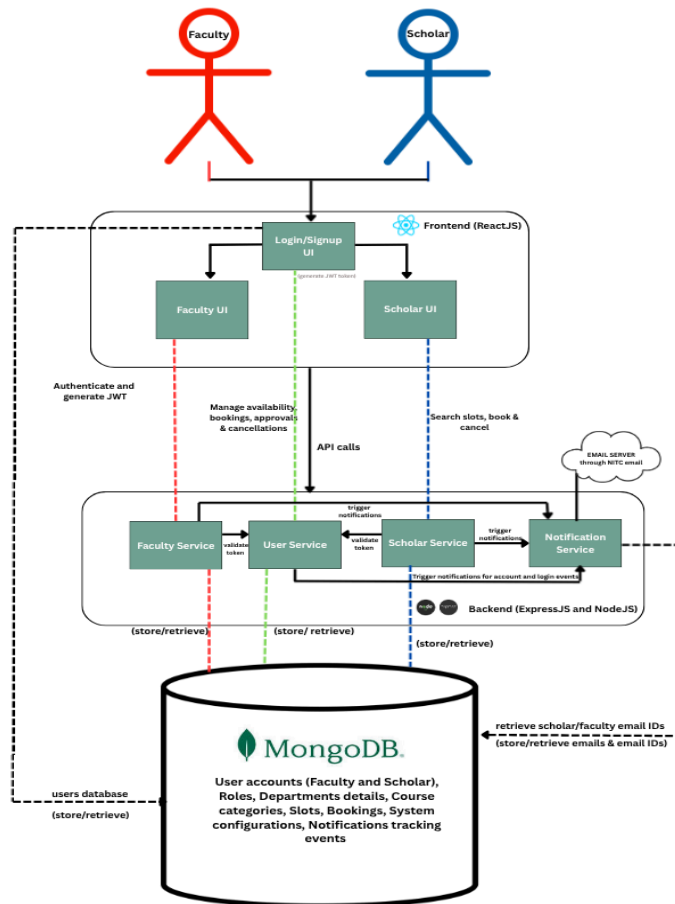
© PanelHub

Scholars can manage the booking from this page.



4. System Architecture

4.1. High-Level Diagram



4.2. Technology Stack

- **Frontend:** React 18.x, React Router 6.x, Axios, Tailwind CSS
- **Backend:** Node.js 20.x, Express 4.x, Mongoose 8.x
- **Database:** MongoDB 7.x
- **Authentication:** JSON Web Tokens (JWT)

4.3. Database Schema (MongoDB Collections)

4.3.1 Role Collection

Stores the type of users the system will have.

```
{
  _id: ObjectId,
  name: String
}
```

4.3.2 Department Collection

Keeps track of departments

```
{
  _id: ObjectId,
  name: String
}
```

4.3.3 Course Category Collection

Keep track of the course category.

```
{
  _id: ObjectId,
  name: string
}
```

4.3.4 Users Collection

Holds data for both Scholars and Admins.

```
{
  _id: ObjectId,
  name: String,
  email: String,           // must be NITC email
  password: String,        // hashed
  role: ObjectId,          // reference to Role collection
  department: ObjectId,    // reference to department collection
  courseCategory: ObjectId, // reference to course category collection
  createdAt: timestamp
}
```

4.3.5 Booking Collection

Core collection for venue slots.

```
{
  _id: ObjectId,
  scholarIds:                // reference to Scholar
  facultyApprovals: {
    Faculty1: {
      facultyId:
      approveStatus:
    }
    Faculty2: {
      facultyId:
      approveStatus:
    }
    Faculty3: {
      facultyId:
      approveStatus:
    }
  }
  status: String,            // "pending", "booked"
  date: Date,
  startTime: String,        // "10:00"
  duration: Number,         // always 1 hr
  department: ObjectId,     // reference to department
  courseCategory: ObjectId, // reference to course category
  createdBy: ObjectId,      // reference to Users (Scholar)
  createdAt: timestamp,
  updatedBy: ObjectId,
  updatedAt: timestamp,
}
```

4.3.6 Faculty Free Slots Collection

Collection which stores the booked slot details.

```

{
  _id: ObjectId,
  facultyId: ObjectId,          // reference to Faculty
  freeSlot: {
    dateToday: {
      08-09:
      09-10:
      10-11:
      11-12:
      12-13:
      13-14:
      14-15:
      15-16:
      16-17:
      17-18:
      18-19:
      19-20:          //All combination of UG/PG/PhD
    },
    dateToday: {
    },
    dateToday: {
    },
    dateToday: {
    },
    dateToday: {
    },
    dateToday: {
    },
    dateToday: {
    },
    dateToday: {
    },
  },
  updatedAt: timestamp
}

```

4.3.7 Notification Collection

To track emails sent from system events.

```

{
  _id: ObjectId,
  recipient: {
    scholarIds: ObjectId[], // reference to user ID
    facultyIds: ObjectId[]  // reference to faculty ID
  },
  // recipient
  event: String,           // booking Created/Cancelled, slot Updated/Deleted
  slotId: ObjectId,
}

```



```
    createdAt: timestamp,  
    sentStatus: boolean  
}
```