

Role-Based Smart Room Booking And Management System For Academic Institutions Using Azure Services

CS19741 Cloud Computing

Submitted by

AGISHRAJ R (2116220701016)

CAROLINE SUJA J S (2116220701048)

CHANDRU S (2116220701051)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING



RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

MAY 2025

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

BONAFIDE CERTIFICATE

Certified that this Project titled “ **Role-Based Smart Room Booking And Management System For Academic Institutions Using Azure Services** ” is the bonafide work of “**AGISHRAJ R (2116220701016), CAROLINE SUJA J S (2116220701048), CHANDRU S (2116220701051)**” who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. E.M. Malathy

HEAD OF THE DEPARTMENT

Professor

Department of Computer Science
and Engineering,
Rajalakshmi Engineering College,
Chennai - 602 105.

SIGNATURE

Dr. M. Sandhiya

SUPERVISOR

Assistant Professor

Department of Computer Science
and Engineering,
Rajalakshmi Engineering
College, Chennai-602 105.

Submitted to Mini Project Viva-Voce Examination held on _____

Internal Examiner

External Examiner

ABSTRACT

" Role-Based Smart Room Booking And Management System For Academic Institutions Using Azure Services " is a platform through which advance educational institutions gain an effective tool to schedule meetings and manage meeting spaces. This system implements Flutter in its frontend section and utilizes Firebase as its backend framework to offer convenience in booking services across all user types including students and faculty members with administrator privileges. Users can access three primary system capabilities through its authorization platform and time-dependent space availability tracking and room attribute viewing including real-time space position and capacity limits. The system allows users to make booking requests whereas administrators receive requests for review together with capabilities to handle room lists and study booking records. Through this system organizations achieve higher resource utilization along with reduced administrative workloads and boosted transparency in the booking procedure. From their profile screen users have access to view their current active bookings including status information and their history of previous reservations. Along with modern cloud technology deployment the platform maintains secure stable system management while delivering real-time data synchronization functions. The solution supports operational enhancement through its problem-solving scheduling abilities as well as its effective academic facility management. The "Smart Room Booking and Management System for Academic Institutions reaches its academic resource allocation goals by blending user-friendly interfaces with robust backend structures for providing dependable secure digital transformation solutions."

ACKNOWLEDGMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavor to put forth this report. Our sincere thanks to our Chairman **Mr. S. MEGANATHAN, B.E, F.I.E.**, our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.**, and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN, Ph.D.**, for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.**, our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to **Dr. E.M.MALATHY** Professor and Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guides **Dr. M. SANTHIYA** .We are very glad to thank our Project Coordinator, **Dr. M. SANTHIYA** Assistant Professor Department of Computer Science and Engineering for his useful tips during our review to build our project.

AGISHRAJ R **2116220701016**

CAROLINE SUJA J S **2116220701048**

CHANDRU S **2116220701051**

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	iii
	ACKNOWLEDGMENT	iv
	LIST OF TABLES	vii
	LIST OF FIGURES	viii
	LIST OF ABBREVIATIONS	ix
1.	INTRODUCTION	1
	1.1 GENERAL	1
	1.2 OBJECTIVES	2
	1.3 EXISTING SYSTEM	2
2.	LITERATURE SURVEY	3
3.	PROPOSED SYSTEM	6
	3.1 GENERAL	6
	3.2 SYSTEM ARCHITECTURE DIAGRAM	6
	3.3 DEVELOPMENT ENVIRONMENT	8
	3.3.1 HARDWARE REQUIREMENTS	8
	3.3.2 SOFTWARE REQUIREMENTS	8
	3.4 DESIGN THE ENTIRE SYSTEM	9
	3.4.1 ACTIVITYY DIAGRAM	9
	3.4.2 DATA FLOW DIAGRAM	11
	3.5 STATISTICAL ANALYSIS	13

4.	MODULE DESCRIPTION	15
	4.1 SYSTEM ARCHITECTURE	15
	4.1.1 USER INTERFACE DESIGN	15
	4.1.2 BACK END INFRASTRUCTURE	16
	4.2 ROOM MANAGEMENT WORKFLOW	16
	4.2.1 ROOM INFORMATION MANAGEMENT	16
	4.2.2 USER AUTHENTICATION AND ROLE SELECTION	16
	4.2.3 SEARCH AND BOOKING REQUESTS	16
	4.2.4 BOOKING VERIFICATION AND APPROVAL	18
	4.2.5 BOOKING HISTORY AND RECORD MAINTENANCE	18
	4.3 SYSTEM WORKFLOW	18
	4.3.1 USER INTERACTION	18
	4.3.2 ROOM SEARCH AND REQUEST SUBMISSION	18
	4.3.3 ADMIN REVIEW AND APPROVAL	18
	4.3.4 CONTINUOUS SYSTEM UPDATE	19
5.	IMPLEMENTATIONS AND RESULTS	20
	5.1 IMPLEMENTATION	20
	5.2 OUTPUT SCREENSHOTS	20

6.	CONCLUSION AND FUTURE ENHANCEMENT	24
	6.1 CONCLUSION	24
	6.2 FUTURE ENHANCEMENT	24
	REFERENCES	25

LIST OF TABLES

TABLE NO	TITLE	PAGE NO
3.1	HARDWARE REQUIREMENTS	8
3.2	SOFTWARE REQUIREMENTS	8
3.3	COMPARISON OF FEATURES	13

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
3.1	SYSTEM ARCHITECTURE	7
3.2	ACTIVITY DIAGRAM	10
3.3	DFD DIAGRAM	12
3.4	COMPARISON GRAPH	13
4.1	SEQUENCE DIAGRAM	16
5.1	LOGIN AND REGISTER SCREENS	21
5.2	USER DASHBOARD	21
5.3	BOOK ROOM AND VIEW AVAILABLE ROOMS	22
5.4	ENTER PURPOSE AND SEND REQUEST	22
5.5	USER PROFILE	22
5.6	CURRENT BOOKINGS AND PREVIOUS BOOKINGS	22
5.7	ADMIN DASHBOARD	23
5.8	LIST OF USERS AND ROOMS	23
5.9	ADD ROOM	23
5.10	REQUESTS AND BOOKING HISTORY	23

LIST OF ABBREVIATIONS

S. No	ABBR	Expansion
1	API	Application Programming Interface
2`	DB	Database
3	GUI	Graphical user interface
4	CRUD	Create, read, update, delete
5	SQL	Structured query language
6	UI	User interface
7	UX	User experience
8	DFD	Data Flow Diagram
9	ID	Identification
10	OTP	One-time password

CHAPTER 1

INTRODUCTION

1.1 GENERAL

"Smart Room Booking and Management System for Academic Institutions" brings modern optimization to the college scheduling and space management operations for committed rooms. The platform delivers productive and secure room booking operations for all users through present-day technological capabilities. The development platform uses Flutter for frontend components combined with Firebase for real-time backend management through synchronized data functions.

Through the platform users can select their roles for login followed by the selection of preferred time slots to view available rooms and file booking requests with defined meeting purposes. Users can easily choose their rooms by seeing critical details including number, block location, floor level, seating capacity and present amenities on display.

The system allows administrators full control over room management duties while they examine booking proposition requests and decide on their approvals or rejections to maintain scheduled disposal without conflicts. The system offers a profile page that lets users both check their booking standing and examine their previous bookings.

Through room utilization improvements and automated process automation and transparent information handling "Smart Room Booking and Management System for Academic Institutions" constructs modernized booking systems which boost educational facility operational effectiveness and digital innovation potential.

1.2 OBJECTIVE

The objective of "*Smart Room Booking and Management System for Academic Institutions*" is to build an intelligent real-time platform which simplifies room booking and management operations at colleges. A user-friendly interface enhances the reservation management functions that students along with faculty and administrators can access without difficulty. The system operates with smooth functionality through its integration of Flutter and Firebase technologies which provides both data security and instant database synchronizing capabilities for all users.

There are four main objectives to achieve through the platform which combine conflict avoidance with space optimization while offering clear scheduling visibility and easier administrative procedures. Students and administrators can access detailed information about meeting spaces while tracking booking statuses using role-based credentials which create an efficient highly organized work environment. Its mission is to upgrade the conventional space booking system by digital transformation so academic institutions achieve better reliability alongside transparent and accessible processes.

1.3 EXISTING SYSTEM

The college meeting room reserving system brings modern optimization to the college scheduling and space management operations for committed rooms. The platform delivers productive and secure room booking operations for all users through present-day technological capabilities. The development platform uses Flutter for frontend components combined with Firebase for real-time backend management through synchronized data functions.

Through the platform users can select their roles for login followed by the selection of preferred time slots to view available rooms and file booking requests with defined

meeting purposes. Users can easily choose their rooms by seeing critical details including number, block location, floor level, seating capacity and present amenities on display.

The system allows administrators full control over room management duties while they examine booking proposition requests and decide on their approvals or rejections to maintain scheduled disposal without conflicts. The system offers a profile page that lets users both check their booking standing and examine their previous bookings.

Through room utilization improvements and automated process automation and transparent information handling "Smart Room Booking and Management System for Academic Institutions" constructs modernized booking systems which boost educational facility operational effectiveness and digital innovation potential.

CHAPTER 2

LITERATURE SURVEY

The research by authors in [1] developed an integrated room scheduling system that showcased educational institutions need automated methods for operating college and university room reservations. Tracking systems presented in the research show outdated approaches while digital methods are established to boost real-time booking features by implementing user authentication with conflict management tools.

Somwong et al. introduced in paper [2] the deployment of a smart scheduling system using cloud technologies. Through this application the researcher demonstrates that Firebase cloud functions enable secure authorization control and real-time space availability tracking and protected database operation which satisfies college facility usage needs.

Sánchez et al. created an automatic manual seminar hall booking system using Android technology for university use [3]. Users from various groups reach their accounts through login then submit booking requests which the administrator accepts. The current system parallels the access functionality that this project implements.

Saravanan and Das presented in paper [4] how they built a platform-compatible classroom management application with a user-friendly interface using Flutter development framework. Research shows that Flutter widgets provide developers the capability to manufacture interfaces which respond quickly to booking requests and presentation schedule modifications and classrooms control inputs.

Implementation of Firebase applications to mobile platforms was explained by the authors of [5] who taught their readers about key advantages, authentication, real-time database operations and book recording functionality in a single centralized infrastructure. The implementation of modular development improves system scalability since new additions such as notifications and analytics become simpler to maintain.

Dighe et al. [6] created a reservation system for meeting rooms which applies conflict resolution techniques to stop double bookings through calendar-based queries during request processing. Through this method the system achieves better consistency and improves room scheduling performance when demand is high.

Research presented in paper [7] demonstrates a modular approach for room management system development which unifies user registration with user time slot approval functions for displaying rooms and managing booking records as a centralized system.

According to Chaichanyut [8] an online meeting scheduler implements role-based access control features which grants administrators the ability to manage scheduling requests. Similar to the present project, this system enables purpose logs during room booking and forwards meeting explanations to the administrator for verification and validation.

The smart booking system developed by Romero et al. [9] features student-specific and staff-exclusive features to enable users to view upcoming and previous booking schedules. The application design includes both Current Bookings and Previous Bookings sections which enables better user interaction in the same manner this system functions.

The author in [10] examined vital implementation practices of Flutter and Firebase mobile development through research into state management technologies and real-time functionality and role-based access controls. Technology implementation in this system generates valuable outcomes for producing superior system performance alongside advanced graphical user interfaces.

Teng documented a smart campus management framework which uses AI enhancement to enable bookings as an integral component in his paper [11]. Modern mobile technology systems allow infrastructure automation alongside predicting infrastructure usage patterns as well as peak operation times.

Xuan [12] described the technical implementation difficulties of establishing booking systems which administrators would operate. The admin interface needs functionality for room maintenance tasks including creation deletion modification booking request evaluation metrics recording which duplicates the features found in the administrative system of this project.

Future versions of this application should include reserved room access protection as proposed by Payne [13] through the integration of building security and authentication enhancements into reservation platforms along with QR codes.

User feedback operations and satisfaction measurements in academic booking platforms are analyzed in the paper by Khawas and Shah [14]. User satisfaction data enables organizations to develop better resource management practices that match resources to user needs.

According to Joorabchi et al. [15] mobile applications should use Firebase to build Flutter applications with maintainable scalable structures. Using a modular approach for development together with data binding practices and asynchronous processing allowed the platform to create a smooth user experience according to research findings.

CHAPTER 3

PROPOSED SYSTEM

3.1 GENERAL

"Smart Room Booking and Management System for Academic Institutions" offers a comprehensive solution that simplifies educational facility booking operations within educational campuses. The platform uses Flutter for building mobile interfaces across platforms while Firebase offers secure real-time operations and data management. Users consisting of students and faculty members alongside administrators can access the system to view vacant rooms while checking time openings and engaging in automatic booking requests as well as reservation management functions. A secure authentication process verifies each user before it assigns roles which determine their unique access limitations and useful features. Users acquire detailed information about rooms through the application because it shows their locations along with seating capacity and facilities and booking histories allowing them to make proper decisions. Real-time updates and approval functions and request management capabilities are granted to administrators through the system. The system delivers swift notifications to users about their booking developments through its integrated notification platform. This system transforms education sector efficiency standards because it performs process automation for manual bookings together with resource maximization and conflict avoidance.

3.2 SYSTEM ARCHITECTURE DIAGRAM

The system architecture, shown in Fig 3.1, presents real-time accessible secure reservation services for seamless operations according to the paper. The system includes three essential features which use Firebase Authentication for Students, Faculty and Admins authorization and Firebase Realtime Database for room database storage along with a booking module that shows rooms and timeslots for booking and updates availability in real time. Through the Administrator Panel users can approve

or disapprove booking requests and add or modify facility rooms. Users receive booking updates through the Notification System and can access their profiles using Profile and History Tracking. Flutter's frontend capabilities lead to a user interface that maintains performance and communicates with the Firebase backend. The system provides real-time updates to avoid scheduling conflicts and secure communication while offering user convenience which results in efficient resource management in academic institutions.

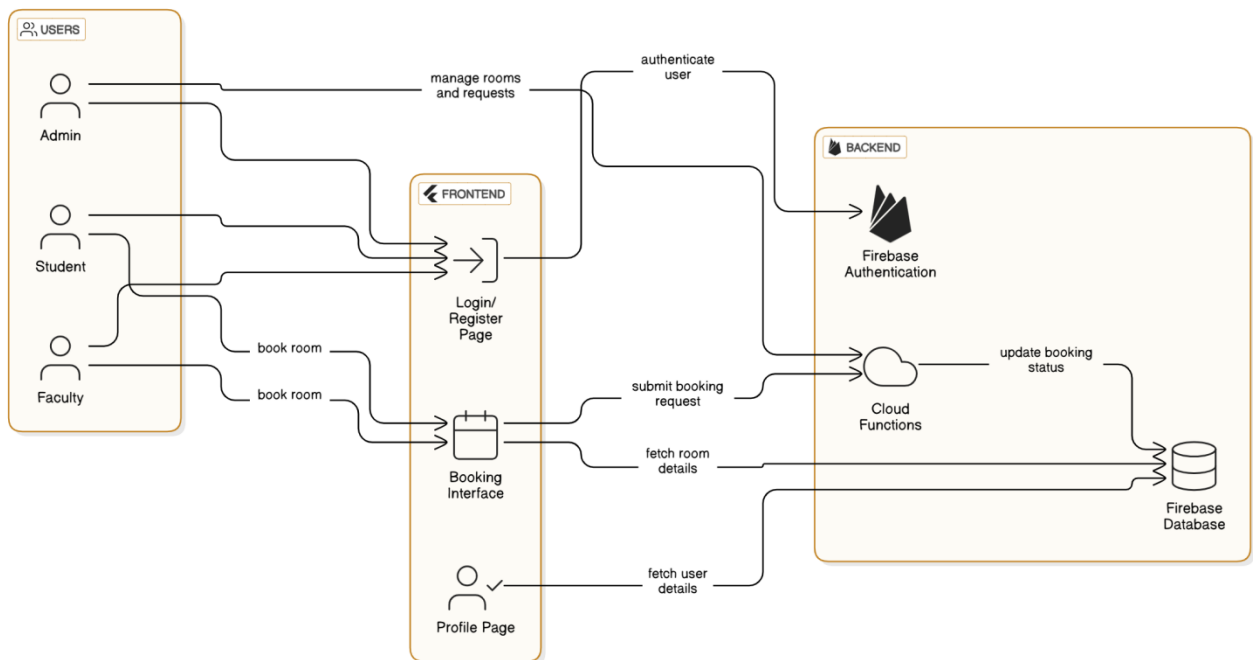


Fig 3.1: System Architecture

3.3 DEVELOPMENTAL ENVIRONMENT

3.3.1 HARDWARE REQUIREMENTS

The hardware specifications could be used as a basis for a contract for the implementation of the system. This therefore should be a full, detailed description of the whole system. It is mostly used as a basis for system design by the software engineers.

Table 3.1 Hardware Requirements

COMPONENTS	SPECIFICATION
PROCESSOR	Intel Core i3 or higher
RAM	4 GB RAM or more
POWER SUPPLY	+5V power supply

3.3.2 SOFTWARE REQUIREMENTS

The software requirements document contains the system specifications. This is a list of things which the system should do, in contrast to the way in which it should do things. The software requirements are used to base the requirements. They help in cost estimation, complete tasks, and track team progress in the development .

Table 3.2 Software Requirements

COMPONENTS	SPECIFICATION
Operating System	Windows 10 or higher
Frontend	Flutter (Dart)
Backend	Firebase (Authentication, Realtime Database, Firestore)
Database	Firebase Realtime Database & Cloud Firestore
Development IDE	Android Studio / Visual Studio Code

3.4 DESIGN OF THE ENTIRE SYSTEM

3.4.1 ACTIVITY DIAGRAM

The activity diagram, Fig 3.2, shows the operational sequence of the Smart Room Booking and Management System. All user activities start when students and staff members access the system through web or mobile interfaces to choose between being students and staff or administrators. Users who belong to the faculty or student categories can find available rooms and inspect details before requesting specific time slots for booking. The system verifies current room availability thus users see only rooms available for the selected time period. The booking request gets transmitted to the Admin for their evaluation after submission. After room availability evaluation the admin makes final booking decisions which lead to status updates in the system database. The system delivers instantaneous status alerts about booking acceptance or denial to the user. Users have the capability to inspect their booking records at any moment. The integrated process delivers timely secure procedures for academic center room scheduling operations which enables users and administrators to have an uncomplicated positive experience.

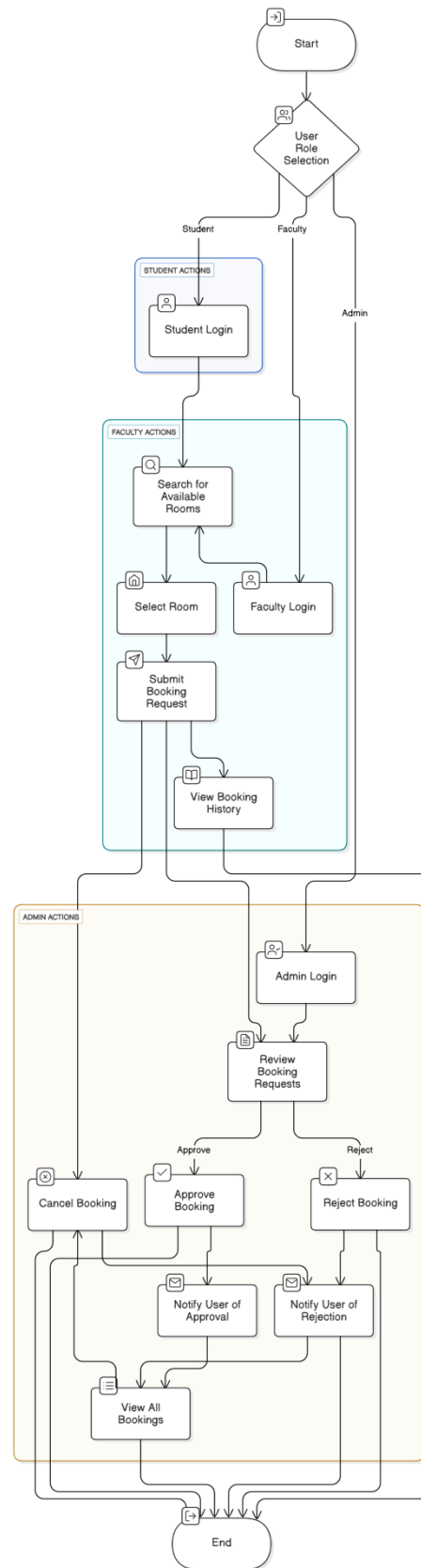


Fig 3.2: Activity Diagram

3.4.2 DATA FLOW DIAGRAM

The data flow diagram, Fig 3.3, outlines the entire process through which users handle room booking requests. The process starts by allowing users to input their requests for selecting a specific room together with a chosen time slot. When enough available rooms exist within Firebase the system produces a booking request. After the admin reviews the booking request the system automatically updates the status. The real-time booking data synchronization system of Firebase operates across all user processes. All room records together with user information and booking event histories are saved within Firebase Realtime Database. Booking confirmations and updates receive notifications to all users through the system. The system operates with flawless transition to process room bookings securely along with managing updates.

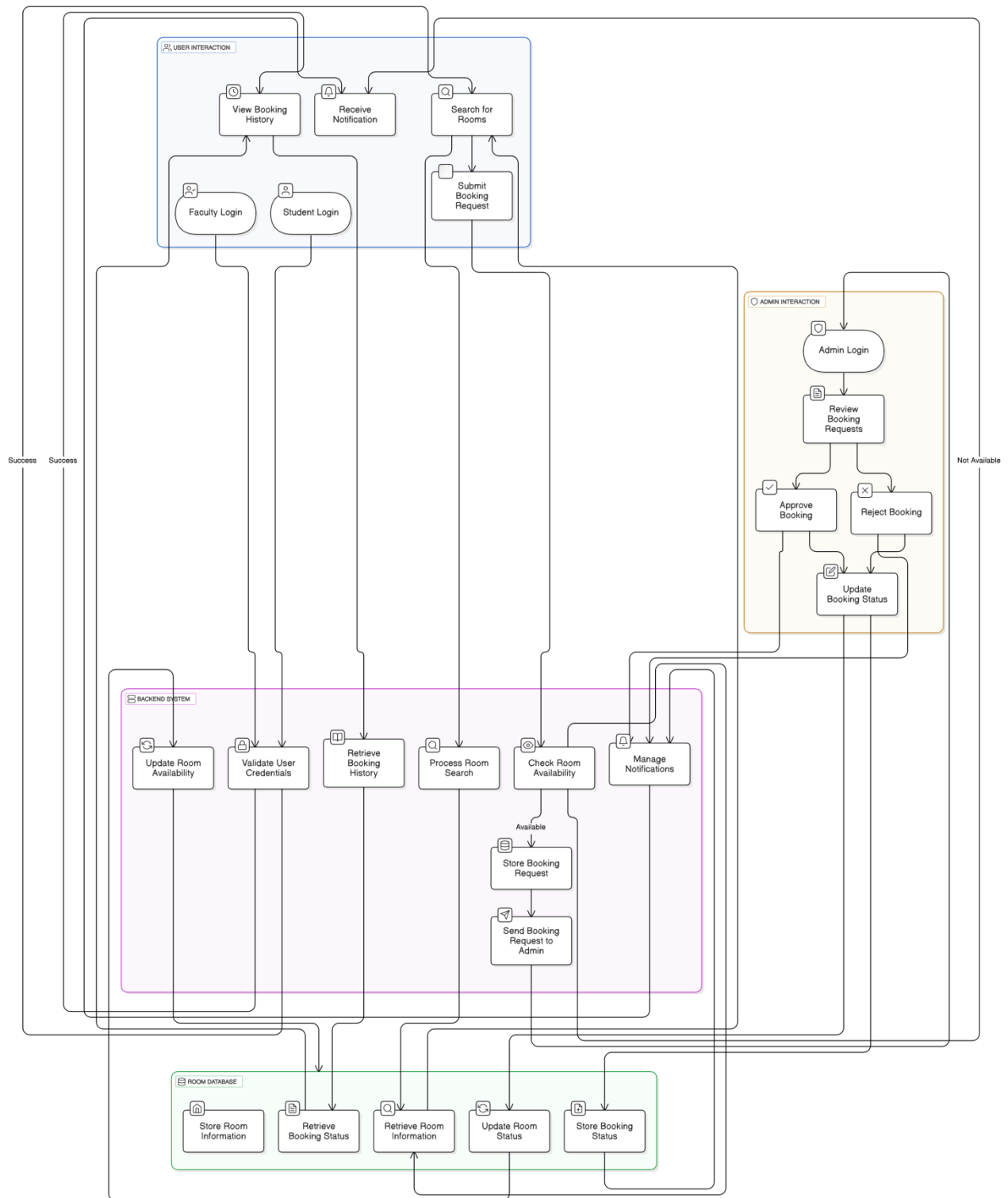


Fig 3.3:Data Flow Diagram

3.5 STATISTICAL ANALYSIS

The feature comparison table demonstrates the contrasts which exist between conventional room booking platforms and contemporary Smart Room Booking and Management System. Through integration of Firebase and Flutter the proposed system implements real-time feature updates and user role assignment systems that perform synchronized data communication for multiple devices. Users of the proposed system will experience both an improved interface design as well as enhanced capability to track room availability.

Table 3.3 Comparison of features

Aspect	Existing System	Proposed System	Expected Outcomes
Room Availability	Manual check or limited updates	Real-time availability check using Firebase	Instant room availability updates
User Interaction	Basic interface	Responsive UI with Flutter	Improved user experience across platforms
Role Management	Limited user roles	Role-based access (Student, Faculty, Admin)	Enhanced user access control and security
Booking Management	Manual updates	Real-time updates and admin panel for approval	Streamlined booking process and faster management
Data Synchronization	Rarely synchronized	Real-time sync across devices with Firebase	Seamless data updates and conflict reduction

The Smart Room Booking and Management System differentiates itself through real-time synchronization along with role-based user management and a responsive interface implemented with Flutter. The system implements features which boost performance and improve user satisfaction to create an adaptive efficient room booking management system. Multiple features of the system leverage a Firebase implementation to ensure real-time synchronization and reduce booking conflicts while creating operational transparency. The platform adjusts to various user roles that enables individualized secure access for students together with faculty members and administrators. The proposed solution exhibits features of scalability and security alongside optimized capacities designed for large academic institutions to bring streamlined control to room bookings along with management activities.

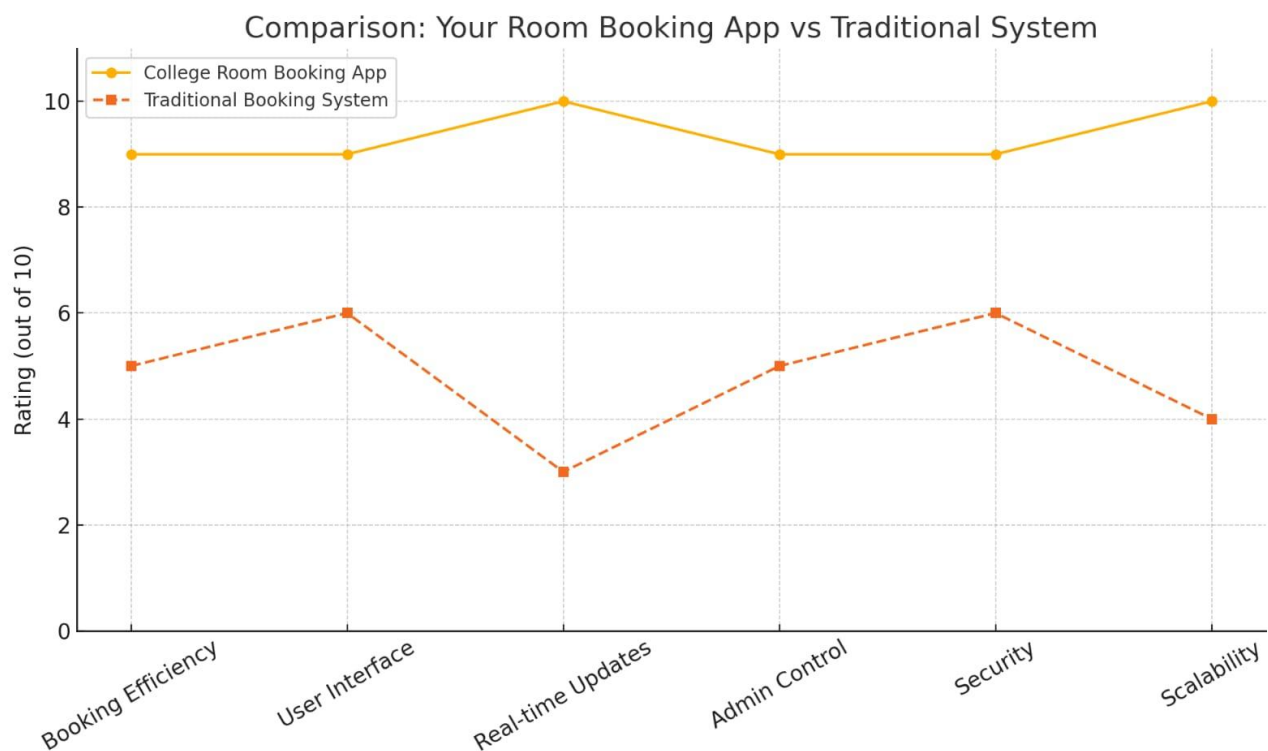


Fig 3.4 : Comparison Graph

CHAPTER 4

MODULE DESCRIPTION

The workflow for the proposed Smart Room Booking and Management System is designed to ensure a structured, real-time, and efficient process for managing room bookings across an academic institution. The system focuses on providing a seamless experience for students, faculty, and administrators through secure authentication, role-based access, and smooth booking management.

4.1 SYSTEM ARCHITECTURE

4.1.1 USER INTERFACE DESIGN

The sequence diagram Fig 4.1 presents an illustration of the booking process which begins when users reach the application or web platform. Users start by logging into the system followed by selecting their role position which includes Student, Faculty or Admin. Users who belong to different roles can perform room searches then examine available details followed by an ability to request specific booking time slots. After reviewing new requests Admins carry out both approval or denial functions while they maintain booking statuses. Whenever booking requests need to be notified users receive alerts which they can also check their previous booking records through the interface.

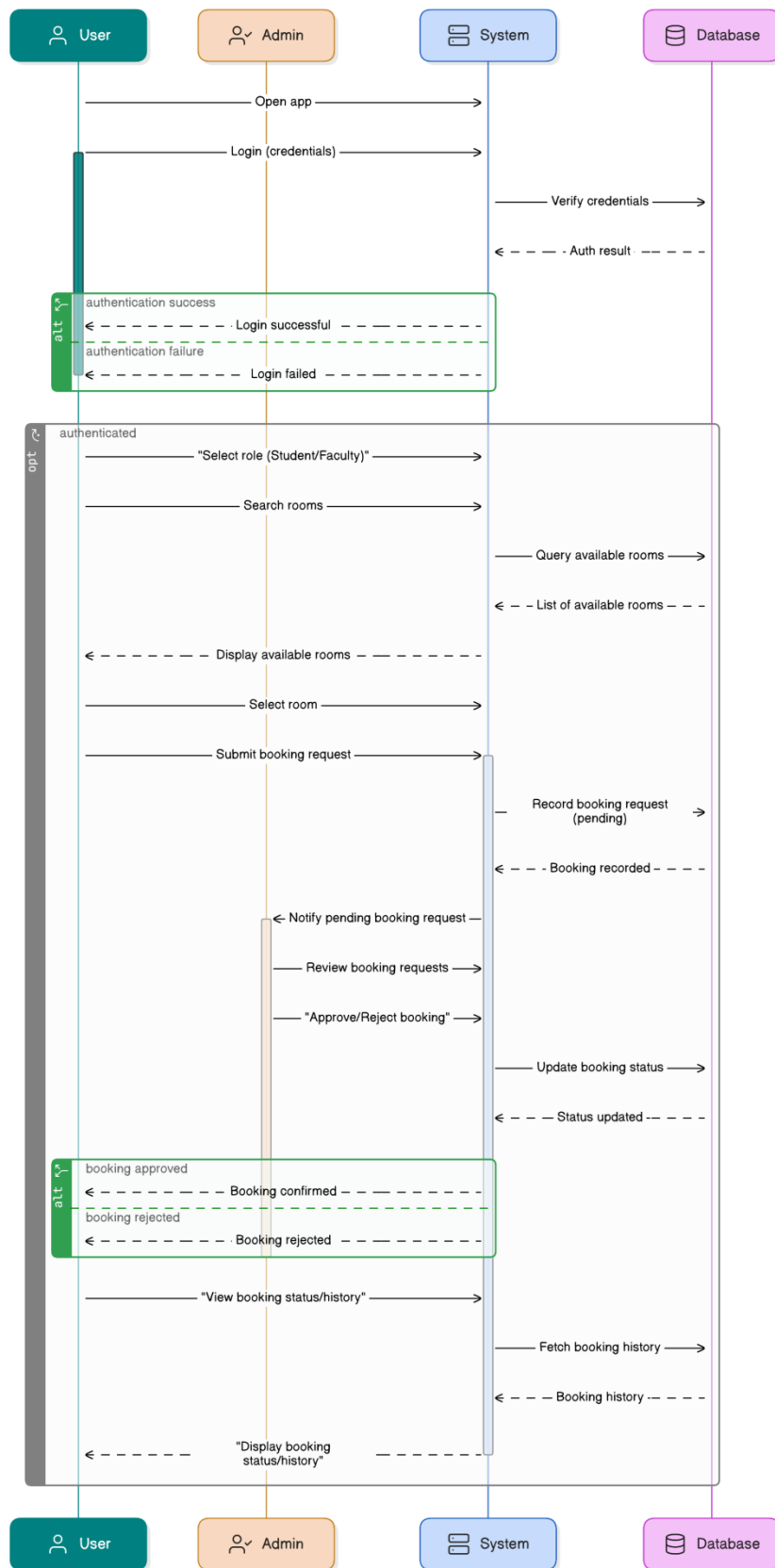


Fig 4.1: SEQUENCE DIAGRAM

4.1.2 BACK END INFRASTRUCTURE

The backend infrastructure supports real-time operations by managing user accounts, room data, booking records, and notifications. It consists of a secure authentication service (like Firebase Authentication) for managing user login and roles, and a cloud-hosted database (like Firestore) for storing room information, booking requests, and approvals. APIs built using frameworks like Flask or Node.js handle the interaction between the mobile/web interface and the database, ensuring smooth communication and real-time updates for all users.

4.2 ROOM MANAGEMENT WORKFLOW

4.2.1 Room Information Management

The system maintains an updated database of all available rooms, including details such as room capacity, resources (e.g., projector, whiteboard), location, and available time slots. Admins can add, edit, or remove room details as needed.

4.2.2. User Authentication and Role Selection

Users sign in securely through the system and select their designated role (Student, Faculty, or Admin). Access rights are role-based:

Students and faculty can book rooms.

Admins have additional privileges to manage bookings and room details.

4.2.3 Search and Booking Requests

Users can search available rooms by filters like date, time, room capacity, and location. After viewing room details, users submit a booking request specifying the desired time slot and purpose.

4.2.4 Booking Verification and Approval

The system checks room availability in real-time. If the slot is available, the request is forwarded to the Admin.

Admins review the requests, approve or reject them based on availability, priority, and institution policies.

Upon approval/rejection, the booking status is updated, and users are notified instantly.

4.2.5 Booking History and Record Maintenance

Users can view their booking history, including past and upcoming reservations.

Admins can monitor overall booking statistics and generate reports for analysis and future planning.

4.3 SYSTEM WORK FLOW

4.3.1 User Interaction:

Users (students, faculty, or admins) interact with the mobile app or web interface to log in, search for rooms, submit booking requests, and view their booking status and history.

4.3.2 Room Search and Request Submission:

Students and faculty search for available rooms using filter options and submit requests for the preferred rooms and time slots. The system dynamically updates available slots based on real-time data.

4.3.3 Admin Review and Approval:

After Admin action, users are notified of the approval or rejection. Confirmed bookings are updated in the database to avoid double bookings.

4.3.4 Continuous System Update:

The system updates the booking database continuously to reflect real-time room

occupancy. Feedback mechanisms can also be introduced to allow users to report issues with rooms or suggest improvements.

This structured workflow ensures efficient room booking management, minimizes scheduling conflicts, improves transparency, and enhances user satisfaction within academic institutions.

CHAPTER 5

IMPLEMENTATION AND RESULTS

5.1 IMPLEMENTATION

The project is makes use of Flutter combined with Dart programming language for creating frontend interfaces which provide consistent cross-platform usability. Firebase manages the application backend through its Authentication feature to secure user management and tracks room information and bookings in the Firestore Database. Through its user-friendly web and mobile interface users can perform registration and login routines while searching rooms and booking them and monitoring their booking records. The system includes an access management system that provides distinct permission levels to students and faculty members and administrators. The administrative section enables administrators to conduct operations including room addition and user and booking management alongside request approval or rejection. The system uses real-time reporting which maintains correct booking-related information at all times. The system enables secured authentication through Firebase Authentication and synchronization occurs automatically with Firestore. The solution supports increased scalability because developers plan to add room usage analytics along with feedback collection capabilities and push notification features that will enhance user interface quality.

5.2 OUTPUT SCREENSHOTS

The output screenshots highlight the major functionalities of the project. Fig 5.1 displays the Login and Register screens where users can create new accounts or log in to access the platform. Fig 5.2 shows the User Dashboard, offering users quick access to book rooms, view their bookings, and manage their profiles. Fig 5.3 showcases the Book Room and View Available Rooms page, allowing users to check available rooms based on date, time, and capacity. Fig 5.4 presents the Enter Purpose and Send Request screen,

where users fill out booking details such as the purpose of booking and submit requests. Fig 5.5 illustrates the User Profile page where users can view and edit their personal details. Fig 5.6 displays the Current Bookings and Previous Bookings section, helping users track their ongoing and past room bookings. Fig 5.7 highlights the Admin Dashboard, providing admins with an overview and management access to users, rooms, and booking requests. Fig 5.8 shows the List of Users and Rooms, allowing admins to view and manage registered users and existing rooms. Fig 5.9 depicts the Add Room screen, where admins can add new rooms by entering details like room name, capacity, and location. Finally, Fig 5.10 illustrates the Requests and Booking History page, where the admin can view all incoming booking requests, approve or reject them, and monitor the booking history effectively. These screenshots collectively demonstrate the complete flow of the room booking system, ensuring a seamless experience for both users and administrators.

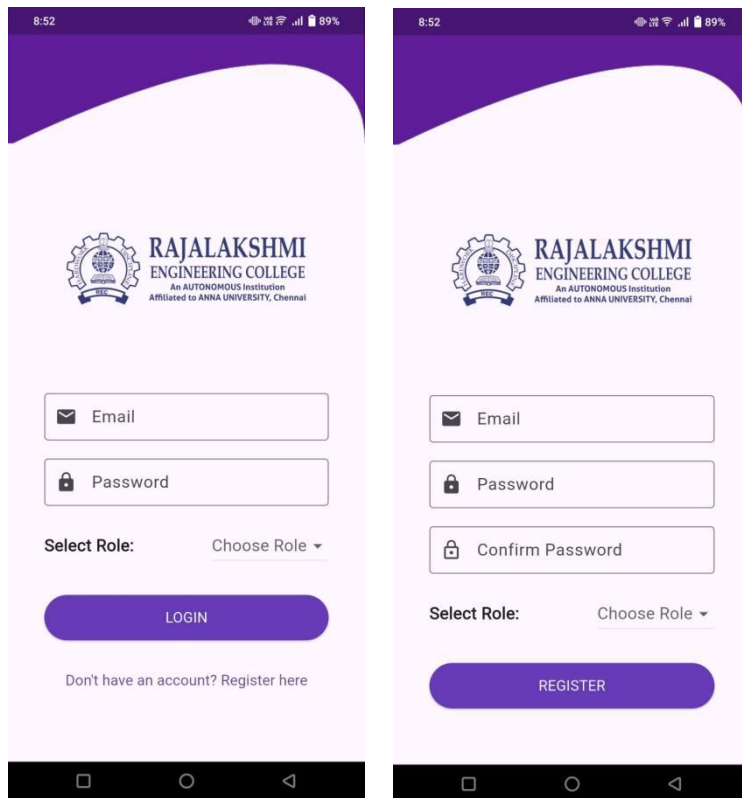


Fig 5.1 Login and Register Screens

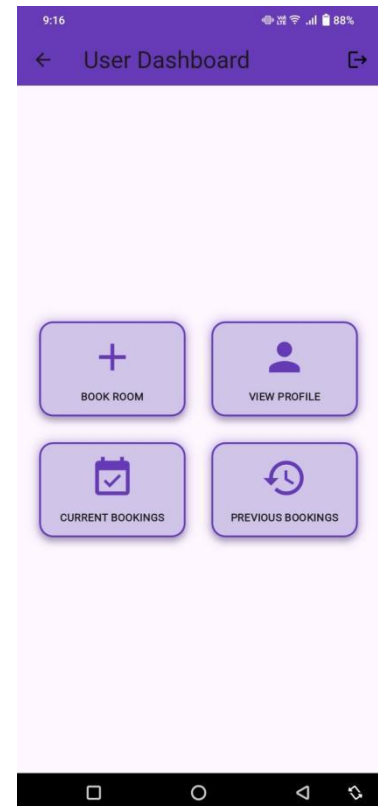


Fig 5.2 User Dashboard

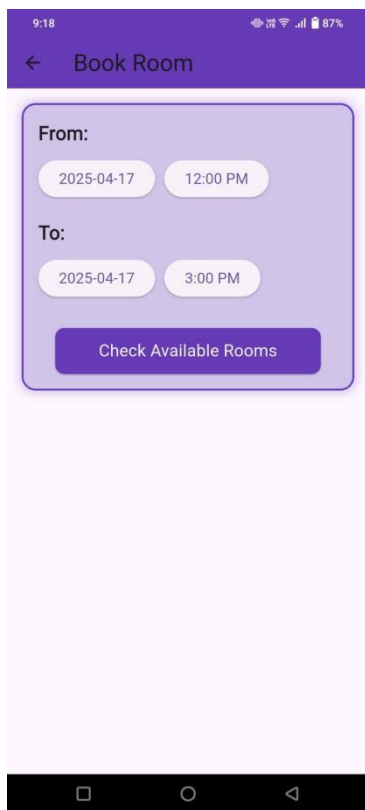


Fig 5.3 Book Room and View Available Rooms

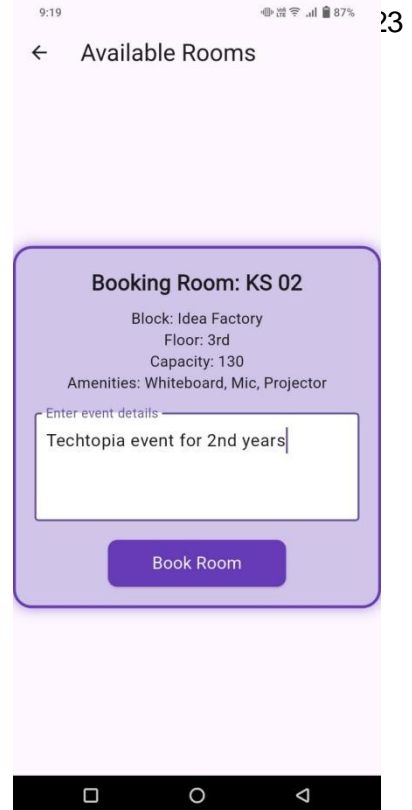
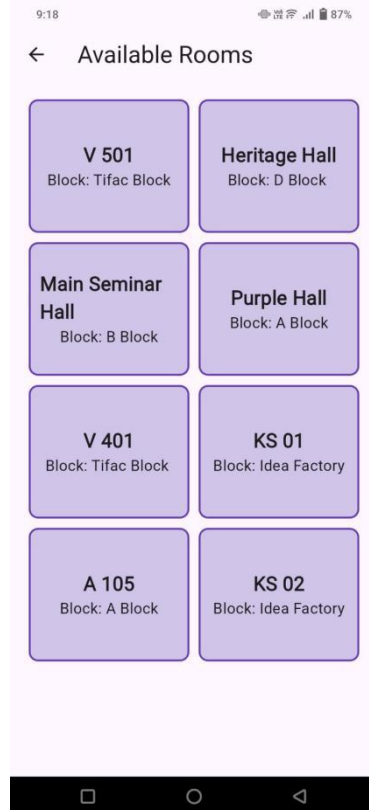


Fig 5.4 Enter Purpose and Send Request

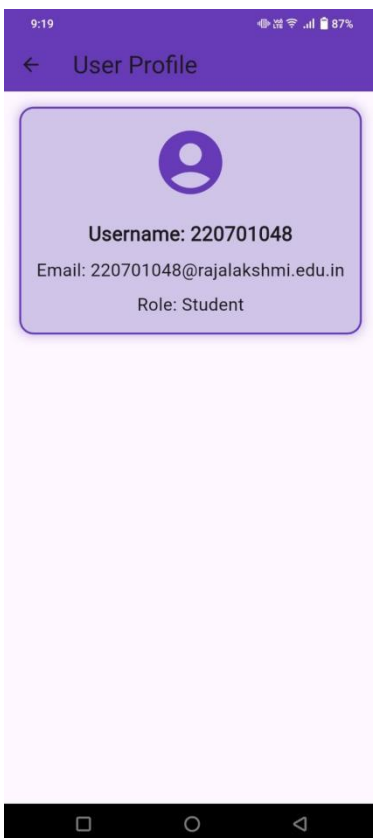


Fig 5.5 User Profile

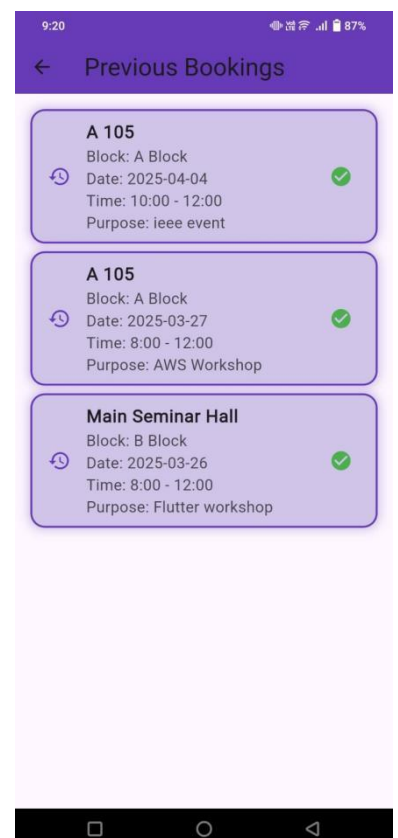
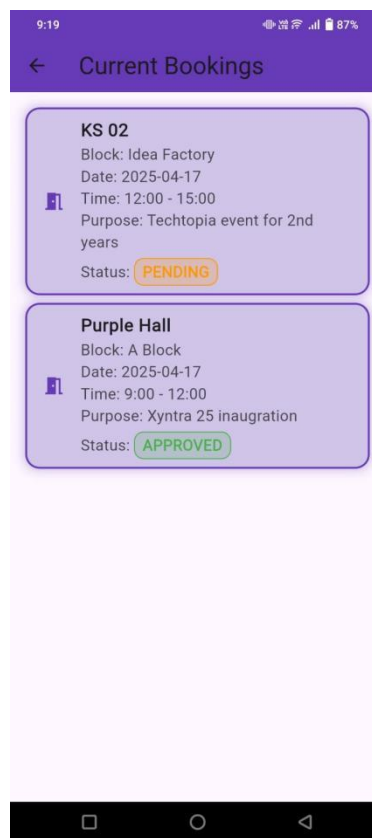


Fig 5.6 Current Bookings and Previous Bookings

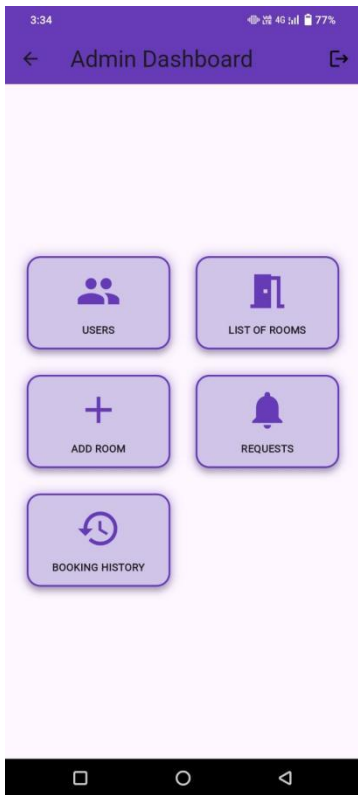


Fig 5.7 Admin Dashboard

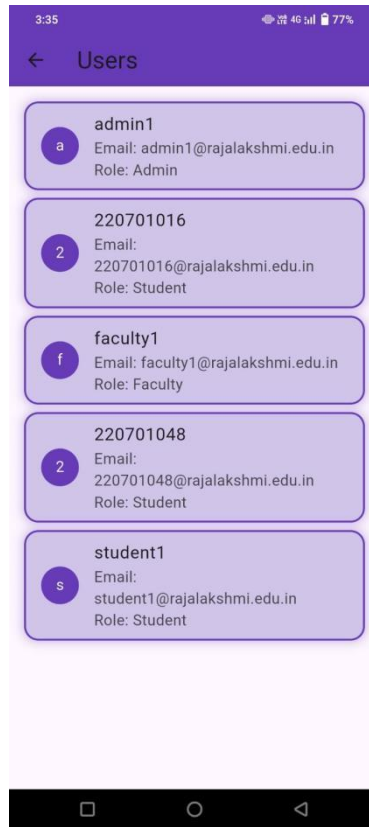


Fig 5.8 List of Users and Rooms

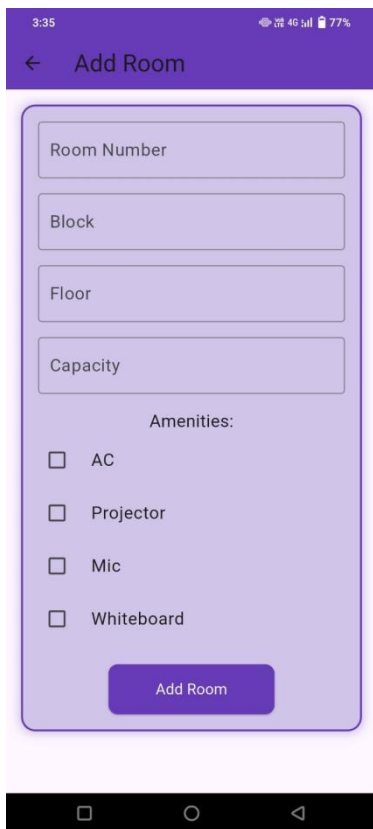


Fig 5.9 Add Room



Fig 5.10 Requests and Booking History



CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

The proposed Smart Room Booking and Management System delivers an effective user-friendly approach for room booking operations. The platform achieves smooth room booking administration for students along with faculty members and administrators by adopting real-time data synchronization mechanisms combined with role-based access control systems and flexible server infrastructure. The system benefits from improved performance and reliability because it incorporates Firebase which secures database management and authentication processes. Traditional Flutter design makes it possible for users to look for available rooms while submitting booking requests which results in real-time notification delivery. Keeping track of bookings becomes efficient for both students and faculty members because of the admin panel's benefits. Real-time updates merged with security capabilities allow this system to maximize room usage along with enhancing the entire booking experience for institutional settings.

6.2 FUTURE ENHANCEMENT

Future enhancements for the Smart Room Booking and Management System would be a mechanism involving machine learning models that analyze historical room usage patterns to produce future predictions. The system should provide support for multi-campus management abilities that enable facilities with multiple locations to schedule bookings among their different campuses. The future iteration of this system will analyze the potential of adding compatibility with intelligent IoT devices which would enable live monitoring of room conditions for increased user satisfaction. The system's integration of these added capabilities would lead to superior scalability together with enhanced flexibility as well as increased user satisfaction.

REFERENCES

- [1] B. Walek, O. Hosek and R. Farana, "Proposal of expert system for hotel booking system," 2016 17th International Carpathian Control Conference (ICCC), High Tatras, Slovakia, 2016.

- [2] P. Somwong, S. Jaipoonpol, P. Champrasert and Y. Somchit, "Smart Room Vacancy Status Checking and Booking System," 2022 37th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), Phuket, Thailand, 2022.

- [3] L. M. Sánchez, I. Díaz-Oreiro, L. Quesada, L. A. Guerrero and G. López, "Smart Meeting Room Management System Based on Real-Time Occupancy," 2019 IV Jornadas Costarricenses de Investigación en Computación e Informática (JoCICI), San Pedro, Costa Rica, 2019,

- [4] M. Saravanan and A. Das, "Smart real-time meeting room," 2017 IEEE Region 10 Symposium (TENSYP), Cochin, India, 2017

- [5] Mastering Firebase for Android Development. (n.d.). Google Books. <https://books.google.com/books?hl=en&lr=&id=RMNiDwAAQBAJ&oi=fnd&pg=PP1&dq=related:ZjHH1k2mx68J:scholar.google.com/&ots=R51yMVYtZs&sig=yItZLmFszGFZsD-iVPpUh6i7ATI>

- [6] Dighe, A., Kokate, S., Khan, I., Jonathan, E., & Jadhav, R. (2025). RooMX: integrating Data-Driven analytics and User-Friendly interfaces for optimized meeting room booking. In Lecture notes in networks and systems (pp. 71–89). https://doi.org/10.1007/978-981-97-9523-9_7

- [7] Haryadi, E., Maelani, D., Wijayanti, D., Yuliandari, D., & Widyastuti, I. (2021). MOBILE-BASED DEVELOPMENT OF MEETING ROOM MANAGEMENT APPLICATION AT PT JIWALU STUDIO. *JURNAL TEKNOLOGI DAN OPEN SOURCE*, 4(2), 166–173. <https://doi.org/10.36378/jtos.v4i2.1703>
- [8] Chaichanyut, W. (2024, August 13). Development of online meeting room reservation. <https://publication.npru.ac.th/handle/123456789/2259>
- [9] Romero, S. L., Velarde, A., & Rosas, M. A. (2021). Smart meeting room scheduling and management system for a university campus using Android tablets with Firebase backend and Headwind MDM. *Proceedings of the 19th LACCEI International Multi-Conference for Engineering, Education, and Technology: “Prospective and Trends in Technology and Skills for Sustainable Social Development” “Leveraging Emerging Technologies to Construct the Future.”* <https://doi.org/10.18687/laccei2021.1.1.383>
- [10] Conference Room Booking Application using Flutter. (2020b, July 1). IEEE Conference Publication | IEEE Xplore. <https://ieeexplore.ieee.org/abstract/document/9182183/>
- [11] Teng, P. (2014). Online Room-Booking System based on database. *Applied Mechanics and Materials*, 513–517, 1748–1751. <https://doi.org/10.4028/www.scientific.net/amm.513-517.1748>
- [12] Xuan, W. (2021). Implementation of a secure room booking system at the University of Manitoba Libraries. *International Journal of Librarianship*, 6(2), 63–72. <https://doi.org/10.23974/ijol.2021.vol6.2.194>

- [13] Payne, R. (2019). Using Firebase with Flutter. In Apress eBooks (pp. 255–285). https://doi.org/10.1007/978-1-4842-5181-2_12
- [14] Khawas, C., & Shah, P. (2018). Application of Firebase in Android App Development-A Study. *International Journal of Computer Applications*, 179(46), 49–53. <https://doi.org/10.5120/ijca2018917200>
- [15]Joorabchi, M. E., Mesbah, A., & Kruchten, P. (2013). Real Challenges in Mobile App Development. *App Development*. <https://doi.org/10.1109/esem.2013.9>