An Najah National University

Faculty of Engineering & Information Technology

Department of Computer Engineering

Edu Sync

Done By:

1-Israa Salman

2-Batool Kittaneh

Supervisor:

Dr.Alaaddin Masri

Dr.Suleiman Abu-Kharmeh

June 18, 2025

Acknowledgements

"First of all, we would like to thank our supervisor, Dr.Alaaddin Masri & Dr.Suleiman Abu-kharmeh, for their constant efforts during the semester, and for being fully prepared in order to help us with scientific support. We thank our family and friends. Those who devoted their undivided attention and had faith in our capability."

Disclaimer

Israa Salman and Batool Kittaneh have written this report as requirements for a Bachelor's degree in Computer Engineering Department. No one modifies or corrects it because it will be evaluated by professors at An Najah National University. It is worth mentioning that An-Najah National University does not have any responsibility for any word in this report.

Abstract

Doctors and users can register using their university email addresses. External parties must wait for supervisor verification before viewing a special page that allows them to publish equipment, not the full application.

The instructors can create group chats by inviting students using their university email address and grouping them according to the type of project (hardware or software). This will provide each supervisor with his/her own personal chat room with his/her students.

Each group will possess a smart calendar where they can plan meeting times, assign tasks, take notes, and make modifications. There will be an upload area as well, where students can upload pictures, videos, or reports.

After a project is completed, it will be stored in the database, and future students will be able to read and learn from previous graduation projects.

Every student will have the capability to post entries, either inquiring about assistance for hardware components or software project ideas.

There shall be a dedicated interface for displaying hardware components, and with full information given for each one of them.

We will also include an AI chatbot that students can interact with for faster searches and inquiries.

In addition, the system will give notifications for various activities

TABLE OF CONTENTS

Acknowledgements 1

Disclaimer 2

Abstract 3

Introduction 9

- **General background** 10
- **Objectives** 10
- **Scope of the work** 10
- **Significance** 11

LETERATURE REVIEW 11

- **Theoretical Background of the Project** 11
- **Previous Similar Works** 12

Methodology 12

- **Constraints and Earlier Coursework** 12
- **Used Technologies** 14

EduSync as a Mobile Application 18

- **.1 Authentication and Validation** 18
  - **Signup** 19
  - **Login** 20
  - **Forget password** 23
  - **Reset password** 24
- **2. User Profile** 25
- **3. Home Screen** 30
- **4. Hardware Equipment** 35
- **5. AI Chatbot Page** 37
- **6. Previous Projects** 40
- **7. Chat Screen** 42

Private Chat Page 43
Group Chat Page 44
8. Smart Calendar Page 47
9.Group Files Page 54
10. Notification Page 55
EduSync as a Website 56
Results & Discussion 69
Result 69
Discussion 69
Conclusion & Future Work 70
Conclusion 70
Future Improvements 70

List of Figures

Figure 1 Login Page Overview....18
Figure 2 Sign-Up Page 19
Figure 3 Sign up Validation 2 20
Figure 4 Sign up Validation 1 20
Figure 5 Login Page 21
Figure 6 Login Verification 22
Figure 7 Forgot Password Page....23
Figure 8 Successful send code 23
Figure 9 Reset Password Page....24
Figure 10 My Activities 25
Figure 11 Personal information 25
Figure 12 My Comments 26
Figure 13 My Likes....26
Figure 14 Saved Posts....27
Figure 15 My Posts 27
Figure 16 My Account 28
Figure 17 Logout 29
Figure 18 Home Screen Page 30
Figure 19 Search for a user 31
Figure 20 Search for an External 32
Figure 21 Search for a Student 32
Figure 22 External user requests 33
Figure 23 External post page 34
Figure 24 Hardware Equipment Page....35
Figure 25 Add equipment....36
Figure 26 Search for equipment 36
Figure 27 AI Chatbot Page 37
Figure 28 AI Chatbot History....38
Figure 29 New AI Chatbot....38
Figure 30 Chat with Ai 39
Figure 31 Previous Projects Page 40
Figure 32 filtering previous projects 41
Figure 33 search in previous projects 41
Figure 34 Chat Screen Page 42
Figure 35 Private Chat Page....43
Figure 36 When a Group Already Exists 44

Figure 37 Creating a group chat 44
Figure 38 Group Screen Page 45
Figure 39 Group Screen filtering 45
Figure 40 Message control 46
Figure 41 Group Chat Page 46
Figure 42 Join Meeting 48
Figure 43 Smart Calendar Page 49
Figure 44 Meeting Type 49
Figure 45 Selecting the hour 50
Figure 46 Selecting a minute 50
Figure 47 Selecting Date 50
Figure 48 Add note 51
Figure 49 The meeting was completed 51
Figure 50 Reactivate Meeting 52
Figure 51 Confirm Attendance 52
Figure 52 End Meeting 53
Figure 53 Meeting In Progress 53
Figure 54 Group File Page For Student 54
Figure 55 Group File Page For Doctor 54
Figure 56 Notification Page 55
Figure 57 Login Page-Web 56
Figure 58 Signup-Web 56
Figure 59 Forget Password-Web 57
Figure 60 Reset Password-Web 57
Figure 61 Home Screen Page-Web 58
Figure 62Personal information-Web 58
Figure 63 My Activities-Web 59
Figure 64 My Account-Web 59
Figure 65 Hardware Equipment-Web 60
Figure 66 Add Equipment-Web 60
Figure 67 AI Chatbot-Web 61
Figure 68 New AI Chatbot-Web 61
Figure 69 Previous Projects Page-Web 62
Figure 70 filtering previous projects-Web 62
Figure 71 Chat Page-Web 63
Figure 72 Private Chat -Web 63
Figure 73 Edit message-Web 64
Figure 74 Groups Chat -Web 64

Figure 75 Group Chat-Web 65
Figure 76 Smart Calendar Page-Web 65
Figure 77 Selecting Date-Web 66
Figure 78 Selecting hour 66
Figure 79 Selecting minute -Web.....67
Figure 80 Meeting Type-Web 67
Figure 81 Group Files Page -Web 68
Figure 82 Notification Page-Web 68

Introduction

Computer tools have been a requirement for managing academic projects in recent years, especially graduation projects, which require continuous communication, planned scheduling, and efficient sharing of assets. However, the majority of the students and faculty members continue to utilize incomplete collaboration sites, which can lead to disorientation, missed deadlines, and ineffective collaboration. This project offers a strong web-based application exclusively designed to support the organization and communication of university graduation projects. The site aims to make student-supervisor communication easier, with functionalities unique to academic environments such as group messaging, smart calendars, file sharing, and project spaces divided by type The system also offers controlled access via registration of students and instructors with university addresses for direct sign-up, with outside agents needing admin permission and limited to a view-only interface for hardware aspects. Through the incorporation of features such as a searchable component interface, an artificial intelligence-based chatbot, and real-time messaging, the application simplifies all the efficiency, accessibility, and organization of graduation project workflows, This application not only facilitates existing project management but also establishes a beneficial repository of previous projects so that future students can learn from previous experience and ideas. By so doing, it closes communication gaps and facilitates a coherent, cooperative scholarly community.

General background

Graduation projects are part of the life of academe because students can apply theoretical knowledge to actual practice. The process of managing these projects—most importantly communication, planning, file sharing, and documentation—is largely not centralized and automated. Currently, tools available in universities are either not appropriate for academic work or do not bring together all the features required in one platform. This project aims to solve this problem by developing a comprehensive web application designed specifically for dealing with and streamlining graduation project processes.

Objectives

In EduSync, we are developing a central platform that facilitates management of graduation projects. It accommodates student and staff registration using their university email addresses and allows instructors to create group chats by project type. It includes an intelligent calendar for meeting management and task follow-up, file loading (images, reports, videos), and searchable hardware component interface. It also has an AI chatbot for instant queries, restricts external access with admin approval, and saves finished projects in a database for easy reference.

Scope of the work

1. Personal Account Creation: Students and lecturers can create their own accounts using their university email addresses, whereas external users have to be approved by the admin.
2. Group Chat Creation: Lecturers can make group chats and group students according to the project type (hardware/software).
3. Private Chat: Private one-to-one chat is provided for users to communicate privately.
4. Smart Calendar: Each group possesses a calendar where meetings, deadlines, and editable notes are scheduled.
5. File Uploads: Students can upload images, videos, and reports for their projects.
6. Project Archiving: Completed graduation projects are stored in the system so that upcoming students could view and learn from them.
7. Post Creation: Posts can be created by students to post queries or request help related to hardware components or software ideas.
8. Component Display: A special interface is available for displaying hardware components in full detail.

9. AI Chatbot Support: Students can be helped using an AI chatbot for quick searching and component advisory.
10. Notifications: Users will receive notifications for significant activities such as file uploads, new messages, scheduled meetings, and new posts.
11. External Access Control: External users can only view component details after receiving admin approval.

Significance

This software provides a unified, easy-to-use platform appropriately tailored to the graduate school context of graduation projects. It allows for student and supervisor communication, reduces organizational overhead, and promotes collaboration. Furthermore, by retaining completed projects and offering component data, it helps later students acquire inspiration and practical resources. The addition of smart features like chatbots and notifications provides a modern and interactive user experience.

LETERATURE REVIEW

Theoretical Background of the Project

It is a web and mobile application website intended to aid students and professors during their graduation project journey. It is not a business or profit organization but a project whose purpose is to make easy collaboration and sharing of knowledge possible. With the system, students can manage group projects, view past projects, communicate with peers and supervisors, search and find information about hardware components, and stay on top of things using a clever calendar and shareable documents. The platform also assists the learning experience through AI-powered aid and an searchable repository of project items.

Previous Similar Works

While the idea of project management platforms is not new, this system differentiates itself from others because it is particularly designed for engineering and IT students working on graduation projects. As opposed to general-purpose application software such as Microsoft Teams or Trello, this application combines academic group discussion forums, storage of projects, navigation of hardware components, and smart scheduling in one package. It also incorporates limited external user access and encourages student participation through post sharing and direct messaging. Whereas other educational sites exist, none reproduce this system's full range of academic and technical support in the same localized, student-focused environment.

Methodology

Constraints and Earlier Coursework

Restrictions and Previous Coursework

1-Economy

Budget Limits:

Since this project is non-profit and student-driven, it was important to produce it on a shoestring budget. We worked with open-source software and free versions of applications like Firebase and Cloudinary in an effort to stay within a tight budget.

Maintenance Charges

Maintenance activities such as bug fixing, security patches, and server upkeep are expected to be performed by student developers or institutional support.

2-Environment

Power Consumption:

The system is designed to be performance-focused with low energy consumption for both client and server systems.

Environmentally Friendly Hosting:

Hosting the system on cloud providers that offer green energy support is the suggestion for reducing the carbon footprint.

3-Society

Information Security:

Authentication through university emails and Firebase safely stores information. JWT tokens and access control also protect user information.

Privacy:

Normal privacy policies are adhered to by the system—users' posts, files, and messages are private to their own group or chat unless posted openly with permission.

4-Politics

Inclusivity and Accessibility:

The site is designed to be accessible and inclusive for all students and teacher members regardless of background or specialization.

Compliance:

The platform is built according to national data protection and academic law.

5-Ethics

Safe Usage:

The platform does not contain objectionable content and encourages healthy academic interaction. Features like reminder for meetings and smart calendars reduce stress and missed deadlines.

Fair Use:

It encourages ethical sharing of knowledge—students who view previous projects must cite them appropriately and not copy.

6-Health and Safety

Digital Wellbeing:

With scheduling of tasks and smart notifications, the system aims to reduce digital overload and have clean communication without spamming.

7-Sustainability

Scalability and Upgradeability:

The system has been designed with the philosophy of modularity as the design approach, hence it is easy to add new features like discussion forums, advanced analytics, or integration with research software in future versions.

Used Technologies

Flutter

Flutter is an open-source UI software development kit created by Google, used to build natively compiled applications for mobile, web, and desktop from a single codebase. It uses the Dart programming language and offers a rich set of pre-designed widgets, which enables the rapid development of cross-platform applications.

Why Flutter?

Cross-Platform Development: Allows the development of applications for Android, iOS, and web with a shared codebase, saving time and effort.

High Performance: Uses a Skia engine for rendering, providing a smooth and fast UI experience.

Hot Reload: Allows real-time code changes during development, enhancing productivity.

Customizable Widgets: Gives liberty to design user interfaces according to the project's need, for example, this application's file management UI.

Firebase

Firebase is a platform for building web and mobile applications developed by Google, which provides a suite of tools like Firestore (a NoSQL cloud database), authentication, and cloud functions. It is used in this project for real-time data storage and synchronization.

Why Firebase?

Real-Time Database: Offers real-time updates for group data and notifications so that users see the most up-to-date information in real-time.

Scalability: Handles growing data needs as the user base and groups increase.

Authentication: Simplifies user login and security management.

Integration: Seamlessly integrates with Flutter for group information handling and participant emails.

Node.js

Node.js is an open-source, free, cross-platform JavaScript runtime environment that executes JavaScript code server-side. It is used in this project to build the backend server using Express.js, handling API calls and notifications.

Why Node.js?

Asynchronous and Non-Blocking: Multiple requests (e.g., notifications, file uploads) are handled effectively without blocking the server.

Rapid Development: Rapid prototyping and deployment of the backend API are supported.

Large Ecosystem: npm packages (e.g., mongoose, firebase-admin) are used for extra functionality.

Event-Driven: Allows for scalable server operations, which is crucial in managing group-based notifications.

MongoDB

MongoDB is a NoSQL database that stores data in a flexible, JSON-like data format named BSON. It is used in this project to store notification data and user data persistently.

Why MongoDB?

Flexible Schema: Handles dynamic data structures, i.e., various kinds of notifications.

Scalability: Supports horizontal scaling for handling increasing data loads.

Node.js Integration: Works well with Mongoose for easy data modeling and querying.

Performance: Read/write optimized for a real-time application.

Visual Studio Code

Visual Studio Code (VS Code) is a highly extensible source-code editor developed by Microsoft, with debugging support, IntelliSense, and Git version control. It is used for frontend (Flutter) and backend (Node.js) development.

Why Visual Studio Code?

Cross-Platform Support: Runs on Windows, macOS, and Linux, matching the project's multi-environment demands.

Extensions: Supports extensions for Flutter, Dart, and Node.js plugins, enhancing development speed.

Debugging: Powerful debugging capabilities for debugging code.

Git Integration: Simplifies version control and collaboration.

Postman

Postman is an API development and testing tool that provides an easy way to construct, test, and document APIs. It is used to test the backend APIs (e.g., file upload, notification endpoints).

Why Postman?

API Testing: Allows quick testing of endpoints like $_apiUrl/storage/upload and $_apiUrl/notifications/send.

Collaboration: Enables simple sharing of API specifications with the team.

Automation: Facilitates automated testing of API responses.

Ease of Use: Provides a user-friendly interface for debugging API calls.

JWT (JSON Web Token)

JWT is an open standard for transferring claims between parties in a JSON object securely. In this project, it's being used for user authentication and authorization.

Why JWT?

Secure Authentication: Offers secure login and token-based authentication for protected routes.

Stateless: Enables the server to authenticate users without maintaining session information.

Compact: Can be easily sent in HTTP headers for API calls.

Integration: Easily integrates with Node.js and Flutter for authentication workflows.

Cloudinary

Cloudinary is a cloud-based service that provides strong features of image and video management, like uploading, storing, processing, and distributing media in an efficient manner. Cloudinary has been used within this project to handle the media upload of both hardware equipment photos and individual chat messages.

Why Cloudinary?

Media Upload & Storage: Facilitates efficient and secure upload of media content and profile images in individual chats.

URL-Based Access: Provides easy access to uploaded images using secure URLs, which facilitates rendering of the image within the UI.

Optimization: Automatically optimizes images for fast loading and improved performance on any device.

Integration: Easily integrates with Node.js and Flutter using SDKs and REST APIs to manage file upload and deletion.

EduSync as a Mobile Application

1. Authentication and Validation

When you enter the application, this interface is displayed(Authentication and Validation).

Because EduSync is regarded as a middleman between the students, each user must register on the platform in order to identify themselves and have access to all of its features. The user needs to register their name, email, and password.

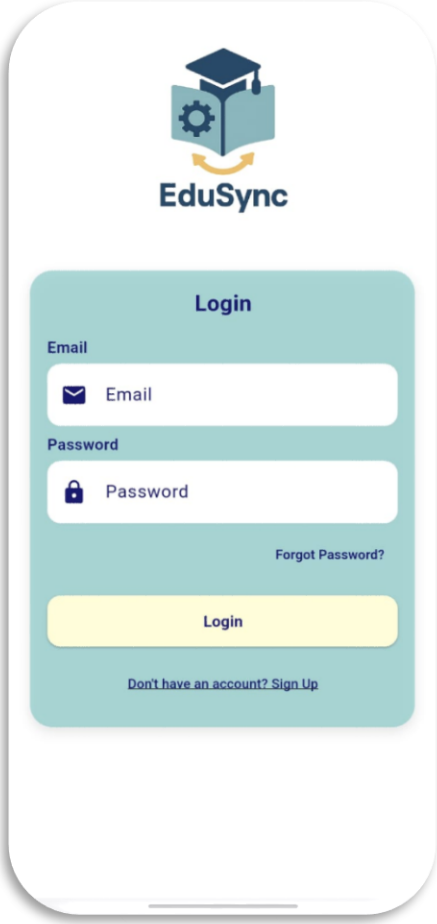


Figure 1 Login Page Overview

Signup

You must enter your full name, job title (Doctor, Student, External), email address, user ID if only Doctor or Student, and password.

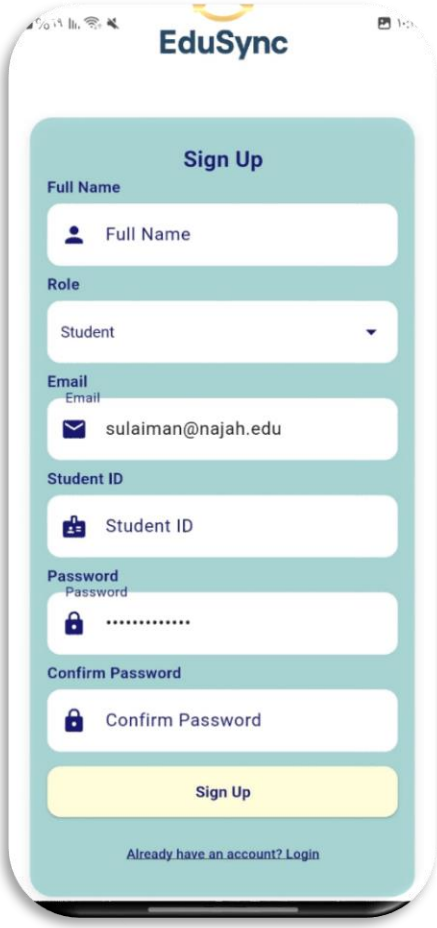


Figure 2 Sign-Up Page

Sign up Validation

The system performs validation checks at registration to encourage data accuracy and completeness. When the user enters invalid or incomplete data—such as a required field left blank, badly formatted email, or weak password—the app shows a clear error message specifying the issue. The system also dynamically validates the email format based on the user type chosen. Those who select "Doctor" or "Student" must provide an email with the ending "@najah.edu", while those who select "External" must provide an email with the ending "@gmail.com". If the email format does not match the type selected, the system will alert the user to this. Moreover, if the email provided is already in the database, the user is notified that the email has been taken. These validations provide data consistency and allow smooth user experience

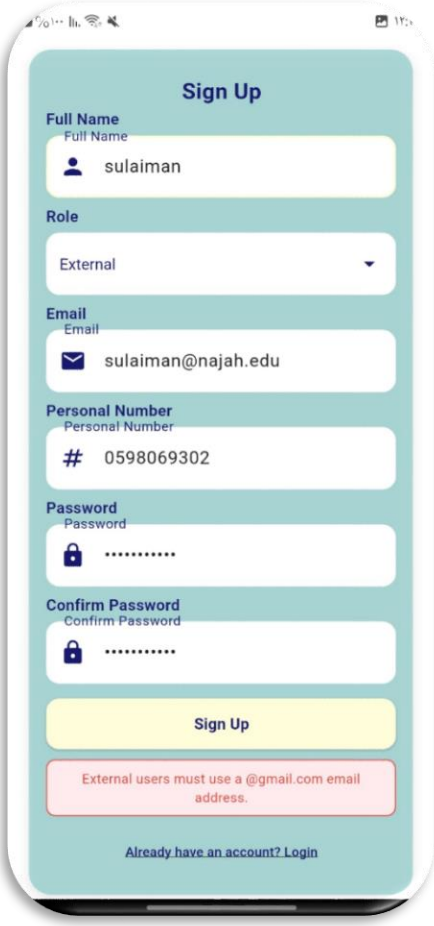


Figure 3 Sign up Validation 2

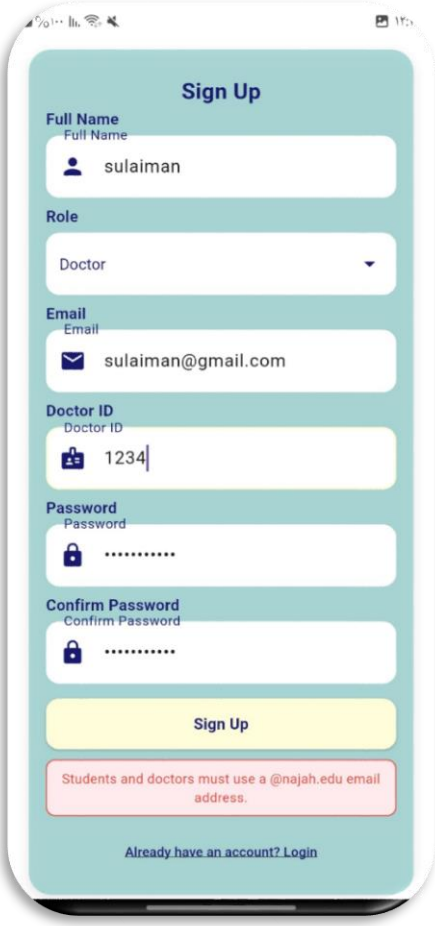


Figure 4 Sign up Validation 1

Login

In order to log in to the website, you must enter certain details such as your password and email, which you have already registered with them.

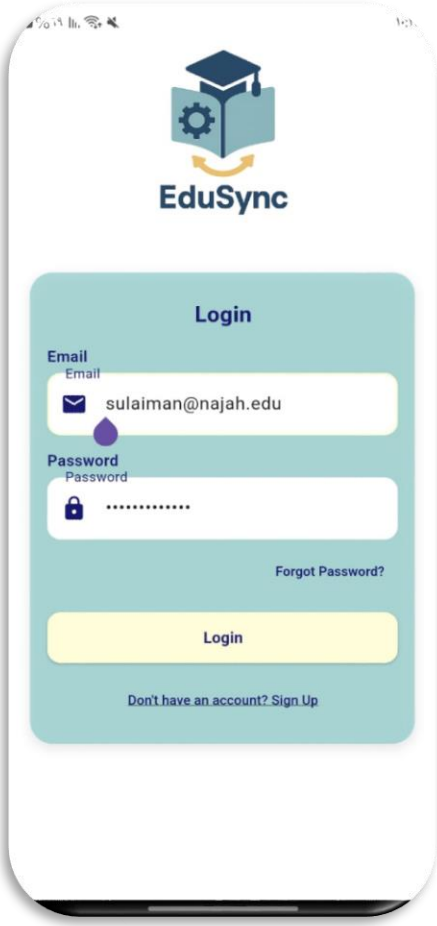


Figure 5 Login Page

Login Verification

When the user attempts to log in, the back-end checks the credentials; if incorrect, the user is alerted, and if they attempt to log in with a non-existent email address, he is also notified.

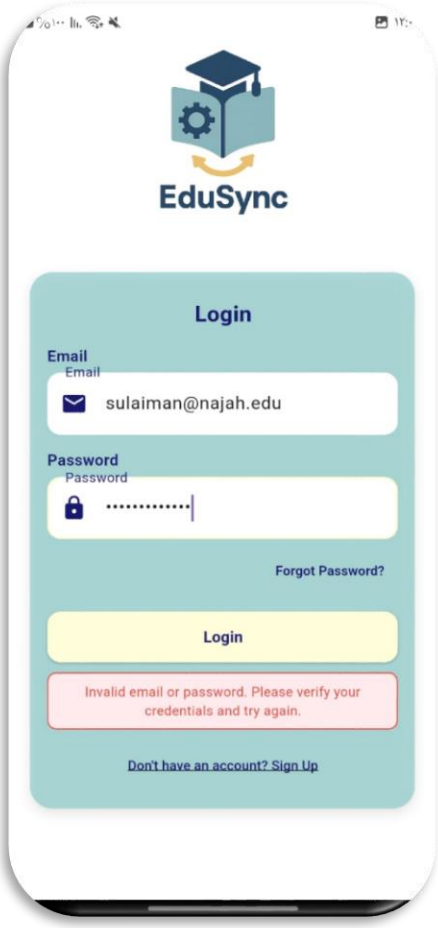


Figure 6 Login Verification

Forget password

When the user wants to log in and clicks on a forgotten password, they will choose the email address to which a code will be sent to use to reset the password.

On pressing the "Send Reset Link" button, a confirmation message appears, and a reset token is sent to the selected email address. At the same time, the system takes the user to the password reset page automatically.

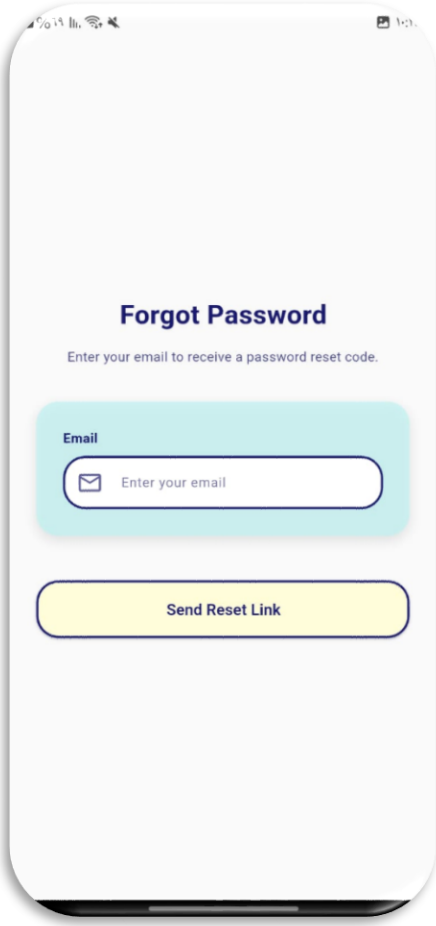


Figure 7 Forgot Password Page

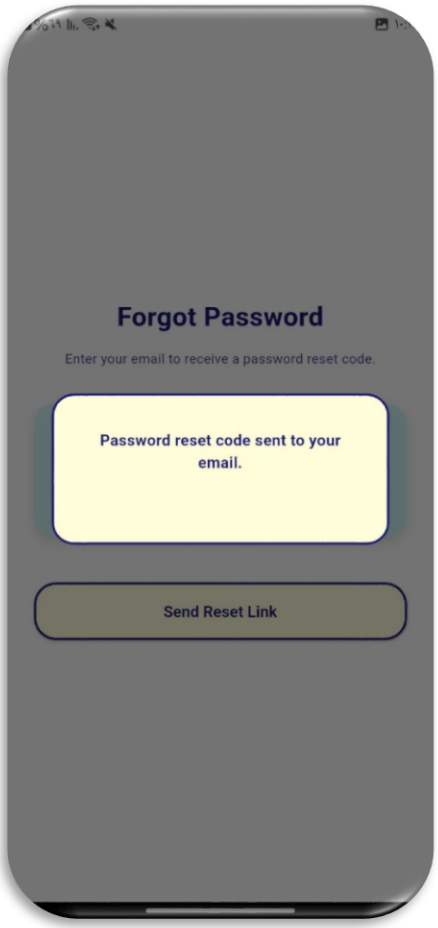
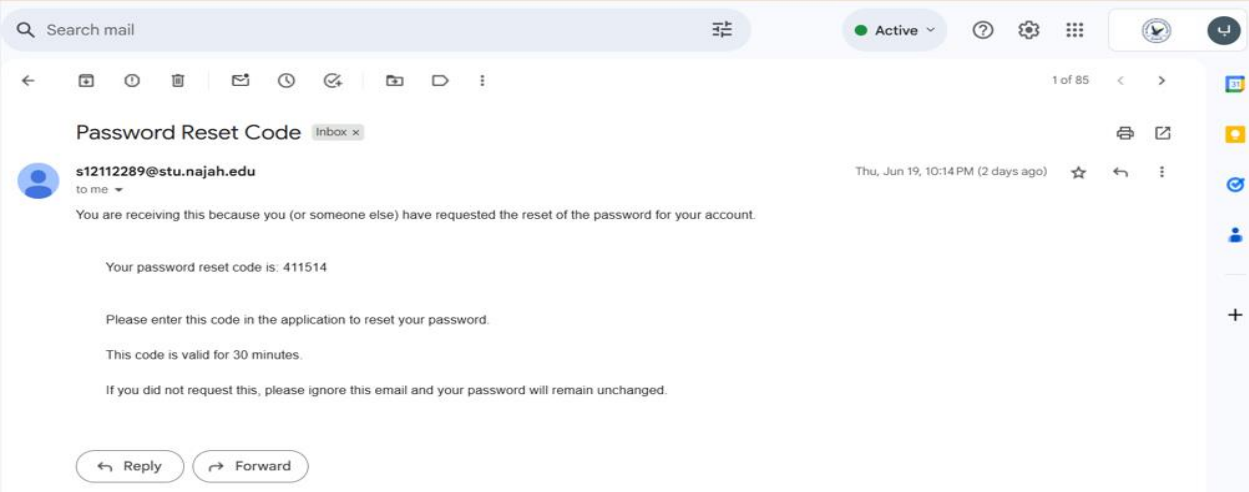


Figure 8 Successful send code

Reset password

When the user wants to reset the password, they must enter the code sent via email and choose a new password.



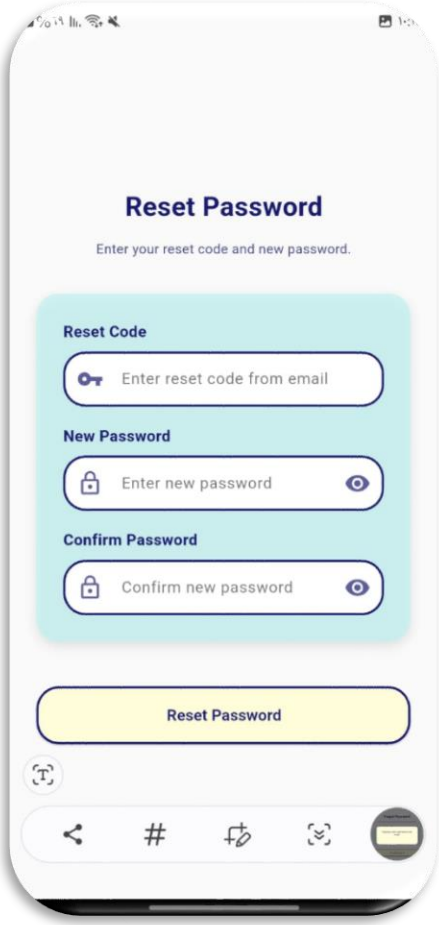


Figure 9 Reset Password Page

2. User Profile

On the Home screen, there is a menu button (three horizontal lines) which, when clicked, reveals the user account options like My Account, My Activity, Saved Posts, and Log Out.

Under My Activity, users can further click to see their My Posts, My Likes, and My Comments, so that they can easily keep track and manage their activity within the app.

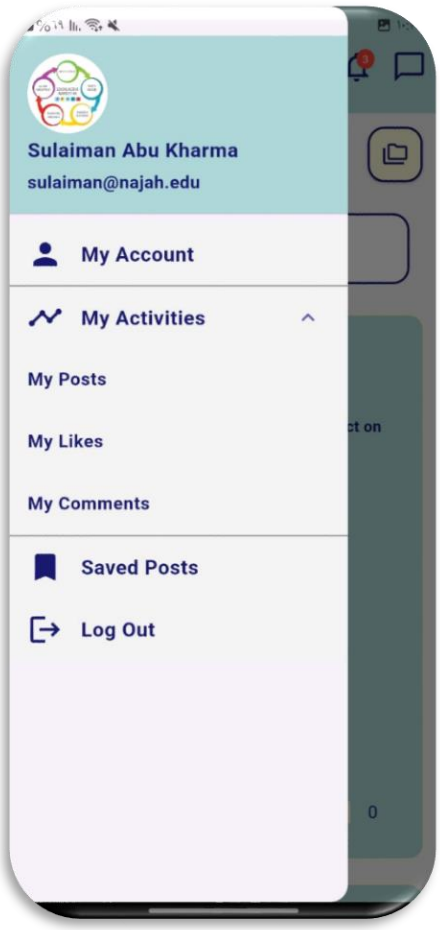


Figure 10 My Activities

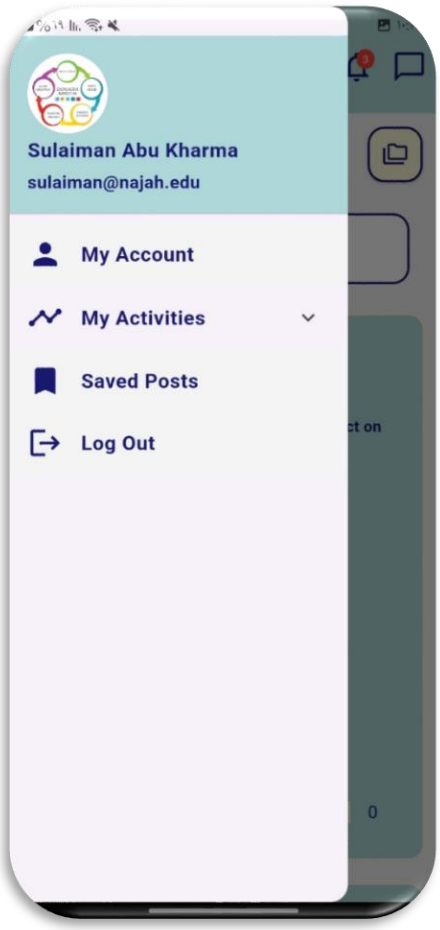


Figure 11 Personal information

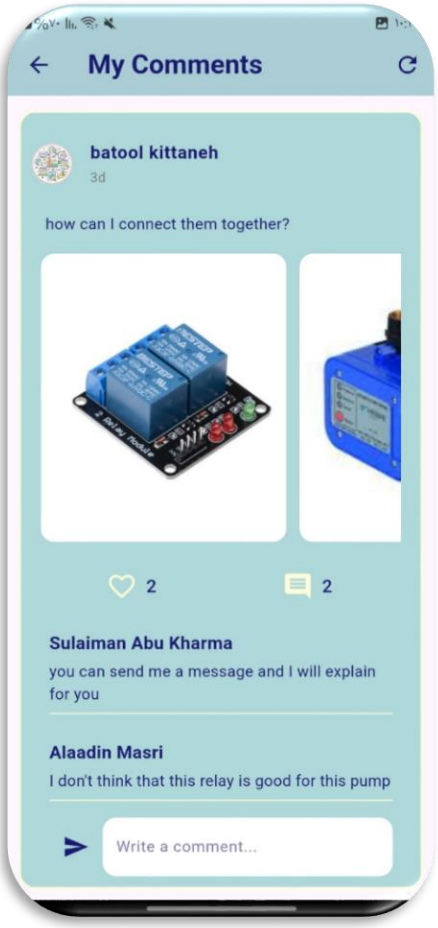


Figure 12 My Comments

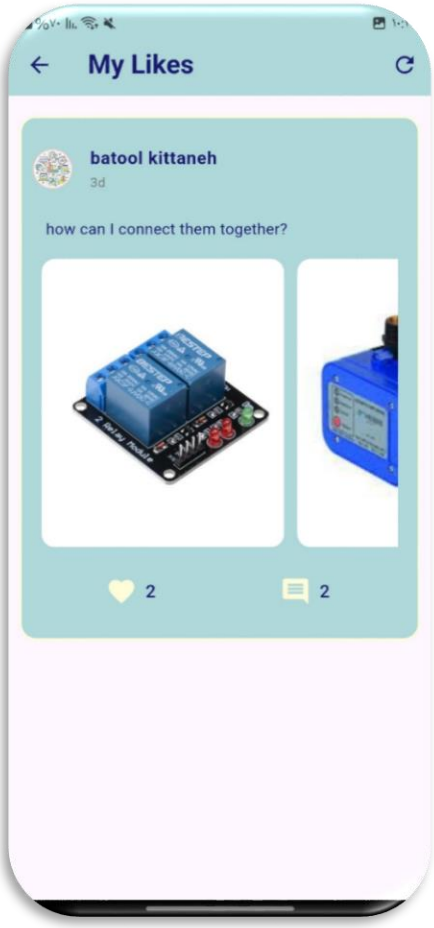


Figure 13 My Likes

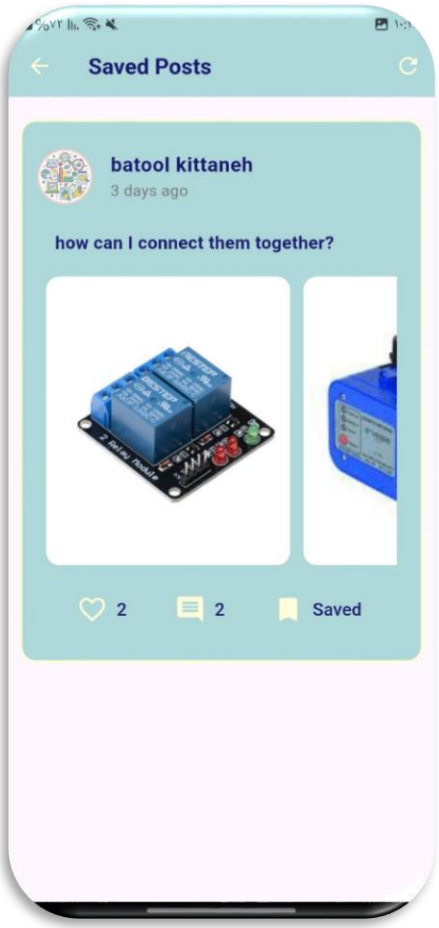


Figure 14 Saved Posts

Figure 15 My Posts

There is a personal profile for each registered user. The user's personal information—full name, password, and profile picture—is displayed when the user clicks My Account. The user can easily change any of these from this area.

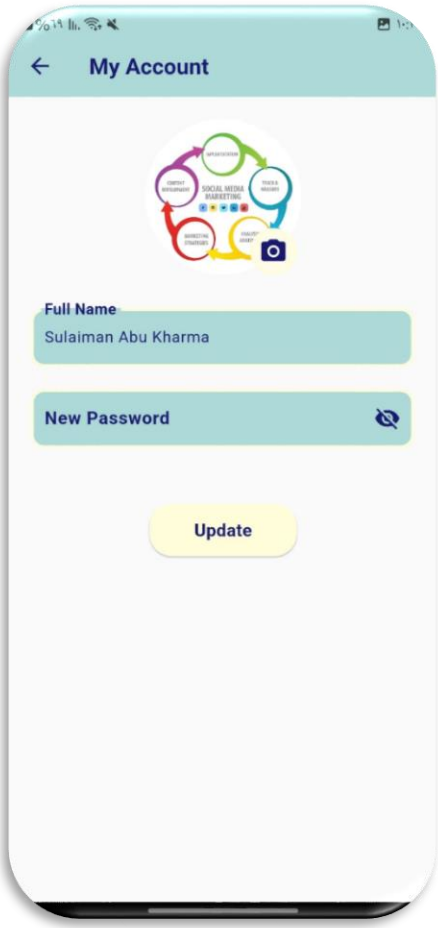


Figure 16 My Account

When the user wants to log out of their account, they can simply tap on the "Log Out" option from the menu to securely exit the application

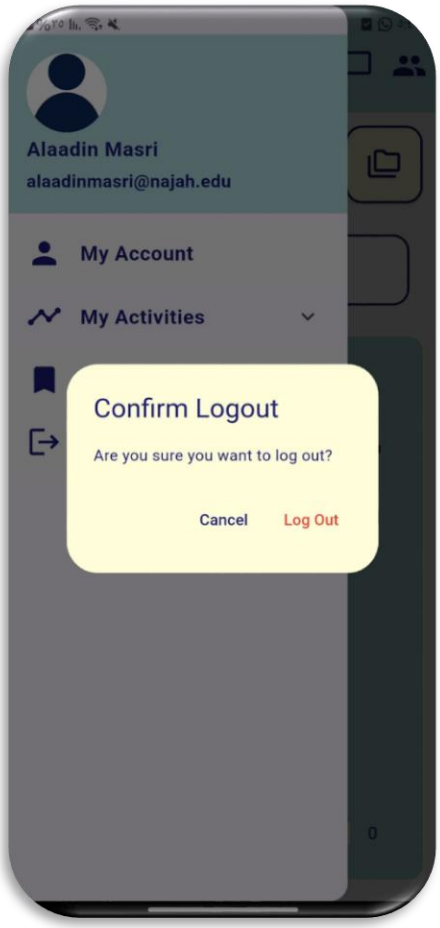


Figure 17 Logout

3. Home Screen

The home screen serves as the main hub of the application. Users can post updates and share their thoughts. They can also engage with posts by liking or commenting. The screen features navigation buttons at the top, including "Hardware Equipment", "AI Chatbot," and "Pre-Projects," each directing users to their respective pages. A bell icon provides access to the notifications screen, while a chat icon leads to the chat interface. Additionally, a search bar allows users to find specific users by name, enhancing connectivity.

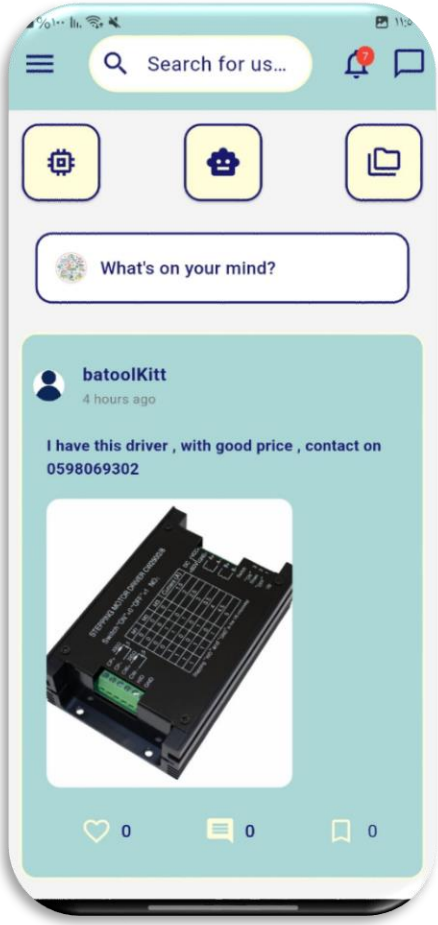


Figure 18 Home Screen Page

When searching for a specific user, the system performs real-time filtering, updating the results dynamically with each character typed in the search field.

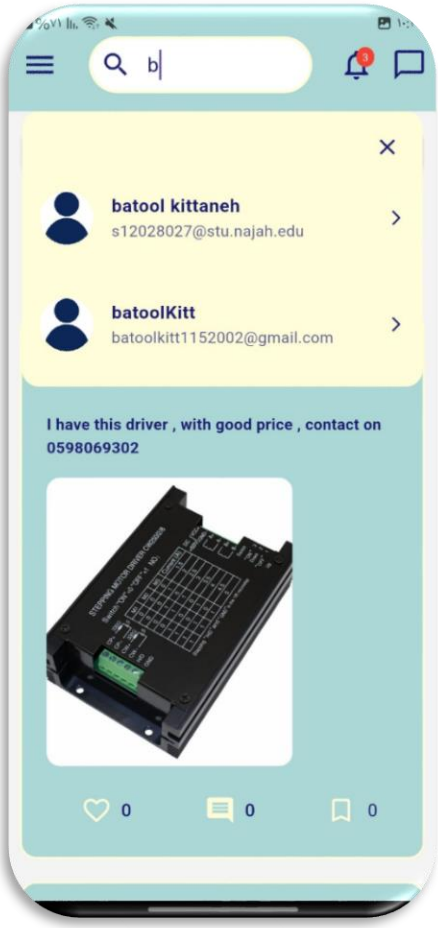


Figure 19 Search for a user

When searching for a user, if the person is a Doctor or Student, the user can initiate a chat with them directly through the application. However, if the person is marked as External, only their phone number is displayed so that communication can be made externally.

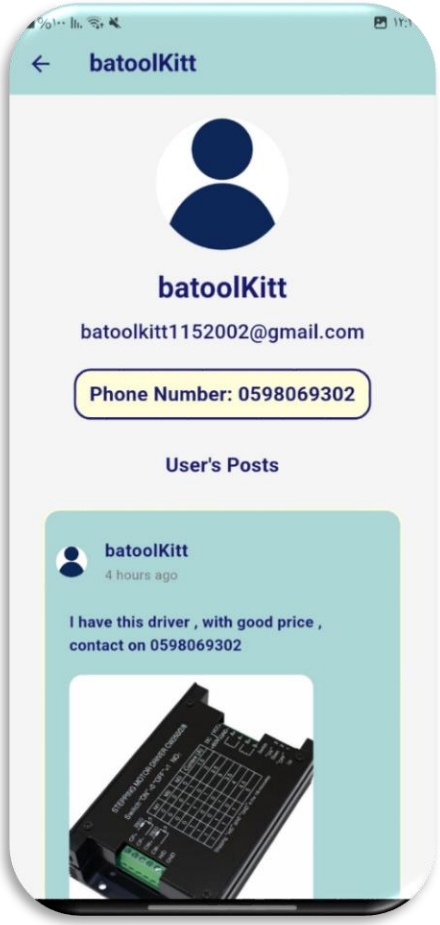


Figure 20 Search for an External

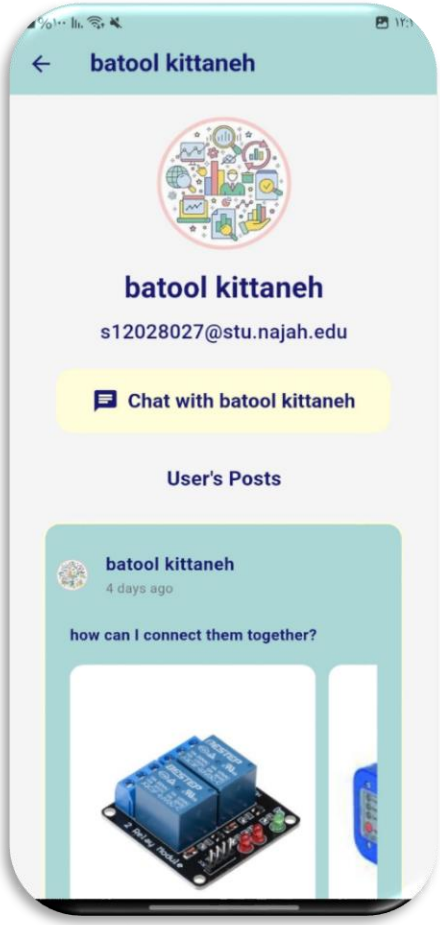


Figure 21 Search for a Student

In the Home screen, there is a dedicated button for external account access requests, which is only visible to the Admin. Through this button, the Admin can view, accept, or reject pending access requests submitted by external users.

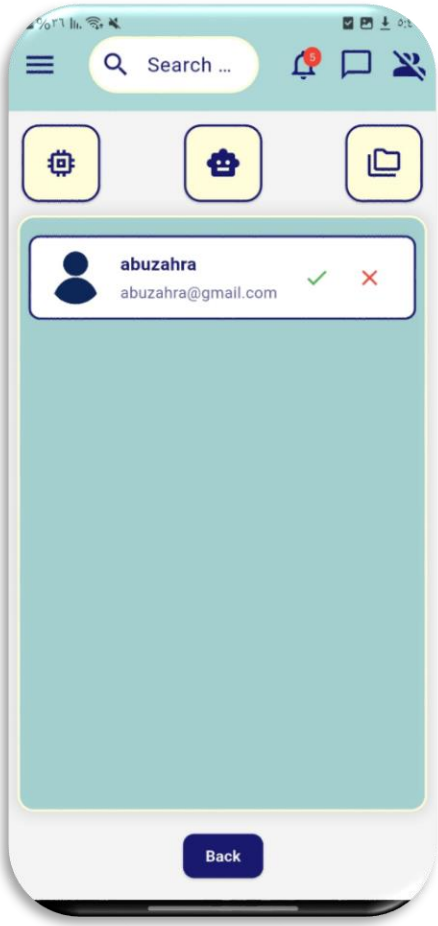


Figure 22 External user requests

Once the request is approved, the external user gains access to a dedicated page where they can submit and publish their hardware components within the system.

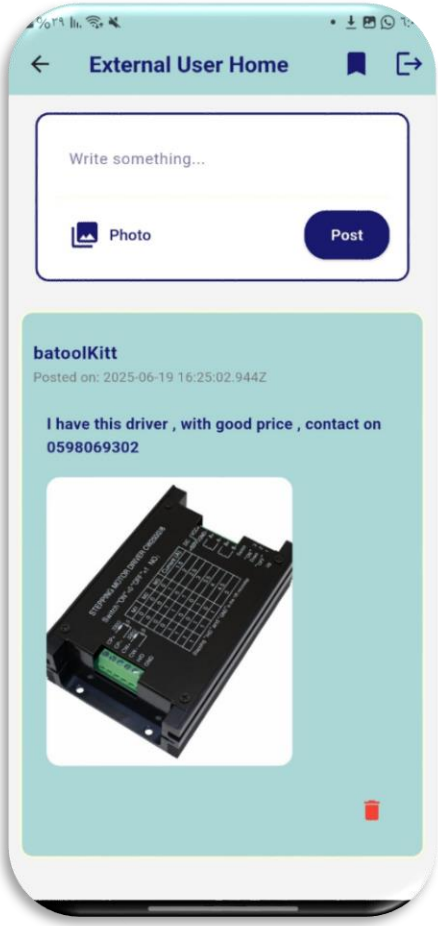


Figure 23 External post page

4. Hardware Equipment

allows students to manage and share their equipment within the app. Each student can post their own equipment, specifying its name, price, and quantity. Only the student who posted it can delete it, ensuring ownership control. Each equipment listing includes a chat option, allowing users to communicate directly with the owner to communicate or inquire. They can also quickly search for the name of the equipment.

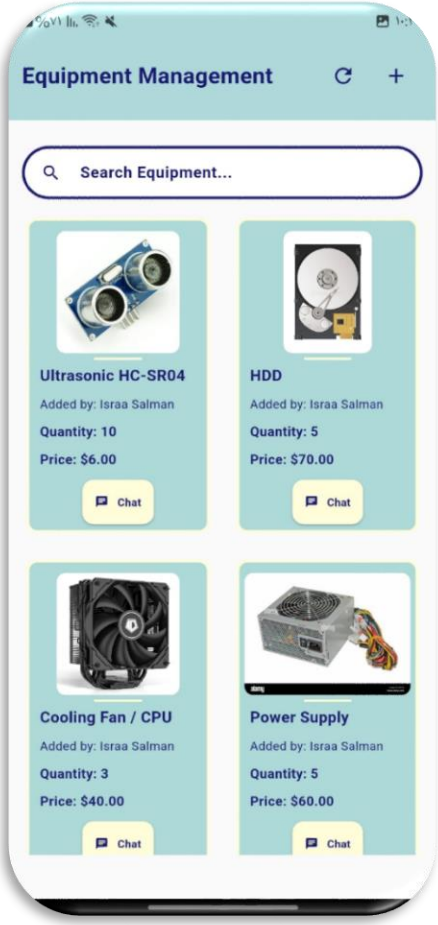


Figure 24 Hardware Equipment Page

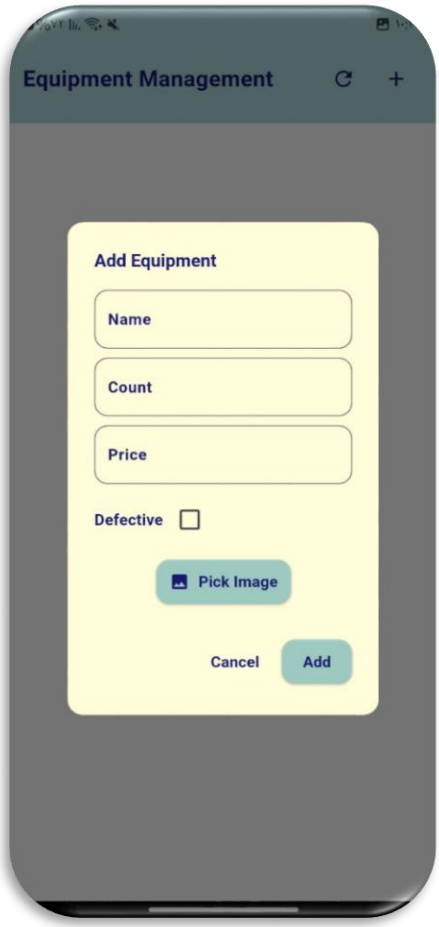


Figure 25 Add equipment

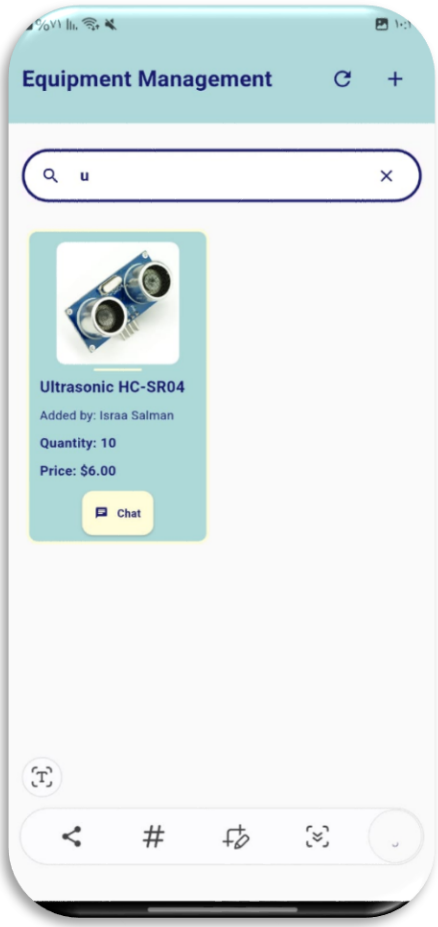


Figure 26 Search for equipment

5. AI Chatbot Page

This page allows the user to interact with an AI-driven chatbot using a user-friendly chat interface. It has a chat history pane on the left to maintain previous conversations, a "New Chat" button to create new sessions, and a main chat window where the user can type and send messages.

Figure 27 AI Chatbot Page

When the user taps on the three-line menu icon, a sidebar appears displaying the chat section. From this menu, the user can create a new chat by specifying a title, as well as view a list of previous chats.

Figure 28 AI Chatbot History

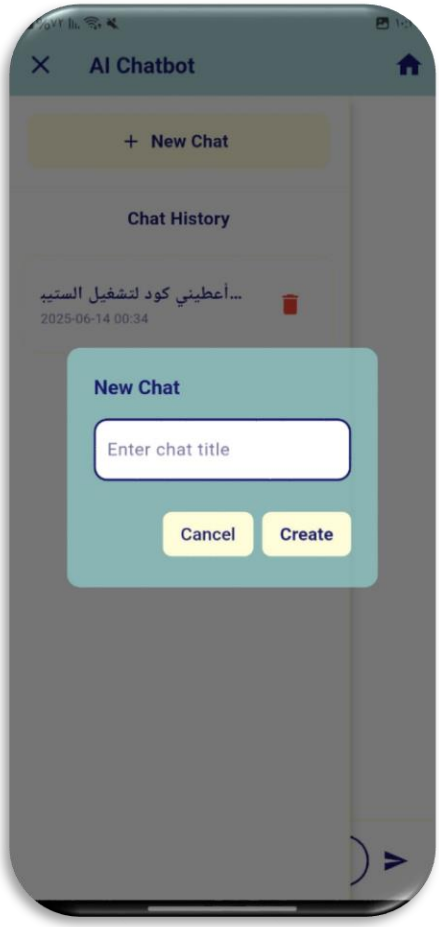


Figure 29 New AI Chatbot

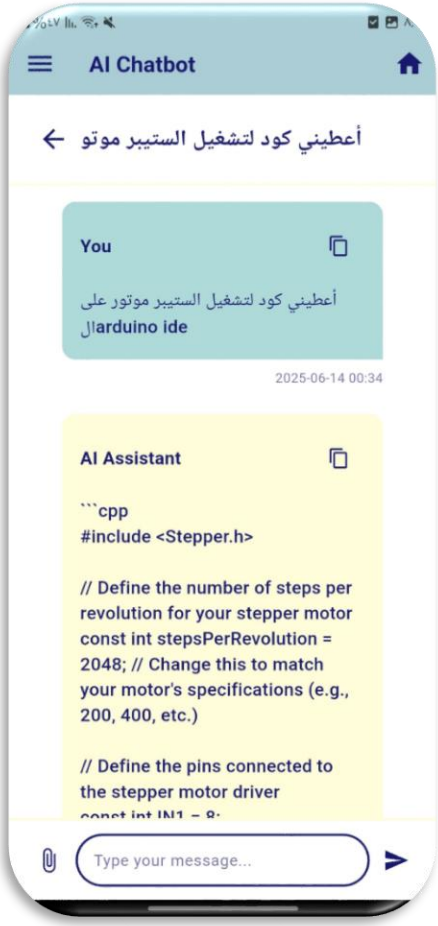


Figure 30 Chat with Ai

6. Previous Projects

This page displays a list of previously submitted projects and contains complete metadata for every project. Projects can be filtered by project type, year, and supervisor, or searched based on project abstract. Every project card displays the project title, year, supervisors, team members, and download link for each submitted file. This page enables users to browse and view previous work easily for reference or evaluation.

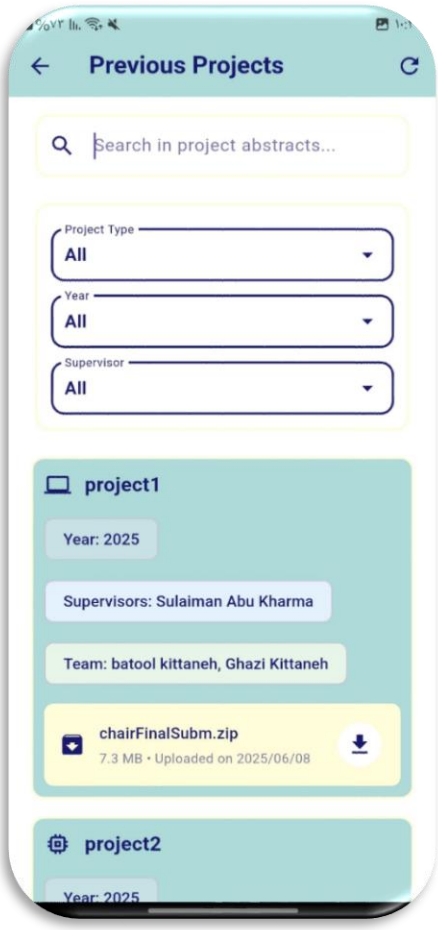


Figure 31 Previous Projects Page

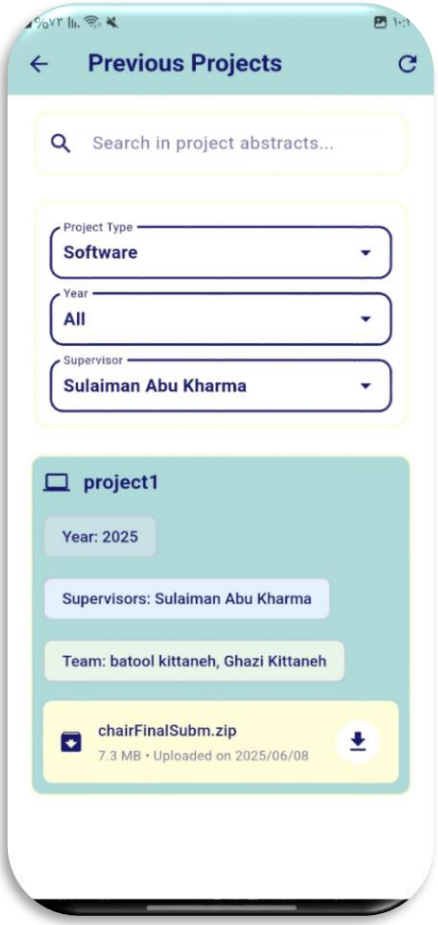


Figure 32 filtering previous projects

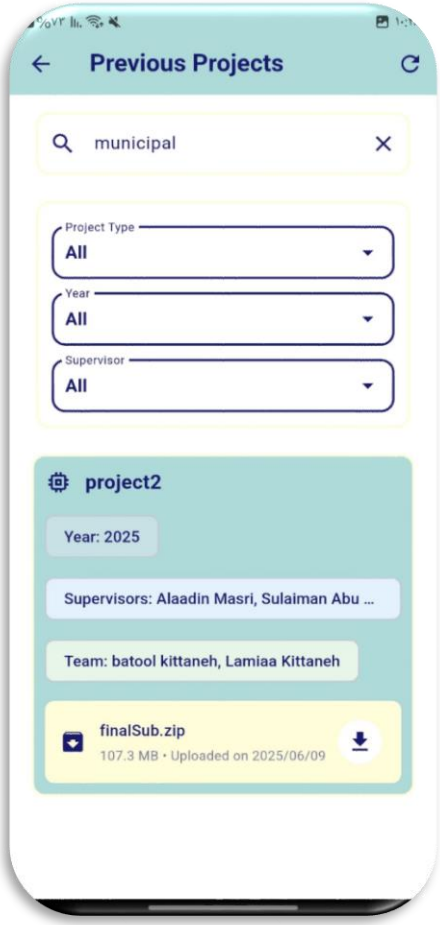


Figure 33 search in previous projects

7. Chat Screen

This page displays the user's chat window and contains two sections: Private Chats and Groups. It displays the name of the person, their profile picture, a preview of the last message or content exchanged, and the time it was delivered. The application allows users to communicate with supervisors or group members conveniently in a structured and organized format.

Figure 34 Chat Screen Page

Private Chat Page

The direct private chat feature allows members to converse in a straight one-on-one conversation with each other. Members can send messages and file attachments such as images or ZIP attachments. Effortless communication is made possible by this feature through which students and superiors can share assignments and discuss project information. The system carries an interface that provides timestamps and includes an option for the user to delete messages. In addition, messages are also distinguished visually by color — the sender's messages have a different color than the receiver's, and both are aligned to the right and left, respectively, so that it becomes easier to read and comprehend the dialogue.

Figure 35 Private Chat Page

Group Chat Page

When creating a new group chat, the user is prompted to enter essential information

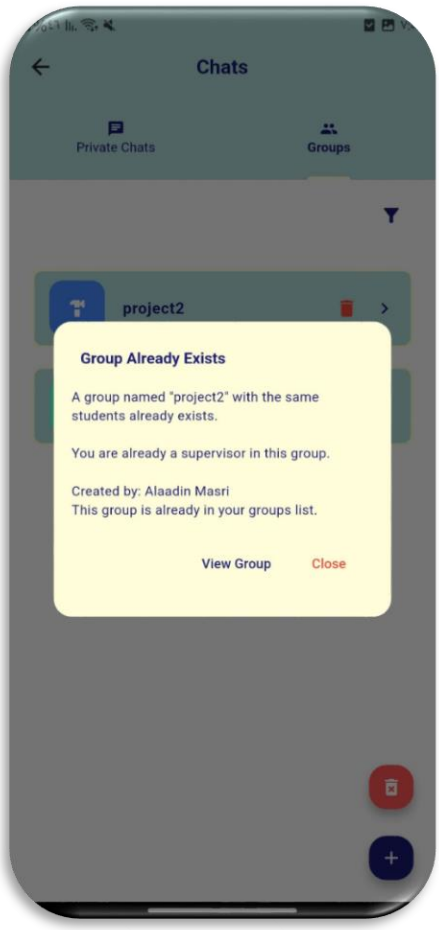


Figure 36 When a Group Already Exists

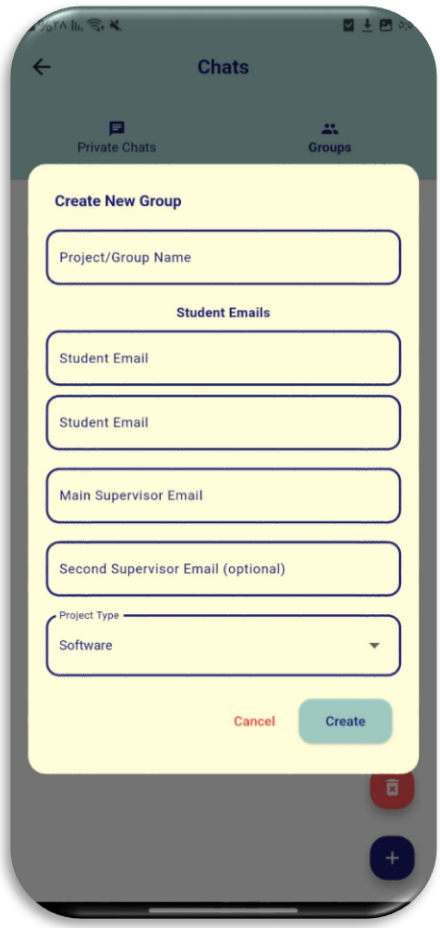


Figure 37 Creating a group chat

Doctors have the ability to create new chat groups as well as delete existing ones within the application.

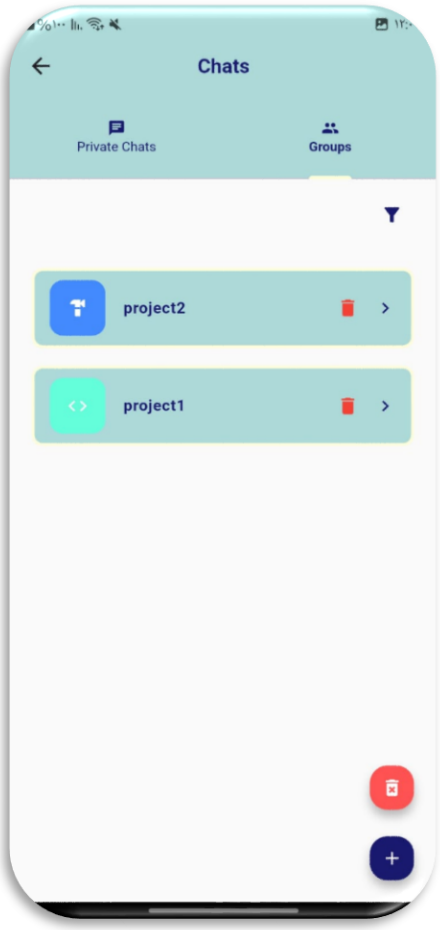


Figure 38 Group Screen Page

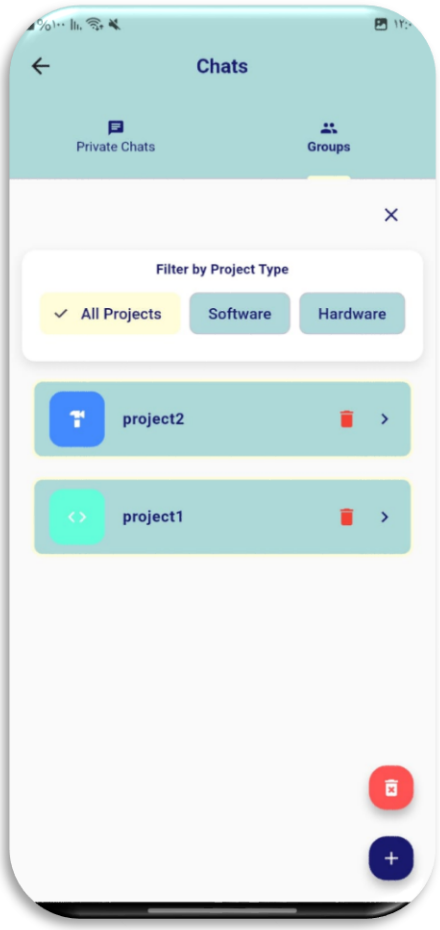


Figure 39 Group Screen filtering

The group chat feature enables multiple users to converse and communicate in a shared conversation window. It enables the sending and receiving of text messages, making it ideal for group project management or discussion forums. The window displays each participant's name clearly along with the message and timestamp.

Additionally, the group chat has two handy tools:

- File Upload – Allows users to share documents, images, and project files via the chat.
- Smart Calendar Integration – Helps users schedule and organize meetings directly from the chat, facilitating team coordination and productivity.

This capability promotes real-time collaboration among members in an organized and interactive space.

Figure 40 Message control

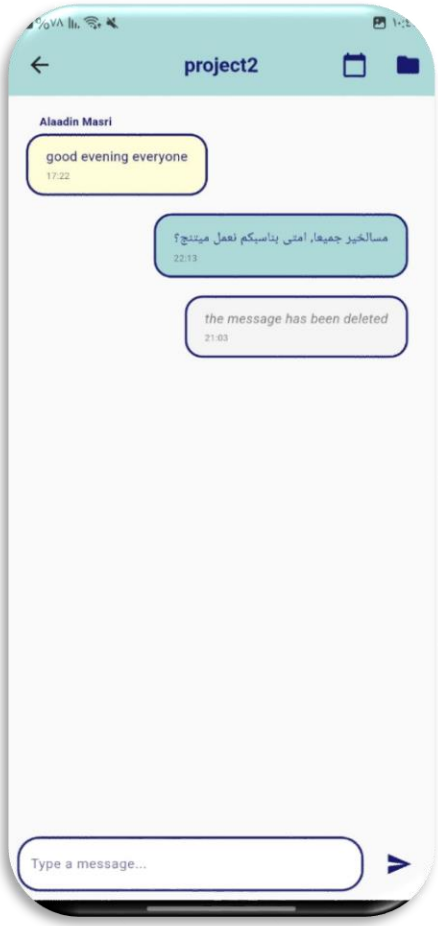


Figure 41 Group Chat Page

8. Smart Calendar Page

The Smart Calendar feature provides an interactive and automatic interface for planning and scheduling team meetings effectively. It is possible to allocate the meeting title, configure the meeting type (physical or online), and choose the date and time. When scheduled, the meeting is shown with the status Scheduled.

Status Lifecycle:

Scheduled: Default status when the meeting is scheduled.

In Progress: Automatically set when the meeting starts.

For online meetings, a Google Meet link is generated and sent at the start.

If no one joins in 10 minutes, the meeting status is cancelled.

For face-to-face meetings, the meeting is tagged as In Progress for 30 minutes. If no one mentions they are attending within this time, it's also tagged as Cancelled.

There is a Reactivate option for face-to-face meetings to tag attendance if it has been missed, turning the status back to In Progress.

After the meeting time elapses, the status gets automatically changed to Completed.

Other Features:

Each meeting defaults to one hour.

Users can add notes to specify the agenda or purpose of the meeting.

Users can adjust the meeting time or reschedule the meeting before it starts.

The smart system facilitates meeting management with ease and provides automatic status reports to reduce manual effort and confusion.

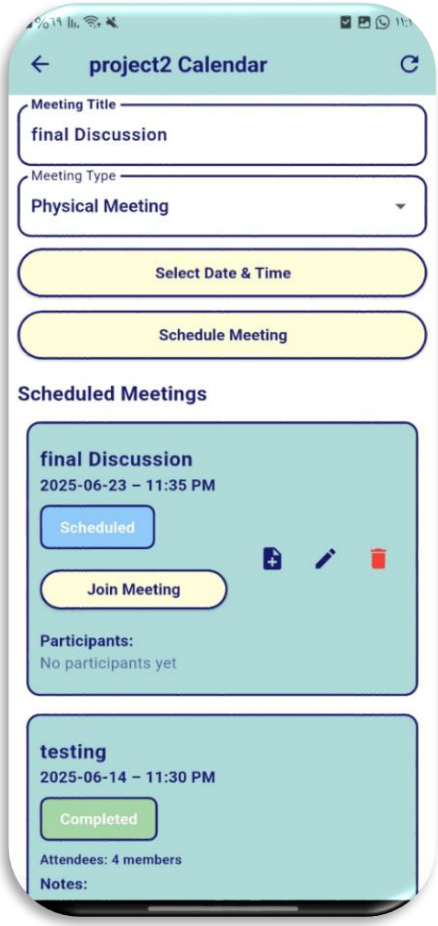


Figure 42 Join Meeting

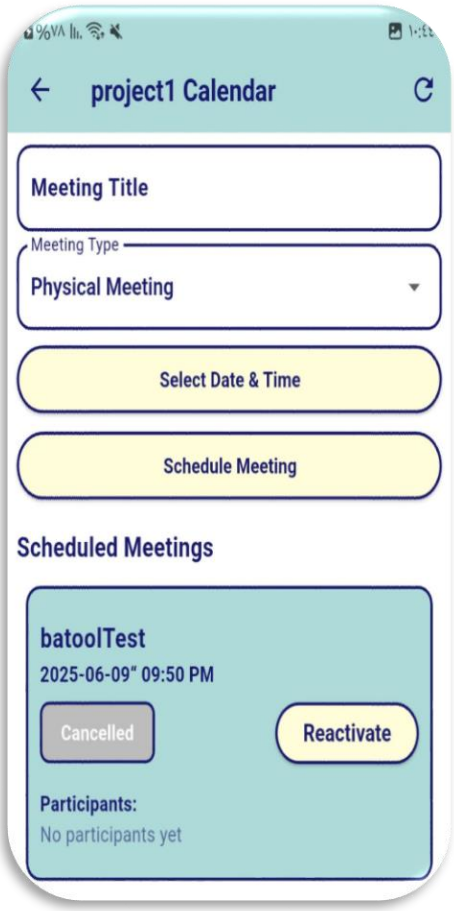


Figure 43 Smart Calendar Page

Figure 44 Meeting Type

When selecting Date & Time

Figure 45 Selecting the hour

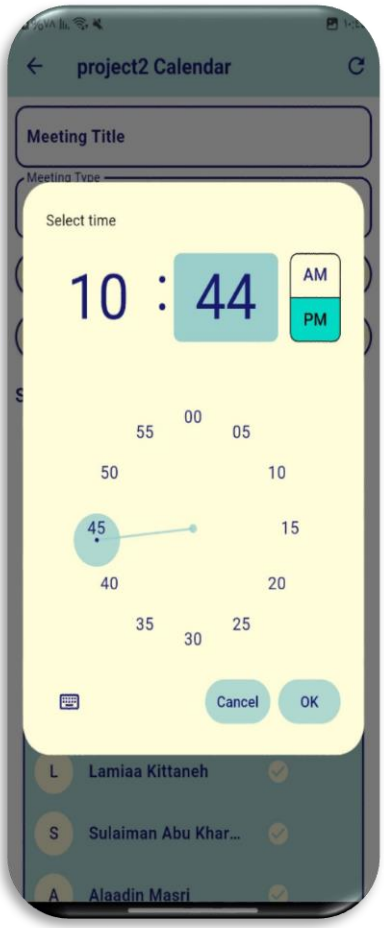


Figure 46 Selecting a minute

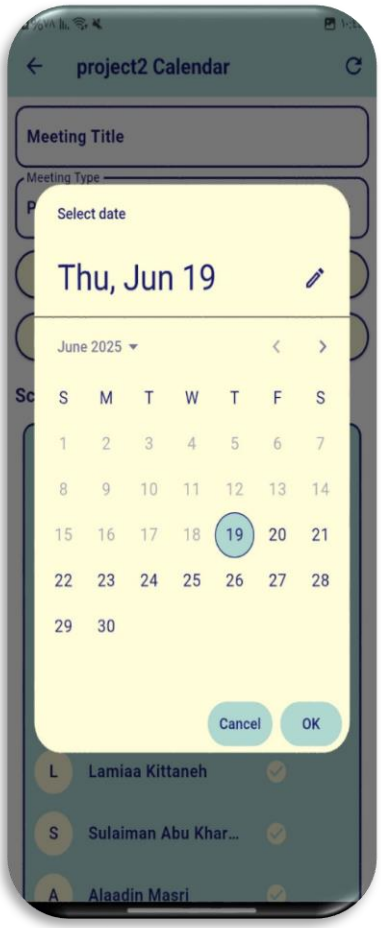


Figure 47 Selecting Date

Figure 48 Add note

Figure 49 The meeting was completed

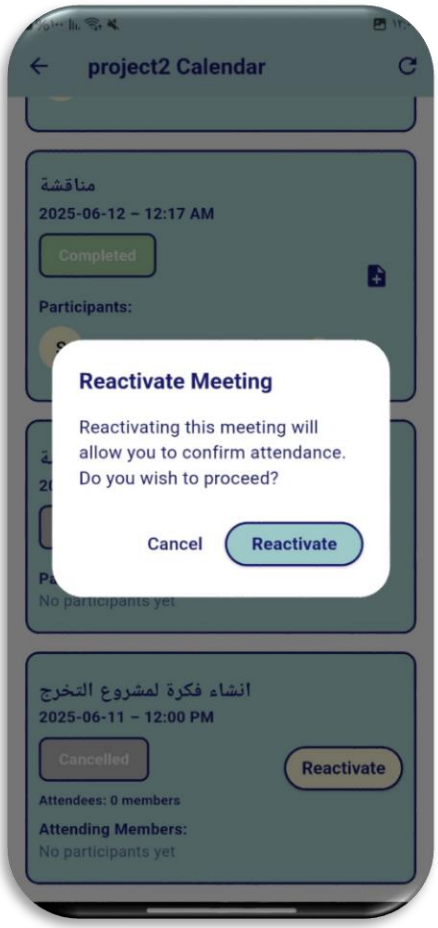


Figure 50 Reactivate Meeting

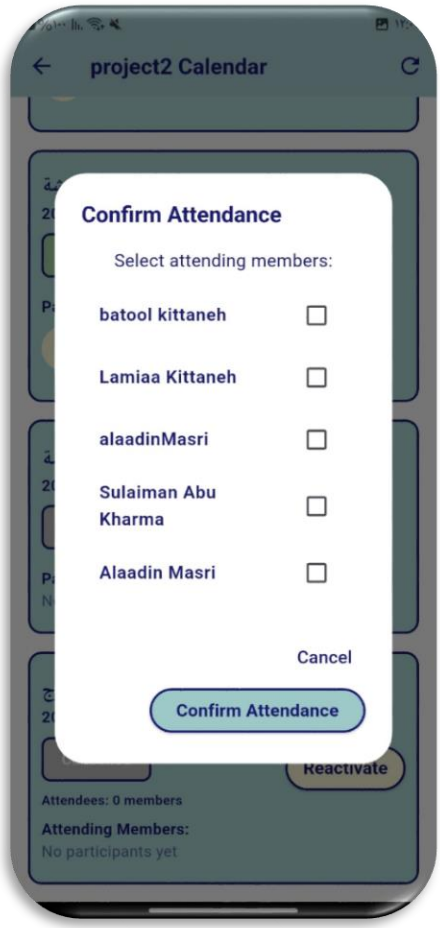


Figure 51 Confirm Attendance

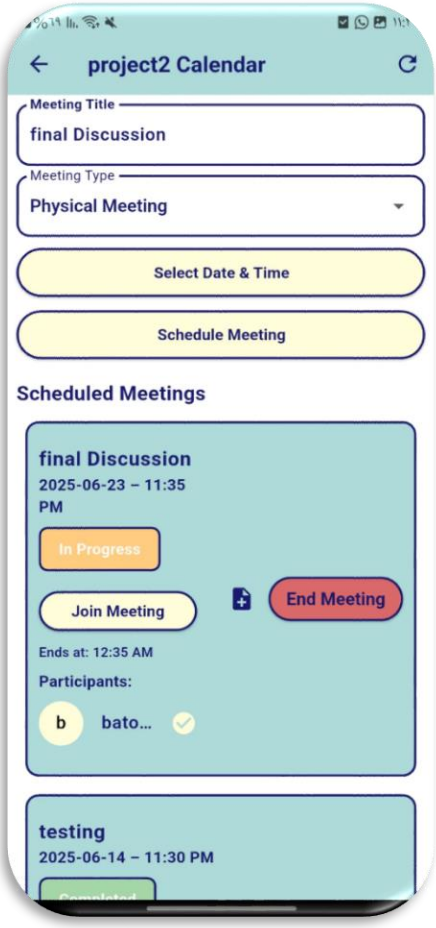


Figure 52 End Meeting

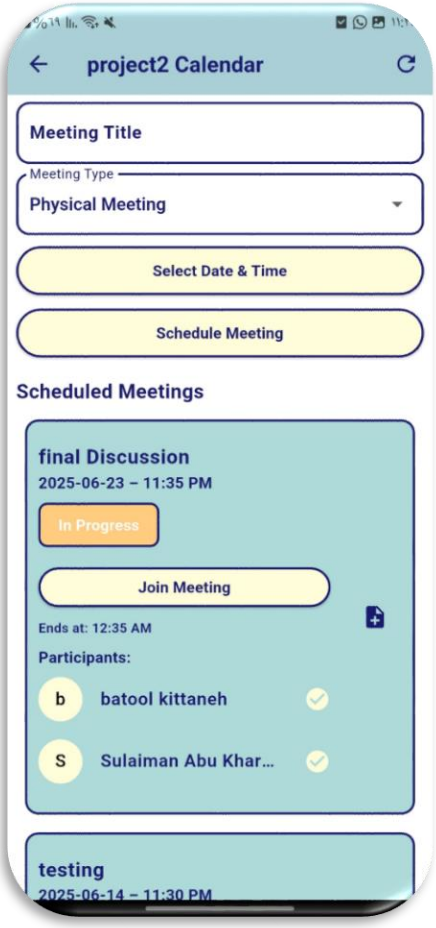


Figure 53 Meeting In Progress

9.Group Files Page

The group file upload interface allows users to share and upload project files in a shared space. The files are uploaded mainly in ZIP format and are displayed with the following details:

File name, Upload time, date, and Uploader's email address

This option keeps members organized and allows all members to have the most updated shared resources in a central location.

Figure 54 Group File Page For Student

Figure 55 Group File Page For Doctor

10. Notification Page

It displays a list of notifications with details such as the type of action, the user involved, and the time of occurrence. A notification badge indicates the number of unread alerts, ensuring users can quickly review and stay updated on group activities and interactions

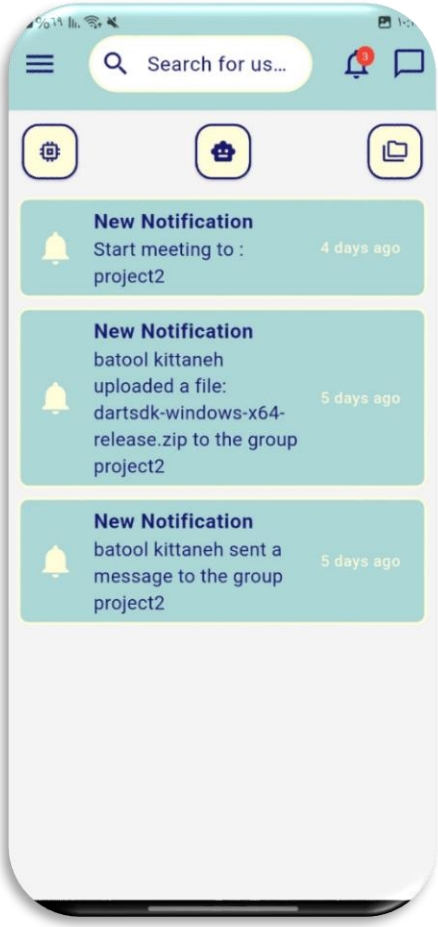


Figure 56 Notification Page

EduSync as a Website

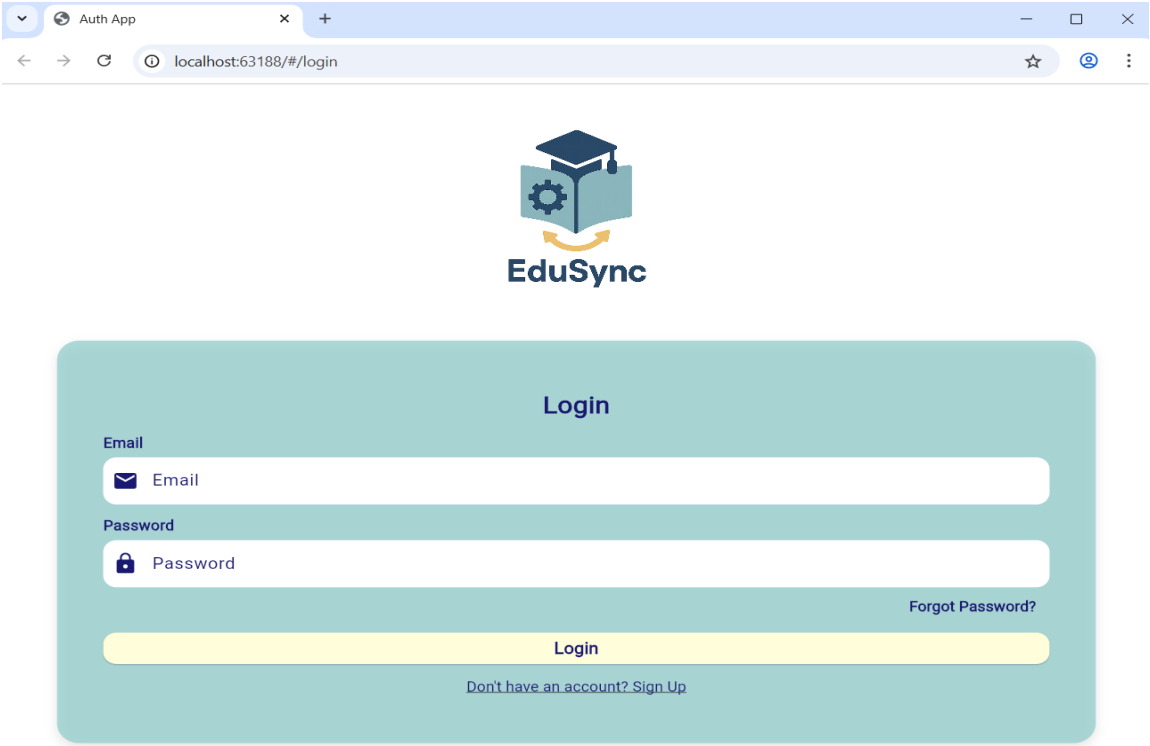


Figure 57 Login Page-Web

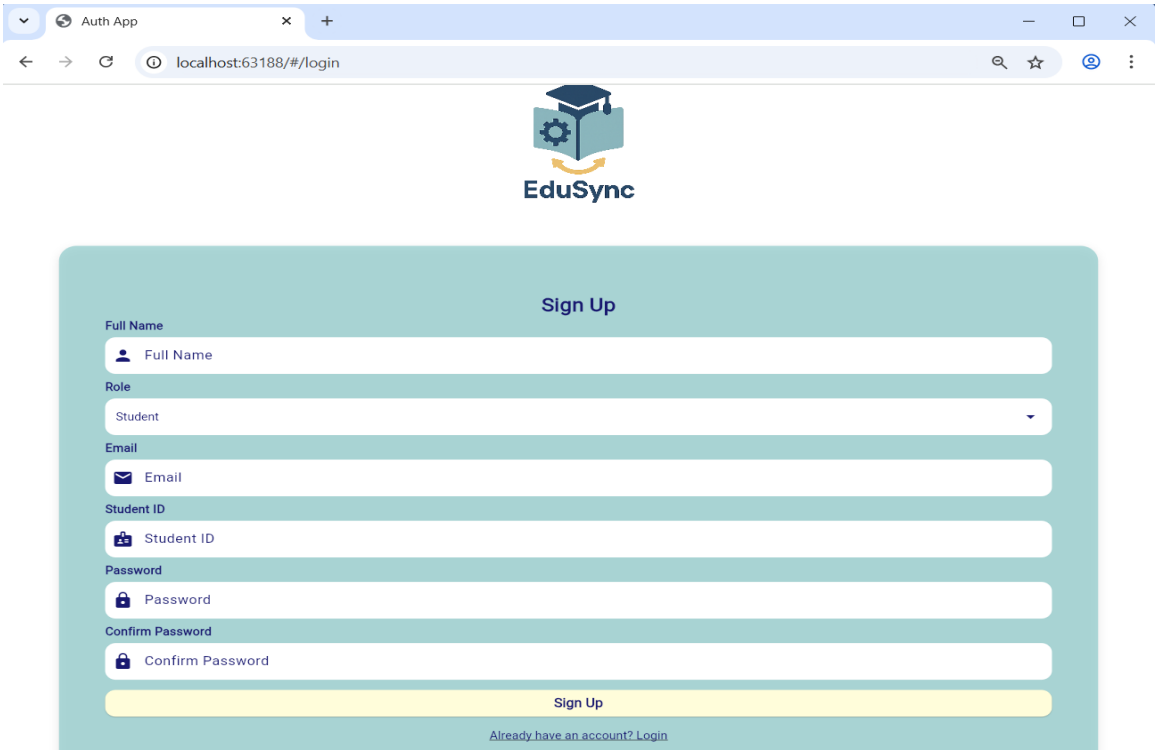


Figure 58 Signup-Web

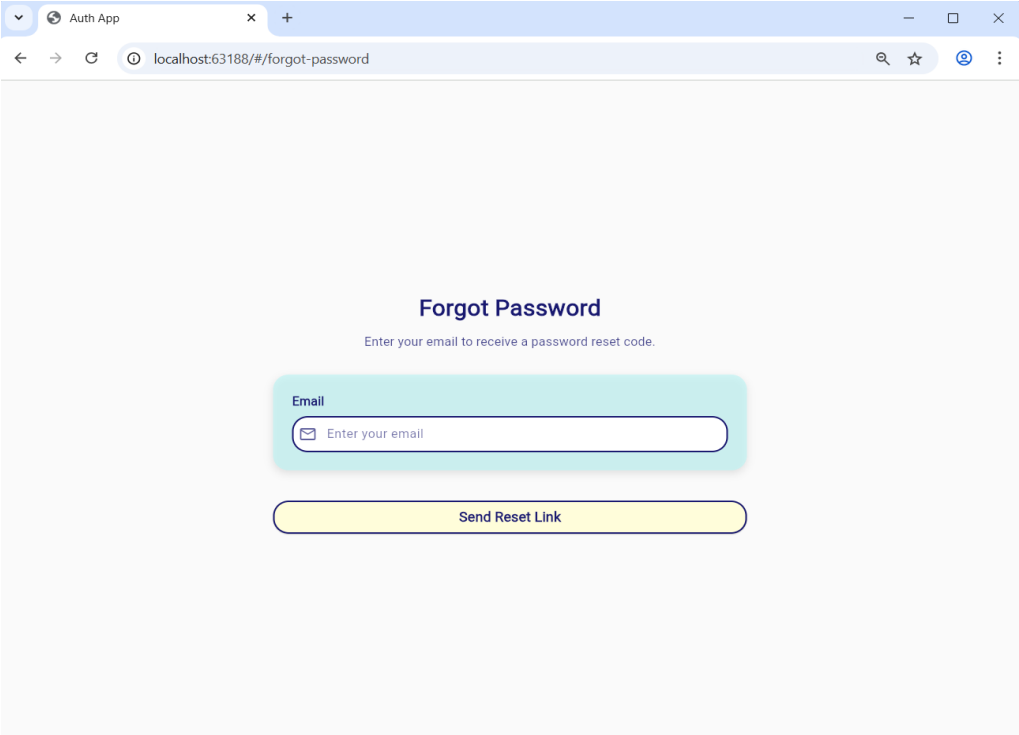


Figure 59 Forget Password-Web

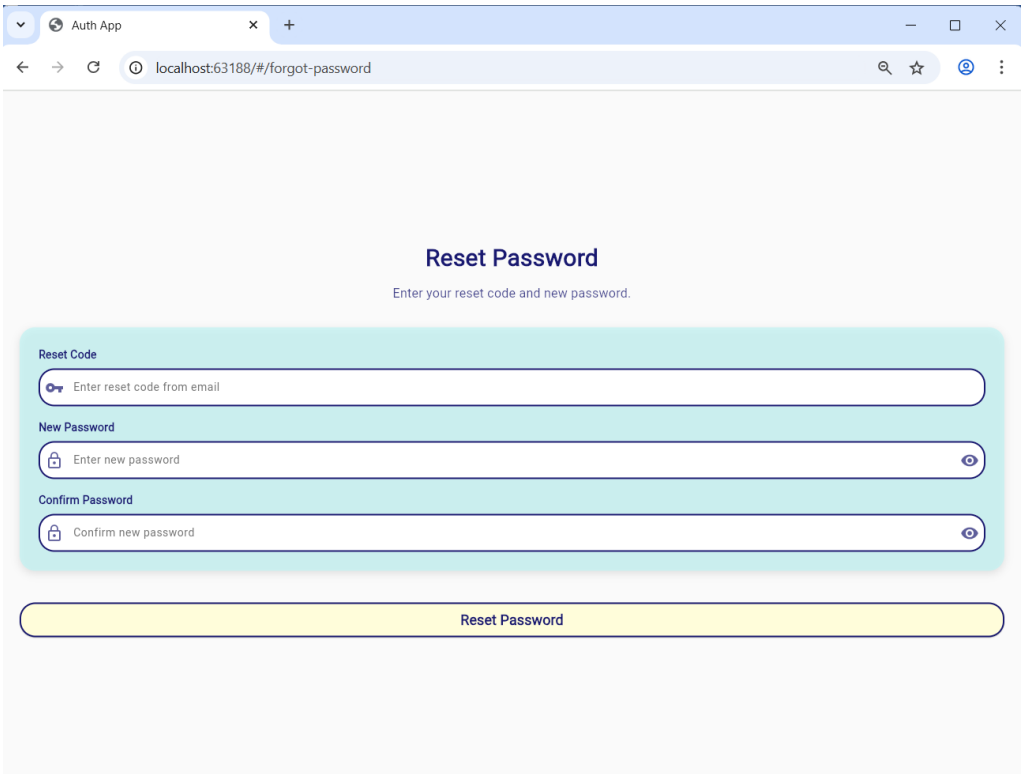


Figure 60 Reset Password-Web

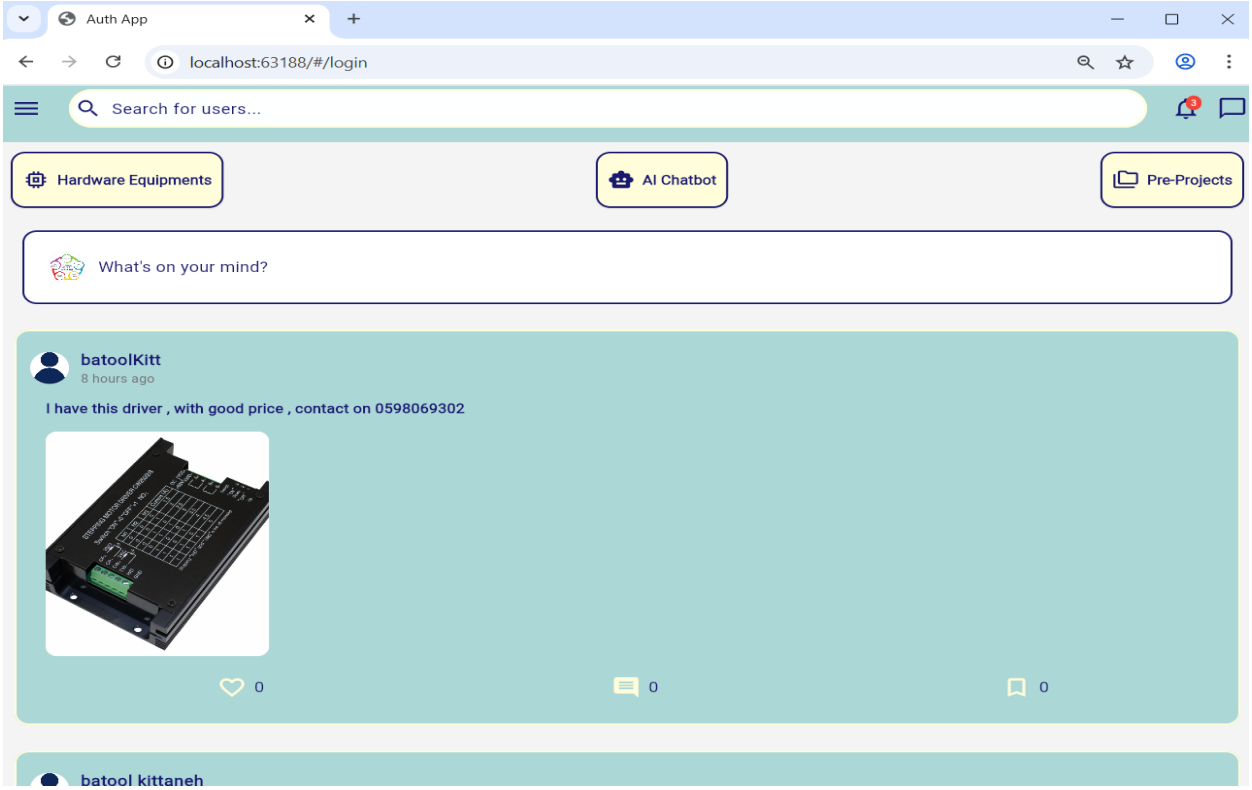


Figure 61 Home Screen Page-Web

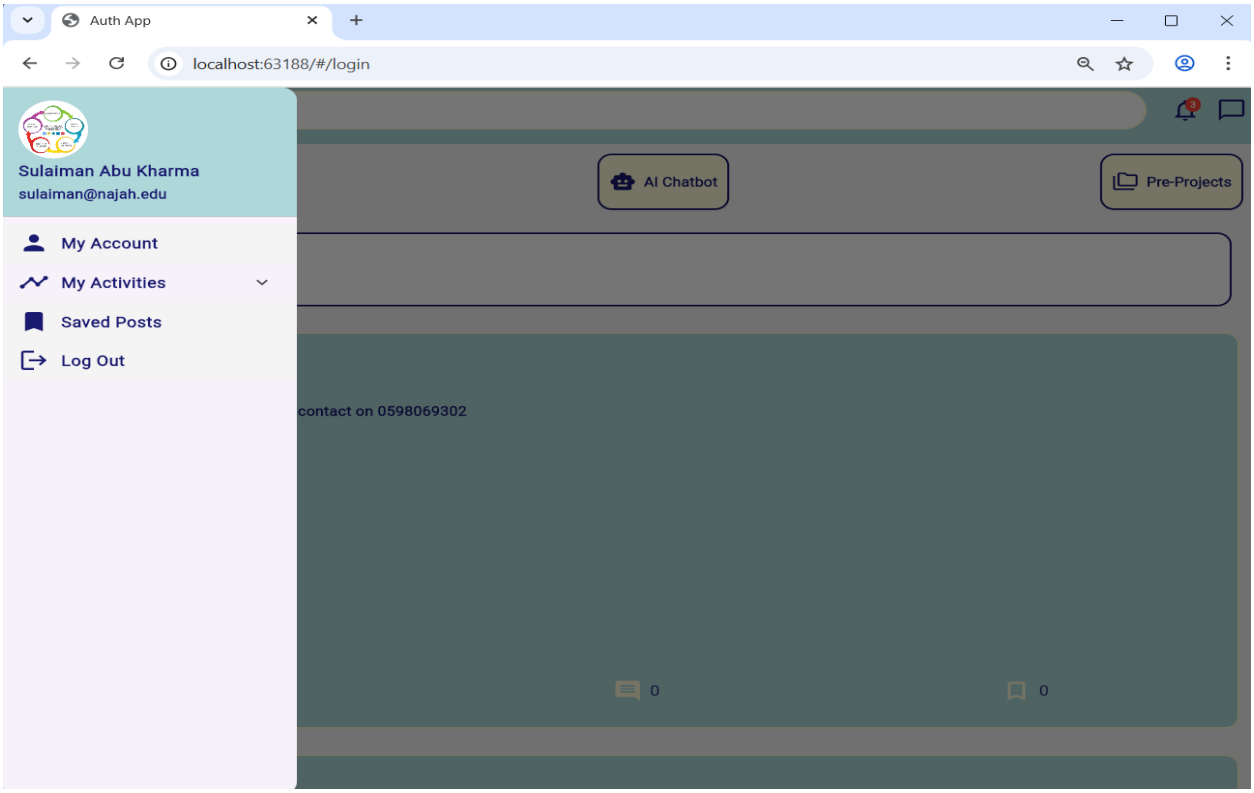


Figure 62Personal information-Web

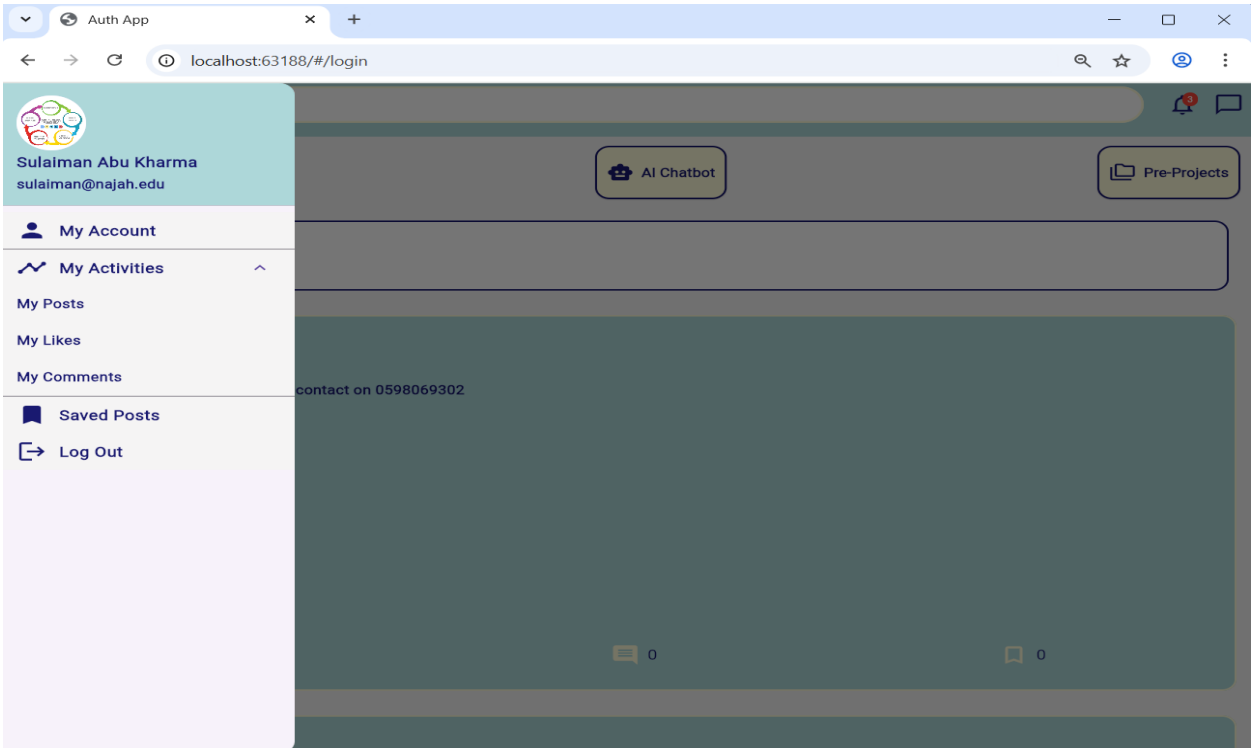


Figure 63 My Activities-Web

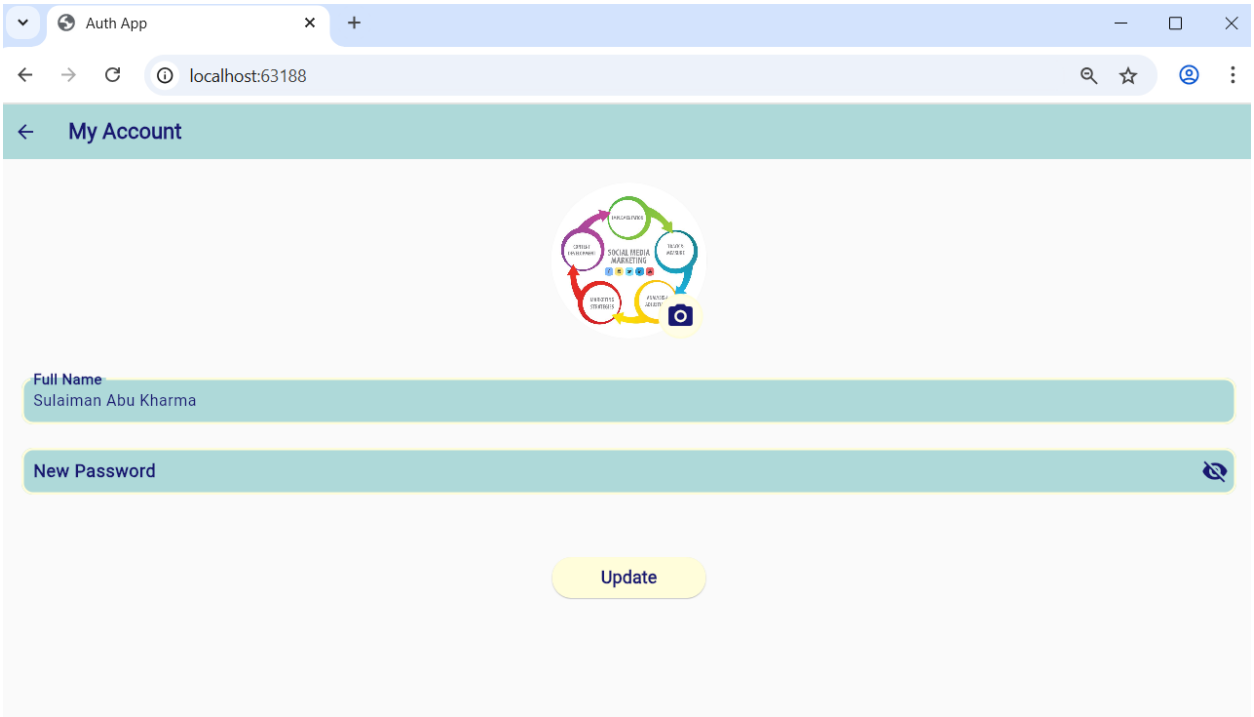


Figure 64 My Account-Web

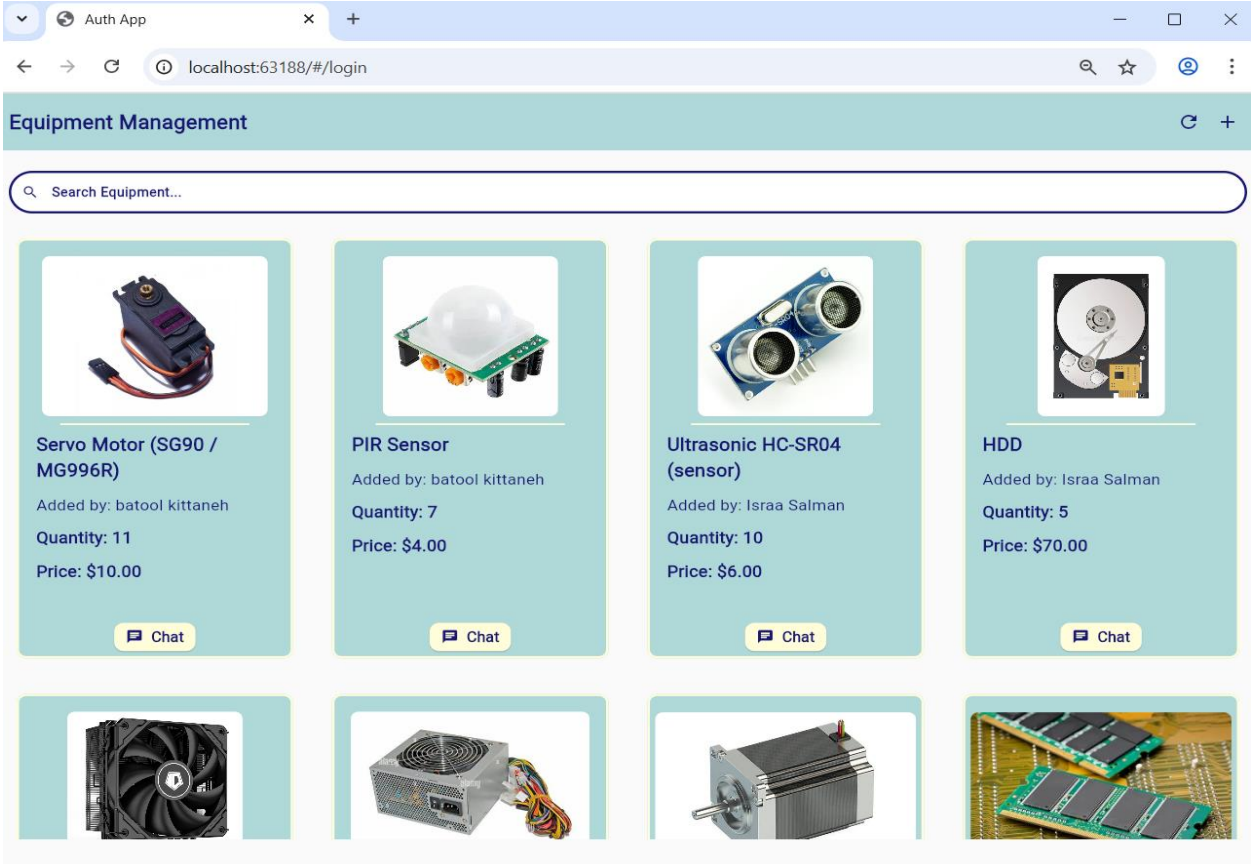


Figure 65 Hardware Equipment-Web

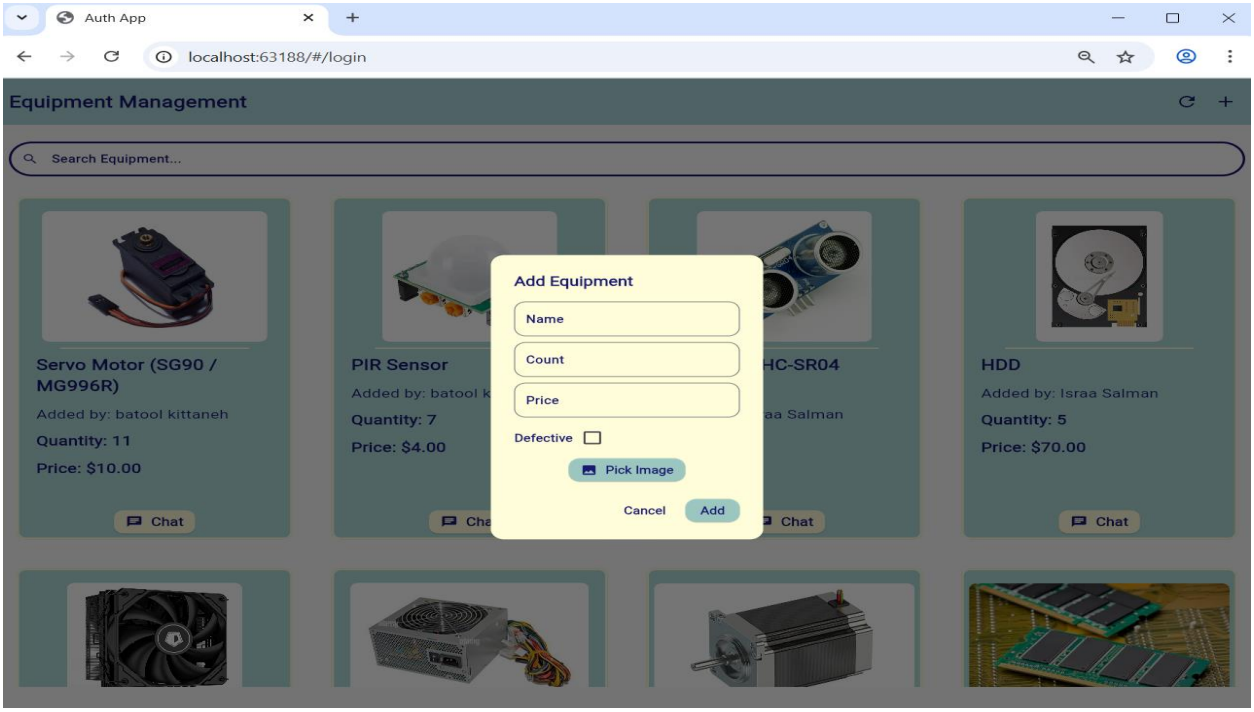


Figure 66 Add Equipment-Web

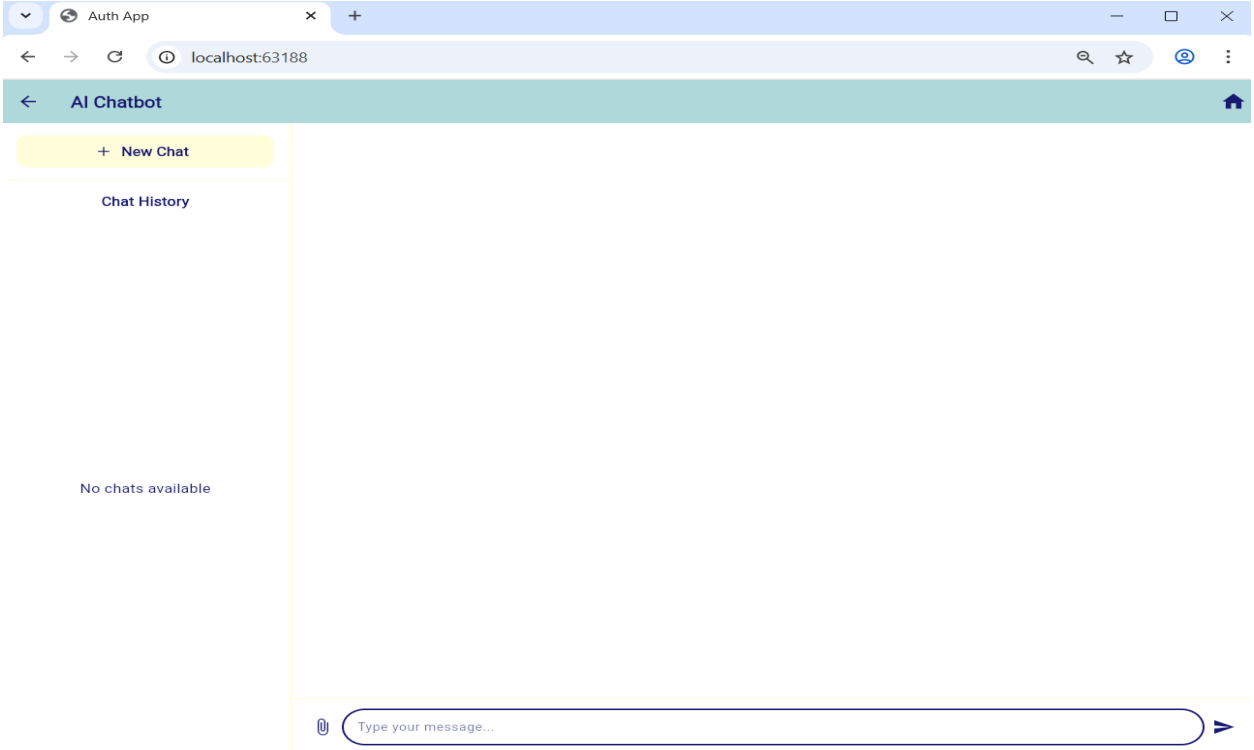


Figure 67 AI Chatbot-Web

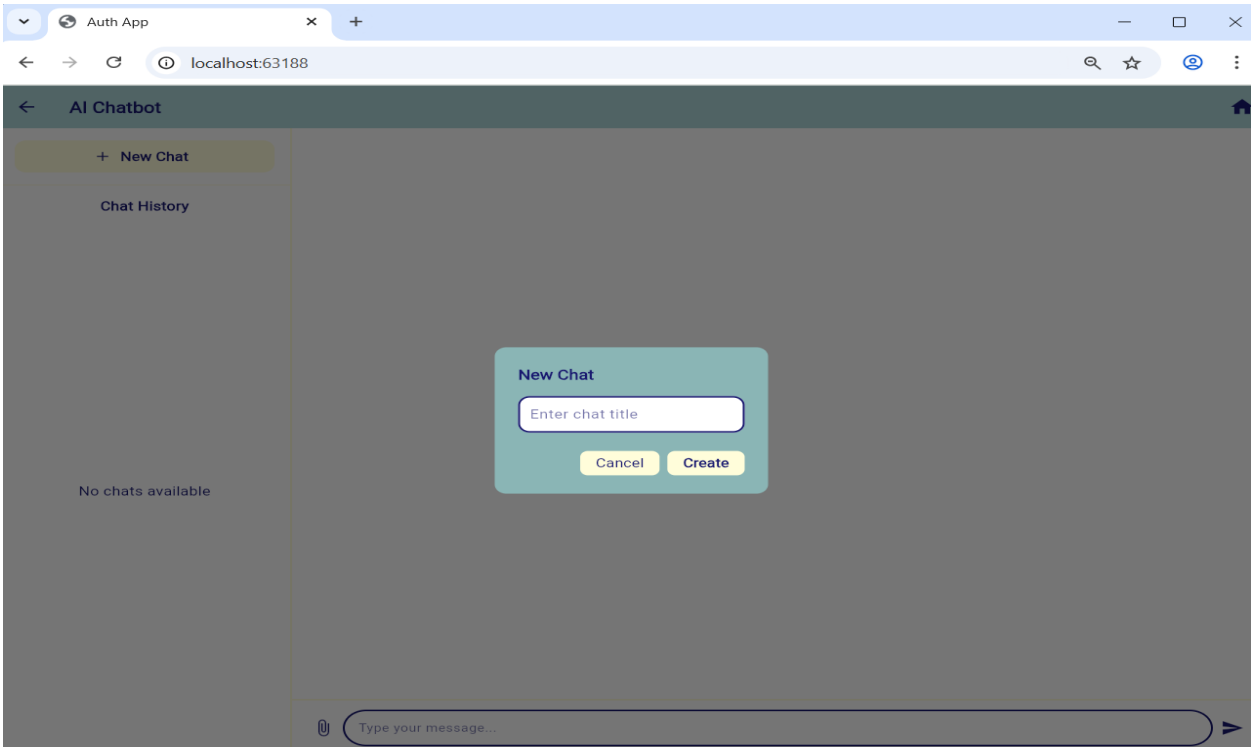


Figure 68 New AI Chatbot-Web

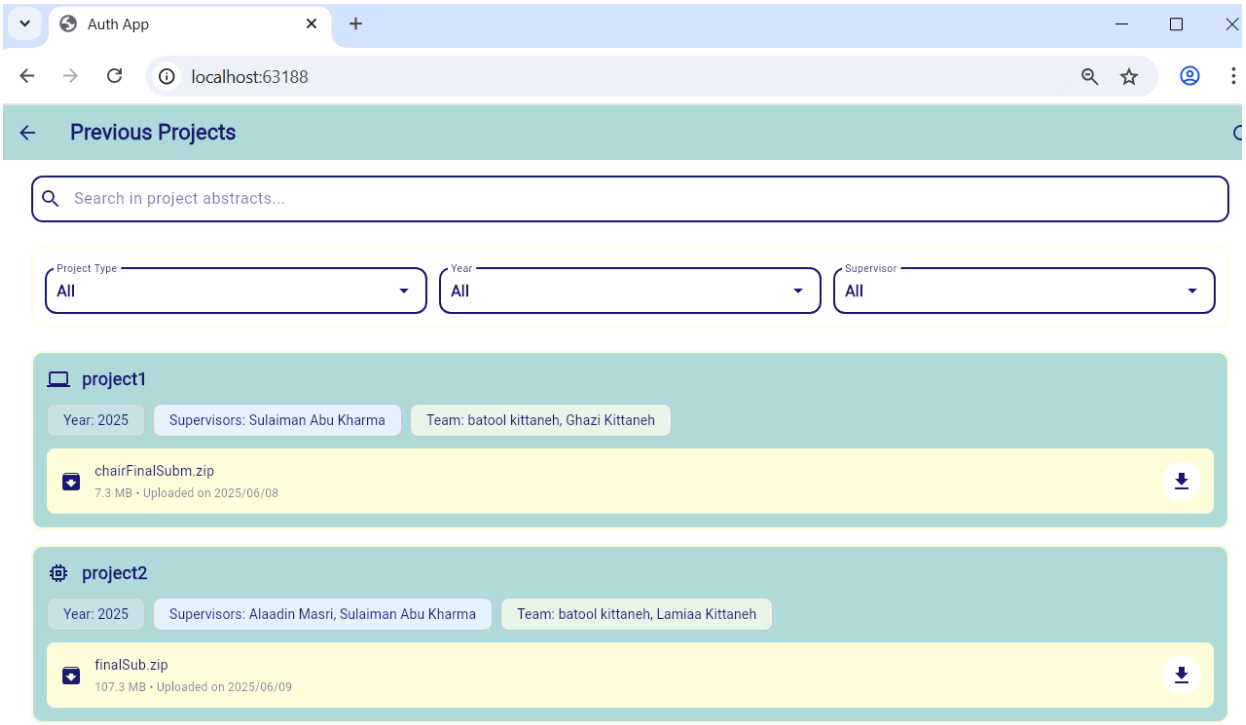


Figure 69 Previous Projects Page-Web

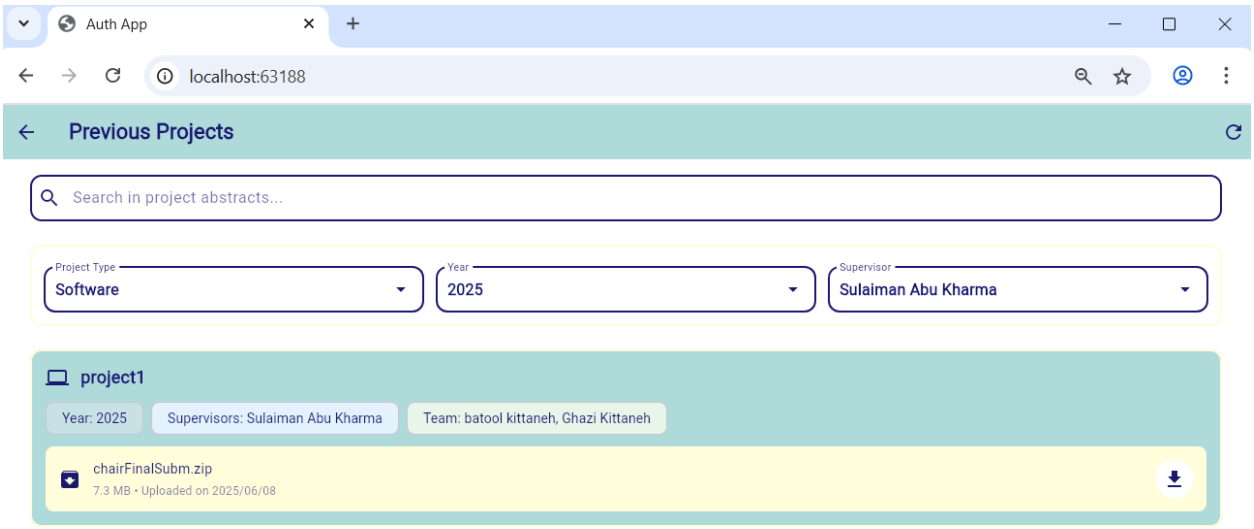


Figure 70 filtering previous projects-Web

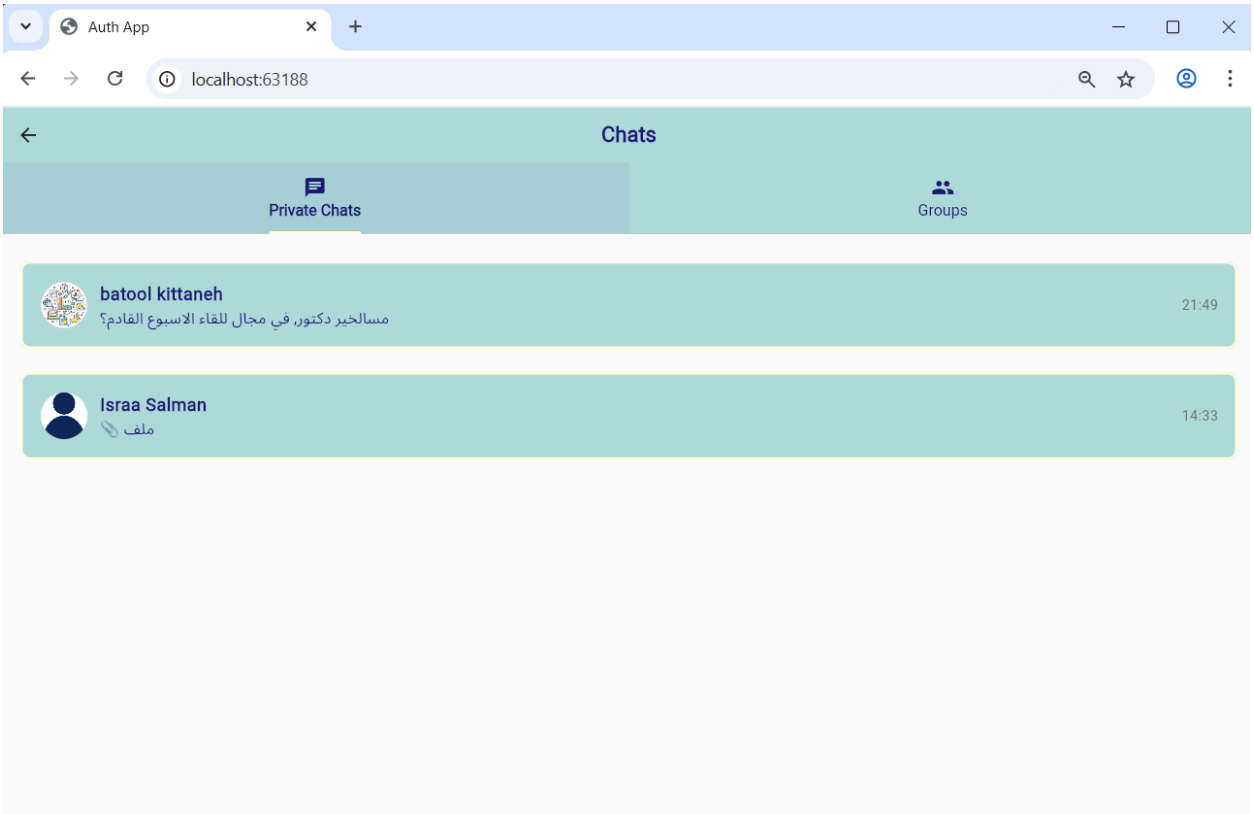


Figure 71 Chat Page-Web

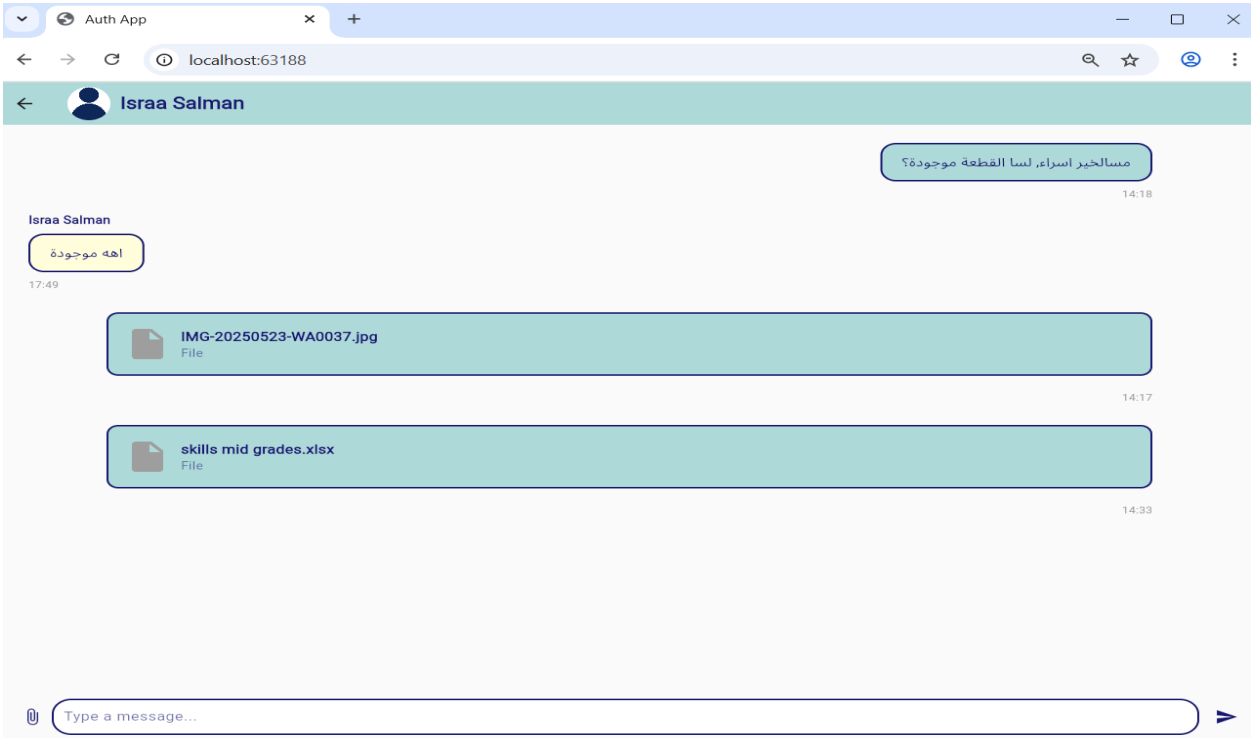


Figure 72 Private Chat -Web

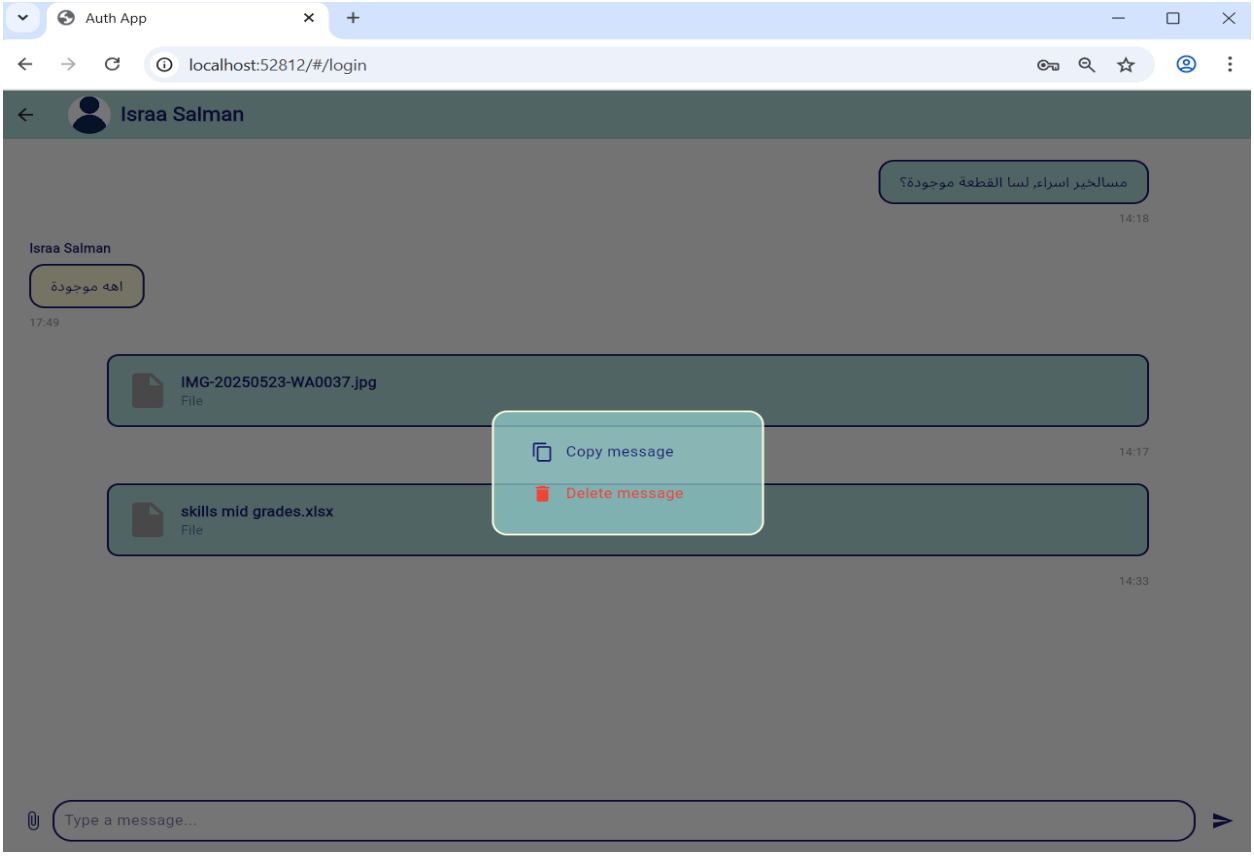


Figure 73 Edit message-Web

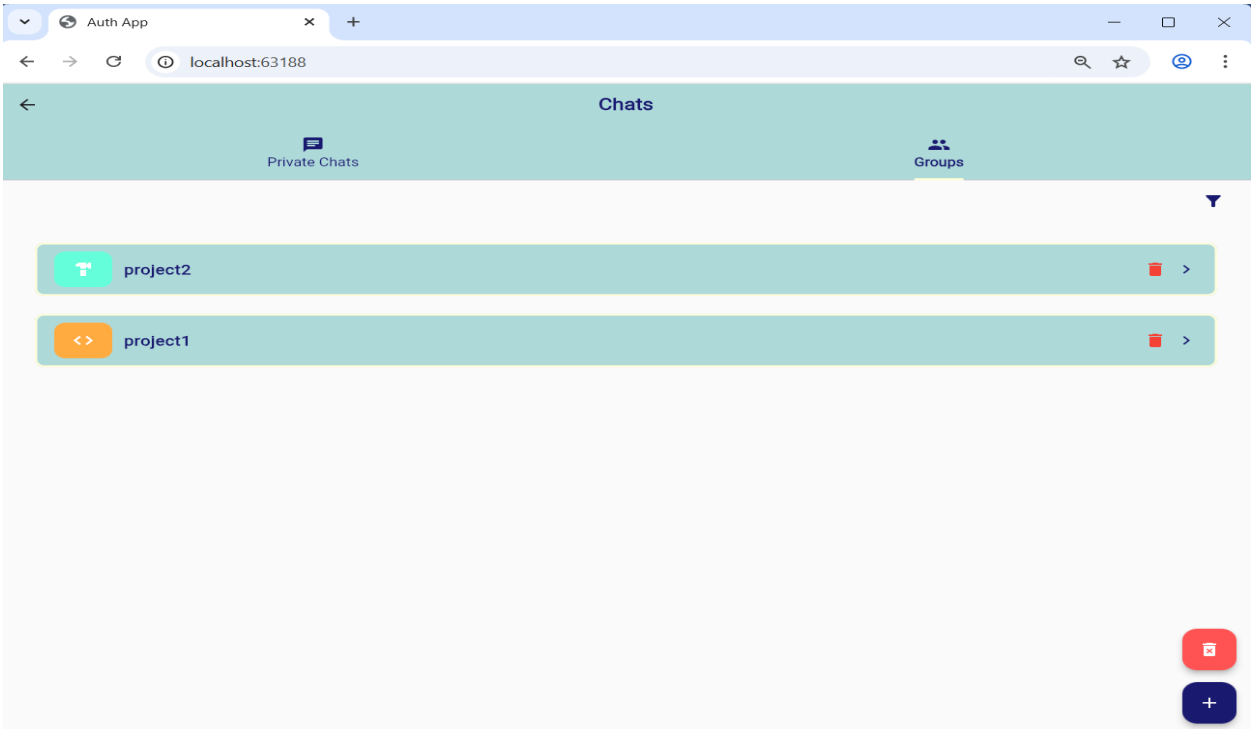


Figure 74 Groups Chat -Web

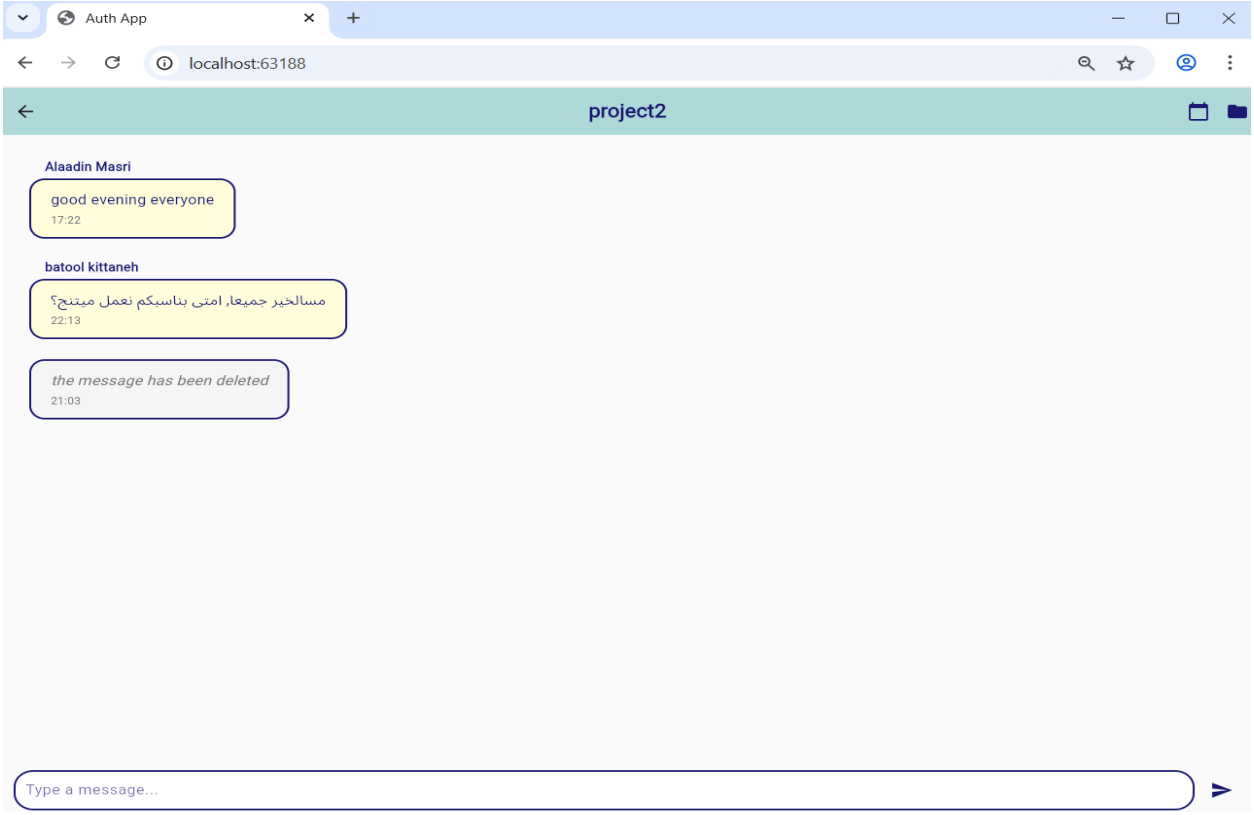


Figure 75 Group Chat-Web

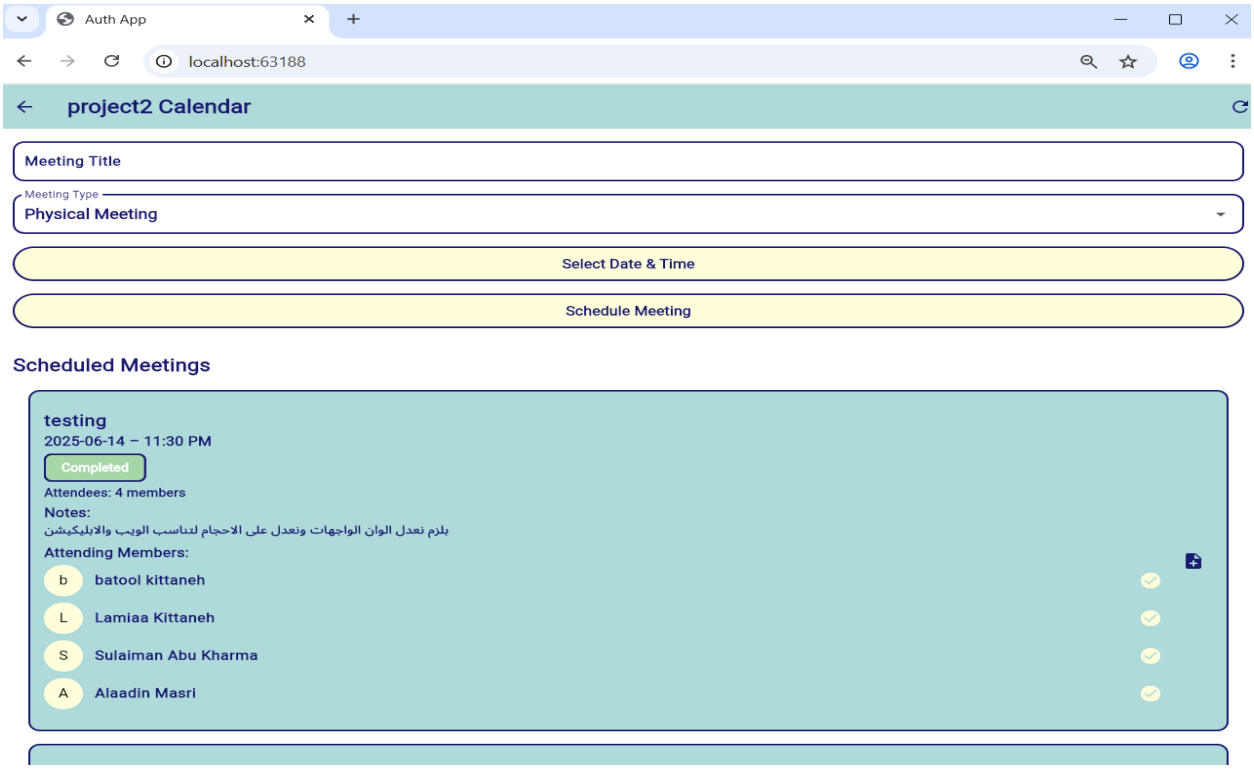


Figure 76 Smart Calendar Page-Web

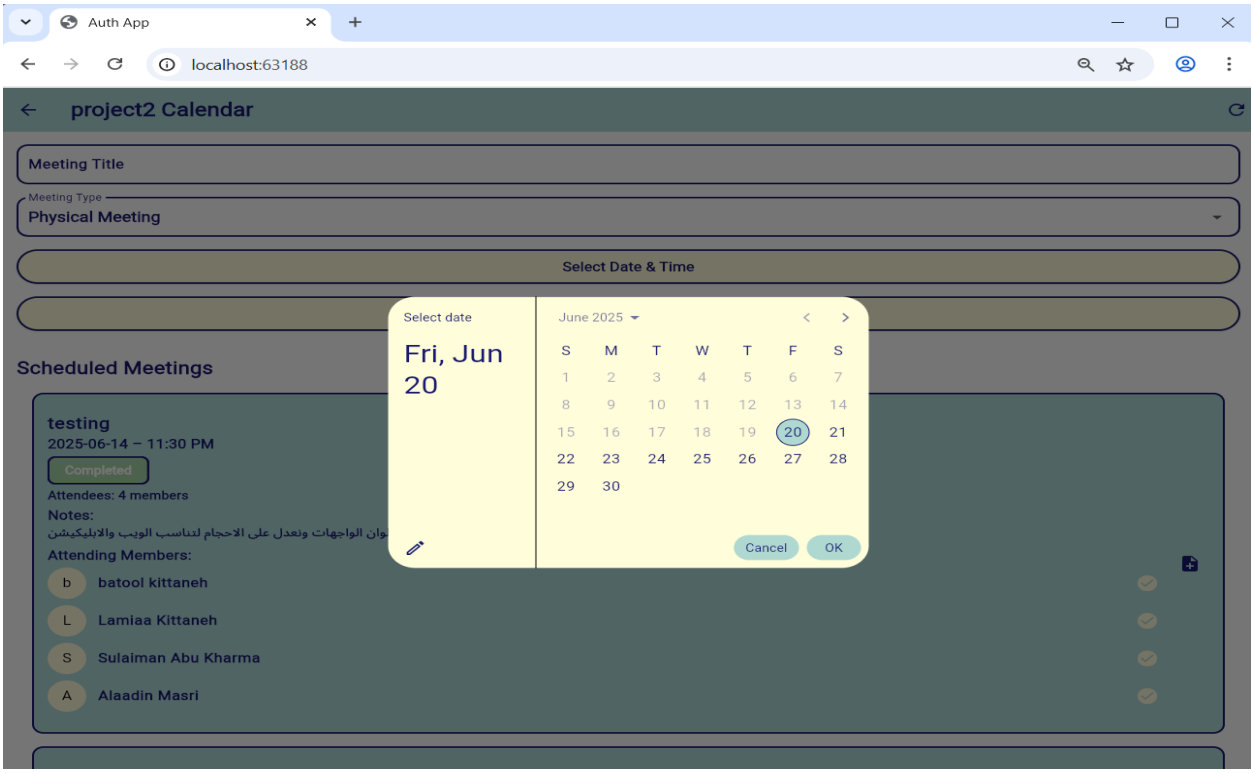


Figure 77 Selecting Date-Web

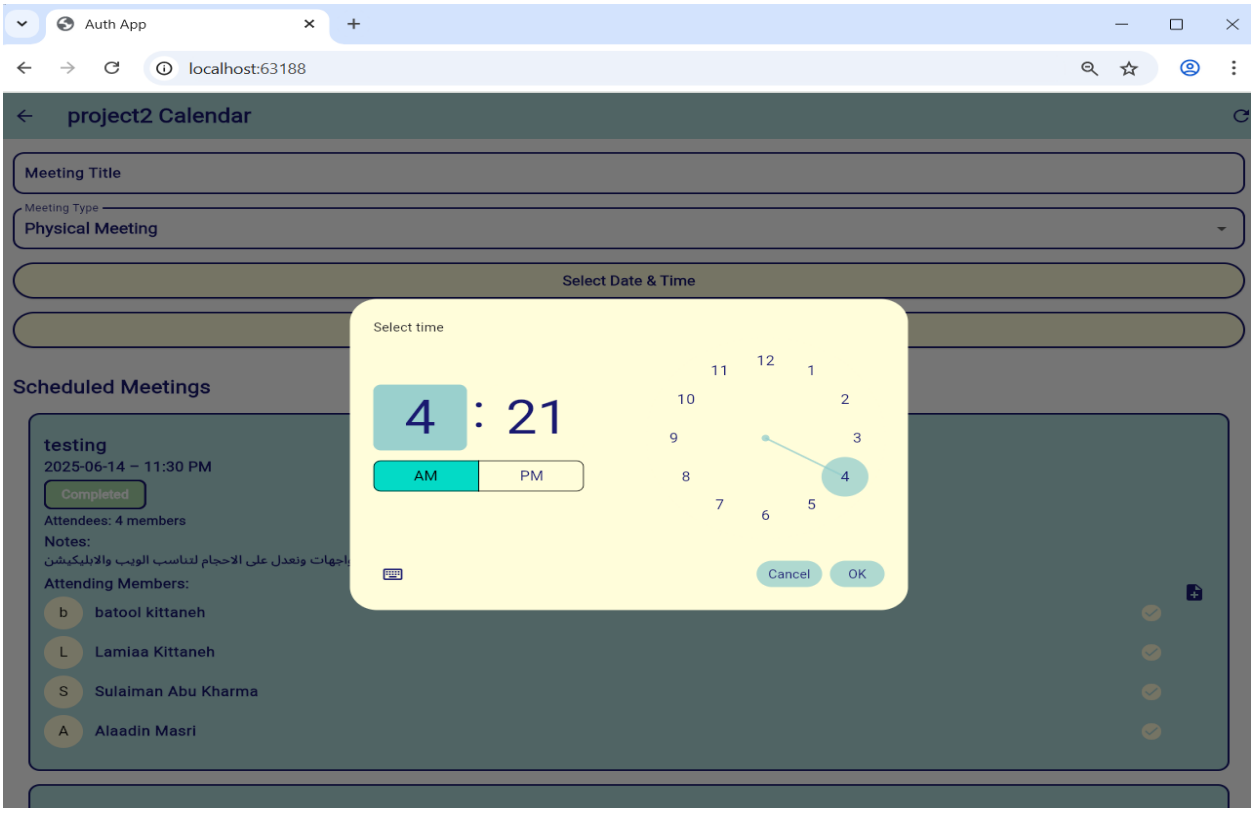


Figure 78 Selecting hour

Figure 79 Selecting minute -Web

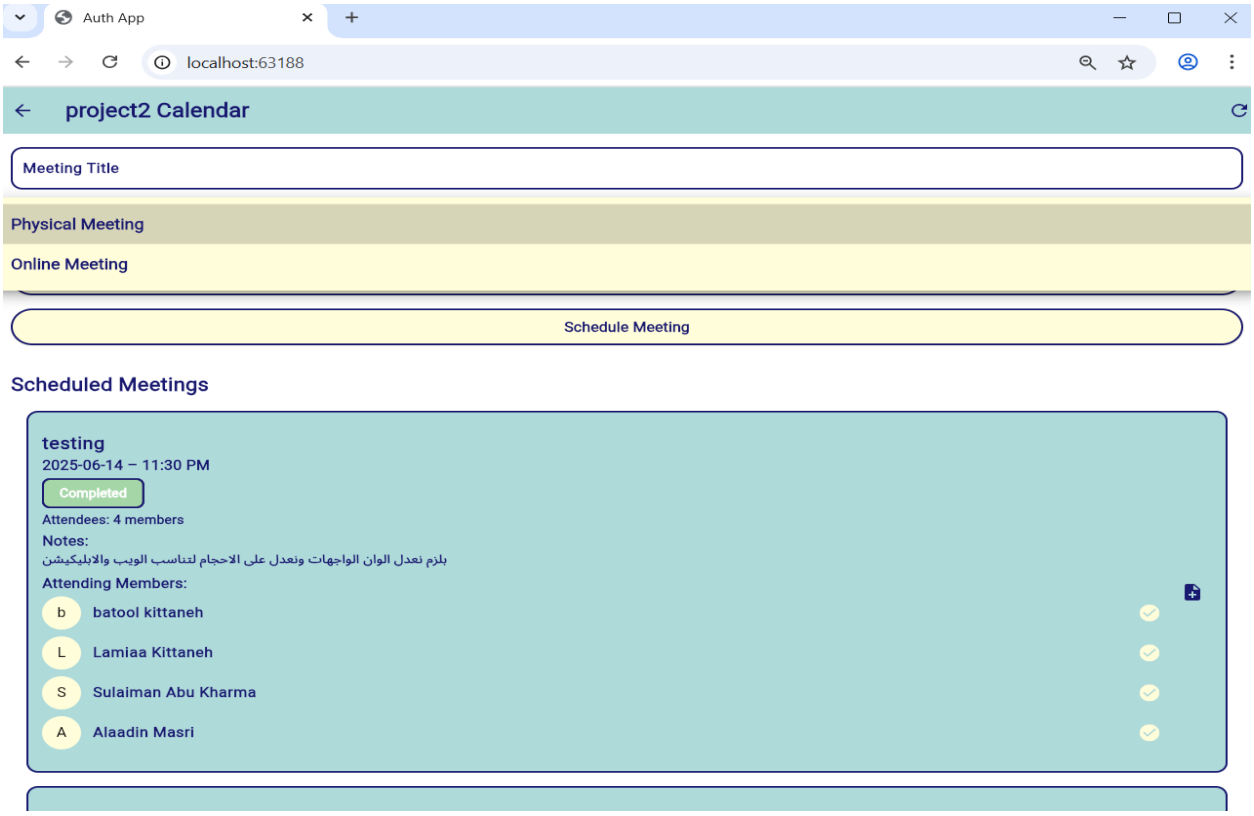


Figure 80 Meeting Type-Web

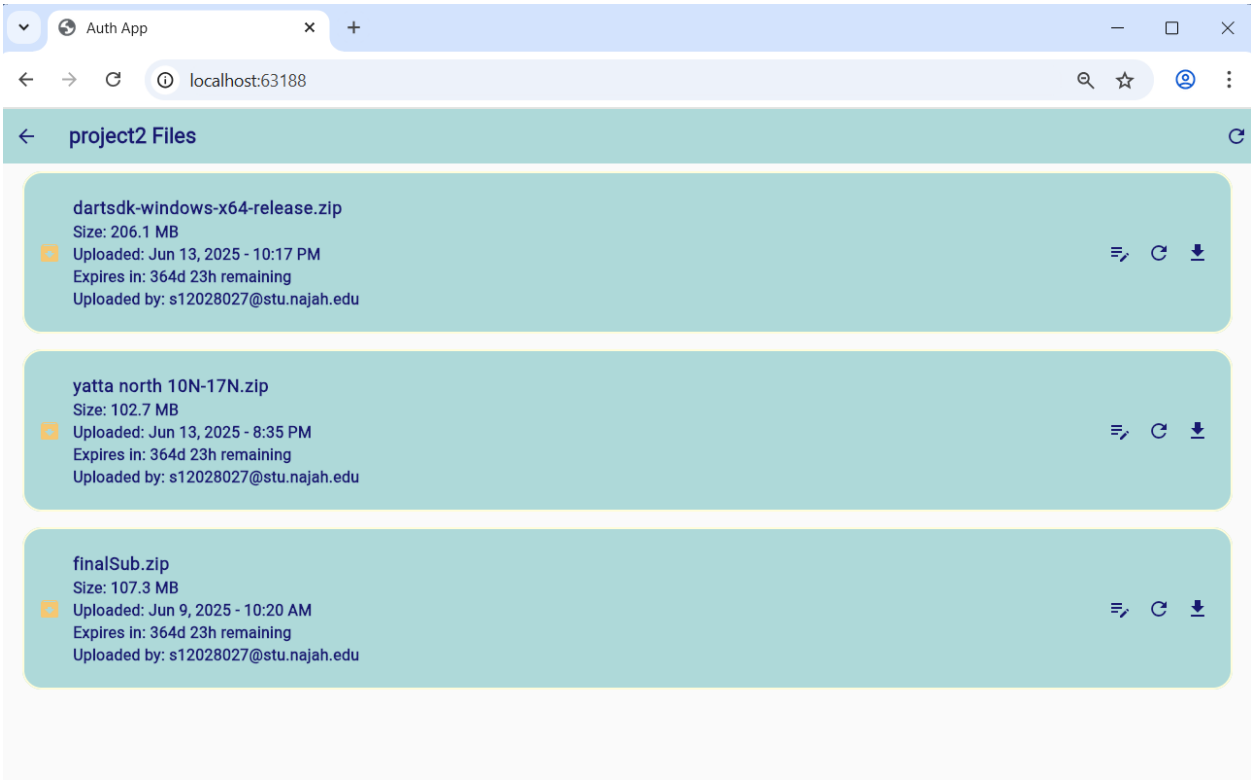


Figure 81 Group Files Page -Web

Figure 82 Notification Page-Web

Results & Discussion

Result

The system is an internet-based system that supports university students and staff by allowing them to view and manage study projects and interact effectively. The system provides the following functions:

1-Users are made able to browse previous graduation projects, view details such as year, group members, supervisors, and download files that were uploaded.

2-Users are made able to search and filter projects by type, year, or supervisor.

3-The users can have group or individual discussions with supervisors and other staff.

4-The users can send and receive messages with attachments.

5-Admins have the authority to manage the system content, i.e., uploaded projects.

Discussion

This system will be used to enhance academic cooperation and knowledge exchange by providing an integrated and simple-to-use interface. It ensures that students and supervisors can access essential resources and maintain correspondence in an orderly manner.

Conclusion & Future Work

Conclusion

The Academic Project Management system offers basic services to students and academic staff. The system allows one to search for previous graduation projects, including descriptions and downloadable reports, in aid of students' knowledge sharing. It also assists in a built-in chat system that allows students and supervisors to communicate and thus coordinate activities and quickly respond to questions. The system also provides an easy-to-use interface with search and filter functions, enhancing usability and navigation.

Future Improvements

Some potential improvements can be added to the platform in the future, including:

1-Adding a project rating system

To allow students and supervisors to rate uploaded projects to highlight quality work.

2-Adding the ability to comment on projects

So users can pose questions, give feedback, or comment on project material directly.

3-Adding a chat archives feature

So users can instantly view and search through their previous conversations.