LEBANESE INTERNATIONAL UNIVERSITY

Repair App

Home Repair and Maintenance Application

Senior Project

By

Hadi Rida

Submitted to the School of Arts & Science of the

Lebanese International University Beirut, Lebanon

In part of fulfillment of the requirements for the degree of

BACHERLOR OF SCIENCE IN

COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

Supervised by: Dr. Issa Kamar

Spring 2024-2025

Dedication:

First, I want to thank Almighty God for giving me the strength, knowledge, and patience to complete this project.

I also thank my dear parents and friends for always supporting me, helping me in my studies, and encouraging me to do my best.

I give special thanks to our respected professor, Dr. Issa Kamar, for guiding us step by step and helping us solve every problem during this project.

Finally, I dedicate this work to the souls of the martyrs who gave their lives to protect our country. We remember their sacrifice with honor and hope for peace in Lebanon.

Acknowledgment:

I have really worked hard and put a lot of effort into this project.

However, it would not have been completed on time or in the way I imagined without the special support of some important people. I would like to give my sincere thanks to all of them.

I am especially thankful to Dr. Issa Kamar for his constant guidance, support, and advice during every step of the project. His help made a big difference and gave me the confidence to keep moving forward.

My thanks and appreciation also go to my parents and friends for their encouragement and motivation. Their support helped me stay focused and finish this project in the best way possible.

Abstract:

This project presents the development of a mobile Home Repair and Maintenance Application designed to connect users with qualified repair specialists through an intuitive, internet-based platform. The primary objective is to streamline the process of finding and hiring reliable professionals such as electricians, plumbers, handymen, and contractors by eliminating traditional barriers like time-consuming searches and lack of communication channels.

The application provides a user-friendly interface for clients to submit repair requests by selecting appropriate service categories, providing descriptions, uploading images of the issue, and specifying their location. After submitting a request, users can monitor the progress of the job, communicate directly with specialists via in-app chat, and leave feedback upon completion.

On the specialist side, professionals can browse available repair requests, respond to jobs, update clients with images and status reports, and maintain open communication throughout the service process.

This report explores each of these features in depth, highlighting how the application enhances the repair experience for both users and specialists. Ultimately, the system is designed to save time, reduce stress, and facilitate trustworthy, location-based repair services—anytime, anywhere.

Keywords: **Home repair, Mobile application, Repair specialists, Online service platform, Client-specialist communication, Location-based services.**

List of Tables:

List of Figures:

Table of Contents:

Chapter 1

In this chapter, I will explain the main problem that I noticed, and how my project helps to solve it. In addition to that, I will mention the main objective of the project. Each part will be explained clearly to show how the idea was developed and why this application is useful.

1.1-Introduction:

Today, people are always looking for fast and easy ways to get things done, especially when it comes to fixing problems in their homes or offices. Finding a trusted specialist like an electrician, plumber, handyman or contractor can sometimes be difficult and time-consuming. That's why I decided to create a mobile application that helps users submit repair requests and connect directly with the right professionals.

This application allows users to describe the issue, choose the repair category, upload images, and share their location. Specialists can then view the requests, respond to them, and update the job progress. The application also includes chat and feedback features to improve communication between users and specialists.

The goal of this project is to make repair services more organized and efficient for everyone. In this report, I will explain the idea behind the project, the features included, and how this solution can help both users and specialists in their daily tasks.

1.2-Objective

The main objective of this project is to create a mobile application that makes it easier for people to request repair services for their homes or offices and get fast help from the right specialists.

Through this application, users can report problems, upload photos, choose the repair category, and share their location. On the other side, specialists can receive requests, respond to them, update the job status, and communicate with the users.

This project also aims to improve the repair process by saving time, reducing stress, and helping people find trusted repair workers without any confusion. It brings everything together in one easy-to-use app that works anytime and anywhere.

1.3-Scope

This project aims to develop a mobile application that connects users with repair and maintenance specialists quickly and easily. Users can report issues, choose a service type, upload photos, and share their location. Specialists can respond, send progress updates, and chat with users.

The app supports key features like request submission, communication, feedback, and job tracking for both users and specialists. The scope is limited to basic home and office repairs such as electrical, plumbing, and general maintenance, and does not include payments or large-scale construction services.

1.4-Problem

When people try to get repair services the usual way, they often face many problems that make it slow and difficult to find the right help:

- People have difficulty finding the right specialist (electrician, plumber, handyman, contractor) for home or office repairs.
- Contacting and reaching specialists often takes too much time and effort.
- Users have no easy way to report repair problems with details and photos.
- Communication between users and specialists is unclear and unorganized.
- Specialists find it hard to manage and respond to repair requests efficiently.
- Lack of a system to track job progress and update users causes frustration.
- Usual way to get repair services can be slow, inefficient, and stressful for everyone involved.

1.5-Solution

Creating an Repair Application that will help in:

- Develop a mobile application that connects users directly with trusted repair specialists like electricians, plumbers, handymen, and contractors.
- Allow users to easily submit repair requests with detailed descriptions, photos, and location information.
- Provide specialists with a platform to view, accept, and manage repair requests efficiently.
- Enable organized communication between users and specialists through in-app chat.
- Specialists find it hard to manage and respond to repair requests efficiently.
- The application includes features to track the progress of repair jobs and keep users updated at every step..
- Make the whole repair process faster, more organized, and less stressful for both users and specialists.

Chapter 2

This chapter starts with the Gantt chart, which shows the project schedule and important tasks. Then, it explains the functional and non-functional requirements of the application. After that, it shows UML use case diagrams and simple scenarios to describe how users and specialists will use the app.

2.1-Gantt Chart

The following Gantt chart outlines the overall project timeline using task numbers for clarity. A detailed description of each task including names, subtasks, and dates is provided in the *Gantt Chart Task Schedule* .

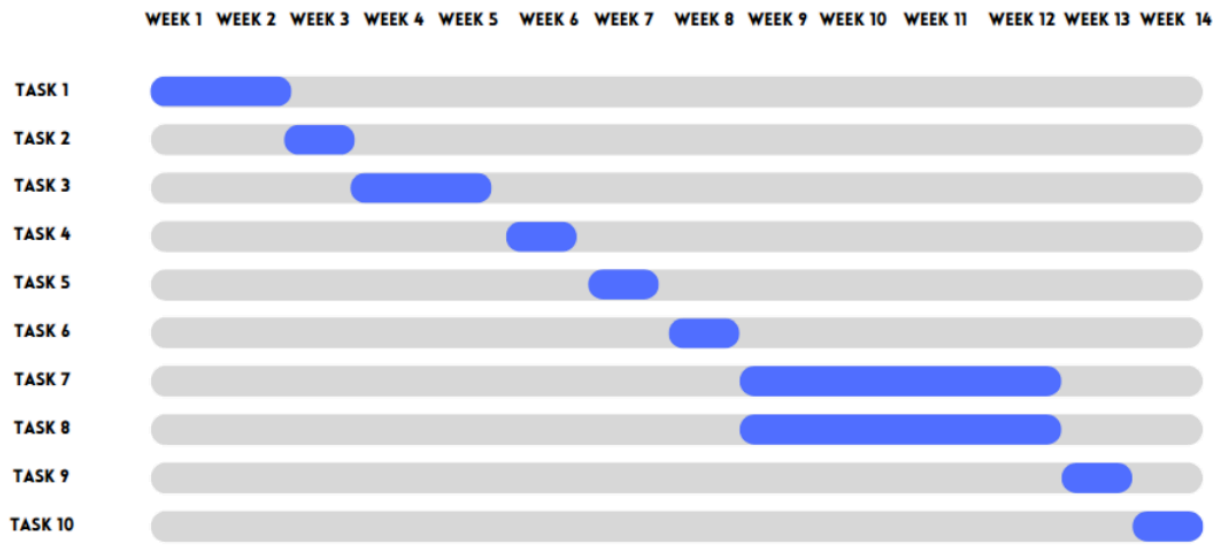

Figure 1 - Gantt Chart Overview

Table 1 - Gantt Chart Task Schedule

Task Number	Task Name	Start Date	End Date	Duration	Dependencies
Task 1	**Planning & Research:** ✗ Identify user needs ✗ Requirement gathering ✗ Define scope	Feb 24, 2025	Mar 9, 2025	2 weeks	-
Task 2	**Presenting Idea & Confirmation:** ✗ Prepare presentation ✗ Define features ✗ get feedback	Mar 10, 2025	Mar 16, 2025	1 week	Task 1
Task 3	**Storyboarding:** ✗ Create wireframes ✗ design user flow	Mar 17, 2025	Mar 30, 2025	2 weeks	Task 2
Task 4	**Business Process Modeling:** ✗ Identify processes ✗ Create diagram	Mar 31, 2025	Apr 6, 2025	1 week	Task 3
Task 5	**Use Case Diagram:** ✗ Identify actors ✗ Draft diagram	Apr 7, 2025	Apr 13, 2025	1 week	Task 4
Task 6	**ERD Design:** ✗ Define entities ✗ Design ERD	Apr 14, 2025	Apr 20, 2025	1 week	Task 5
Task 7	**Back-end Development:** ✗ Develop user authentication ✗ Implement database connection ✗ Create API ✗ Testing and debugging	Apr 21, 2025	May 18, 2025	4 weeks	Task 6
Task 8	**Front-end Development:** ✗ develop UI based on storyboard ✗ Implement user authentication ✗ Integrate APIs with back-end ✗ Testing	Apr 21, 2025	May 18, 2025	4 weeks	Task 6
Task 9	**Documentation:** ✗ Prepare technical documents ✗ Create user manual ✗ Submit required files	May 19, 2025	May 25, 2025	1 week	Task 7 & 8
Task 10	**Final Presentation**	May 26, 2025	Jun 1, 2025	1 week	Task 9

2.2-Functional and Conceptual Study

2.2.1-Functional Requirements:

1. **User Registration and Authentication**
    - Users can register and log in securely.
    - Sanctum-based API token management.
    - The system supports different user roles: User and Specialist.
2. **User Profile Management**
    - Users can update their profile information, including name, email, profile photo, location (with autocomplete).
    - For specialists: an additional bio field.
    - Change password, delete account, and logout functionality.
3. **Report Creation**
    - Users can create a report describing a repair issue.
    - Reports include title, category , location, description, and images.
4. **Specialist Matching**
    - System finds available specialists based on category, sorted by location.
    - Specialists can monitor incoming requests.
5. **Job Tracking**
    - Users can track the progress and status of their repair job.
    - Specialists can upload images and progress comments.
6. **Chat System**
    - Users and specialists can send direct messages.
    - Chat shows the latest message per conversation.
    - Messages are stored and retrieved based on user ID pairs.
7. **Feedback System**
    - After job completion, users can submit feedback and rate the specialist.
    - Feedback is associated with the specialist’s profile for future reference.

2.2.2-Nonfunctional Requirements:

1. **Usability**

   The app should provide a simple and intuitive interface for users and specialists, with clear navigation and consistent design.

2. **Performance**

   System responses should be fast and efficient, ideally within 1–2 seconds under normal conditions.

3. **Security**

   User data must be protected using encrypted passwords and token-based authentication, with role-based access control in place.

4. **Scalability & Maintainability**

   The system should support future growth in user numbers and features. Code and APIs must follow clean, modular practices to allow easy updates.

5. **Scalability & Maintainability**

   The Flutter app must run smoothly on both Android and iOS. The system should be accessible at all times with proper error handling for downtime or connectivity issues.

6. **Reliability**

   The system must operate consistently without crashes or data loss, especially during critical actions.

2.3-UML Use Case Diagram & Scenario:

2.3.1-Use Case Diagram

The Repair App helps people who need repair services connect with skilled specialists who can do the job. The system has different features for two main types of users: the User and the Specialist.

The User uses the app to manage their repair needs from the beginning to the end. Important actions for the User include: logging in, registering a new account, logging out, viewing the home page, updating their profile, deleting their account, viewing chat messages, creating a damage report, searching for specialists, tracking the status of a report, and viewing their past reports. For example, when the User creates a damage report, they can describe what is broken and send a repair request. When they track a report, they can see updates about how the repair is going. Other features like changing the password, contacting support, sending or receiving messages, giving feedback, and deleting a report are usually part of the main actions, not separate ones. Also, when a User logs in for the first time, they usually register as a new user.

The Specialist uses the app to view new repair jobs, talk with users, and update the status of the work. The main actions for the Specialist involve handling and completing the repair tasks they receive. A Specialist is also a User, so they can do everything a regular User can do—like updating their profile, sending messages, and creating reports—along with their special tasks as a repair expert. These actions will be explained more in the next sections.

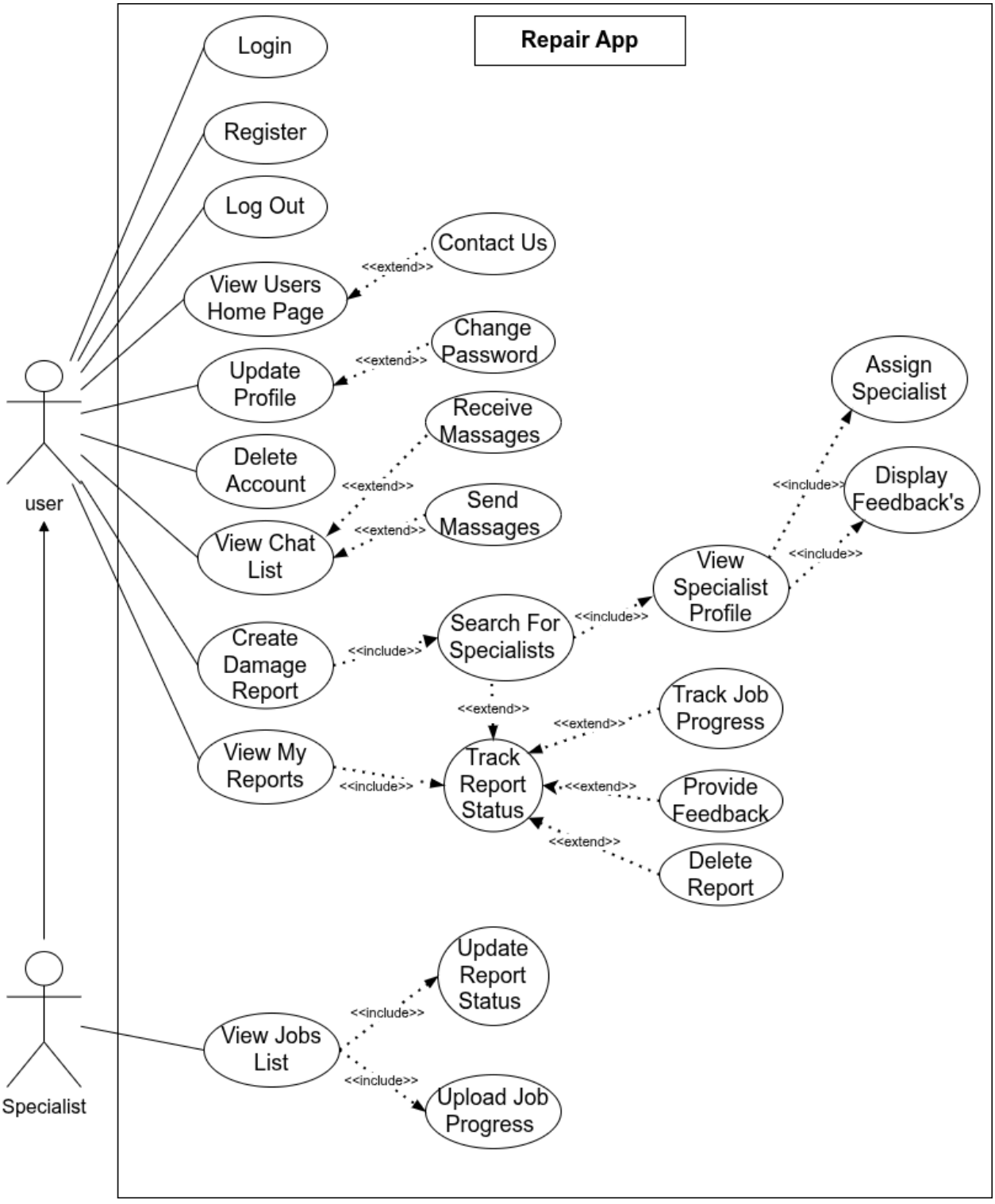

Figure 2 - Use Case Diagram

2.3.2-Use Case scenarios (user actor)

This section describes the main actions that a regular user can do in the app. Each scenario shows how the user interacts with the system step by step. These scenarios help explain the user's role and how the app supports their needs.

Table 2 - Login Use Case

Use Case ID	UC01
Use Case Name	Login
Actor(s)	User
Description	User logs into the system with email and password.
Preconditions	User is registered
Main Flow	1. User enters credentials 2. System verifies 3. Redirects to home
Alternate Flow	1. Wrong credentials → Show error
Postconditions	User is authenticated and session is started

Table 3 - Register Use Case

Use Case ID	UC02
Use Case Name	Register
Actor(s)	User
Description	User creates a new account.
Preconditions	User is not logged in.
Main Flow	1. User fills out the registration form. 2. System validates. 3. Account is created.
Alternate Flow	1. Validation fails → Show error.
Postconditions	New user account is created and stored.

Table 4 - Logout Use Case

Use Case ID	UC03
Use Case Name	Logout
Actor(s)	User
Description	User logs out of the application.
Preconditions	User is logged in.
Main Flow	1. User taps logout. 2. Session is cleared.
Postconditions	User is logged out.

Table 5 - View Users Home Page Use Case

Use Case ID	UC04
Use Case Name	View Users Home Page
Actor(s)	User
Description	User views dashboard and available actions.
Preconditions	User is logged in.
Main Flow	1. System displays a homepage. 2. User navigates to modules.
Extended Use Case	Contact Us (optional view or submit support queries)
Postconditions	Homepage with modules displayed.

Table 6 - Update Profile Use Case

Use Case ID	UC05
Use Case Name	Update Profile
Actor(s)	User
Description	User update their profile details.
Preconditions	User is logged in.
Main Flow	1. User edits profile fields. 2. System saves changes.
Extended Use Case	Change Password
Postconditions	Profile is updated.

Table 7 - Delete Account Use Case

Use Case ID	UC06
Use Case Name	Delete Account
Actor(s)	User
Description	User permanently delete their account.
Preconditions	User is logged in.
Main Flow	1. User confirms deletion. 2. Account is removed.
Postconditions	User data is deleted or anonymized.

Table 8 - View Chat List Use Case

Use Case ID	UC07
Use Case Name	View Chat List
Actor(s)	User
Description	User views and manages conversations.
Preconditions	User is logged in.
Main Flow	1. User opens chat list. 2. Messages are displayed.
Extended Use Case	Send or Receive Messages
Postconditions	User can communicate with specialists.

Table 9 - Create Damage Report Use Case

Use Case ID	UC08
Use Case Name	Create Damage Report
Actor(s)	User
Description	User submit a report about damage.
Preconditions	User is logged in.
Main Flow	1. User describes issue. 2. Searches specialists. 3. Views profile. 4. Assign one. 5. Submits report.
Included Use Cases	Search Specialist, View Specialist Profile, Assign Specialist, Display Feedbacks
Postconditions	Report is created.

Table 10 - View My Reports Use Case

Use Case ID	UC09
Use Case Name	View My Reports
Actor(s)	User
Description	User checks history of submitted reports.
Preconditions	User is logged in.
Main Flow	1. User navigates to reports. 2. Views list and details.
Included Use Cases	Track Report Status
Extended Use Cases	Delete Report, Provide Feedback, Track Job Progress, Search Specialist
Postconditions	Reports and statuses are displayed.

Table 11 - Search For Specialist Use Case

Use Case ID	UC10
Use Case Name	Search for Specialist
Actor(s)	User
Description	User search for specialists based on issue type or location.
Preconditions	User is logged in.
Main Flow	1. User enters search query. 2. System filters and shows specialists.
Postconditions	List of matching specialists is shown.

Table 12 - View Specialist Profile Use Case

Use Case ID	UC11
Use Case Name	View Specialist Profile
Actor(s)	User
Description	User views details of a specialist.
Preconditions	Specialist must exist.
Main Flow	1. User selects specialist. 2. System shows profile.s.
Postconditions	Specialist info is displayed.

Table 13 - Assign Specialist Use Case

Use Case ID	UC12
Use Case Name	Assign Specialist
Actor(s)	User
Description	User assigns a selected specialist to a damage report.
Preconditions	User has selected a specialist and filled report details.
Main Flow	1. User selects assign. 2. System confirms assignment.
Postconditions	Specialist assigned to the report.

Table 14 - Display Feedbacks Use Case

Use Case ID	UC13
Use Case Name	Display Feedbacks
Actor(s)	User
Description	User sees past feedback for a specialist.
Preconditions	Specialist must have feedback.
Main Flow	Feedback is shown.
Postconditions	User sees feedback list.

Table 15 -Track Report Status Use Case

Use Case ID	UC14
Use Case Name	Track Report Status
Actor(s)	User
Description	User tracks current status of submitted reports.
Preconditions	User has at least one submitted report.
Main Flow	1. User selects report. 2. System shows current status.
Postconditions	Report status is visible.

Table 16 - Provide Feedbacks Use Case

Use Case ID	UC15
Use Case Name	Provide Feedback
Actor(s)	User
Description	User leaves feedback for a completed report or specialist.
Preconditions	Report is completed and specialist is assigned.
Main Flow	1. User writes feedback. 2. System saves and links to report.
Postconditions	Feedback saved and displayed for future viewing.

Table 17 - Track Job Progress Use Case

Use Case ID	UC16
Use Case Name	Track Job Progress
Actor(s)	User
Description	User views updates/images of job work being done.
Preconditions	Job must be in progress.
Main Flow	1. User opens job progress. 2. System shows updates.
Postconditions	Job progress details are visible.

2.3.3-Use Case scenarios (specialist actor)

This section describes the main actions that a specialist can perform in the application. Since specialists are registered users, they also have access to all user functionalities described in section **2.3.2**. In addition to those, specialists can manage tasks assigned to them by users and report progress.

Table 18 - View Assigned Reports Use Case

Use Case ID	UC17
Use Case Name	View Assigned Reports
Actor(s)	Specialist
Description	Specialist views a list of damage reports assigned to them.
Preconditions	Specialist is logged in and has at least one report assigned.
Main Flow	1. Specialist opens the report section. 2. System displays assigned reports.
Postconditions	Specialist sees the report list and can choose to update or track them.

Table 19 - Update Report Status Use Case

Use Case ID	UC18
Use Case Name	Update Report Status
Actor(s)	Specialist
Description	Specialist updates the status of a report based on progress.
Preconditions	Specialist is logged in and has at least one report assigned.
Main Flow	1. Specialist selects a report. 2. Chooses a new status. 3. Confirms update.
Postconditions	Report status is changed and visible to the user.

Table 20 - Upload Job Progress Use Case

Use Case ID	UC19
Use Case Name	Upload Job Progress
Actor(s)	Specialist
Description	Specialist uploads images and comments to show the progress of work.
Preconditions	Specialist is logged in and assigned to the report.
Main Flow	1. Specialist selects a report. 2. Adds image and comment. 3. Submits the update.
Postconditions	Progress is saved and shown in the report tracking for the user.

Chapter 3

This chapter describes the system design, including the database of the project, the related tables and the relations between them.

3.1-System Design:

3.1.1-ER-Diagram

This Entity Relationship (ER) Diagram represents the data model of the Repair Application system. It illustrates the main database tables (entities) and the relationships between them. The diagram helps visualize how data is organized and connected within the system to support its functionality.

The main entities of the Repair Application system include: User, Specialist, Report, Job Progress, Feedback, and Chat System.

Repair App System attributes:

User: ID, First Name, Last Name, Email, Password, Location, Role.

Specialist: ID, Bio, Specialization.

Report: ID, User ID, Specialist ID, Title, Description, Images, location, Issue-type, Status.

Job Progress: ID, Report ID, Image, Comment.

Feedback: ID, User ID, Specialist ID, Comment.

Chat System: ID, Sender ID, Receiver ID, Massage.

3.1.2-Relational Diagram

Refer to *Figure 4 – Relational Diagram* for detailed view of each entity, in addition to the relations between entities.

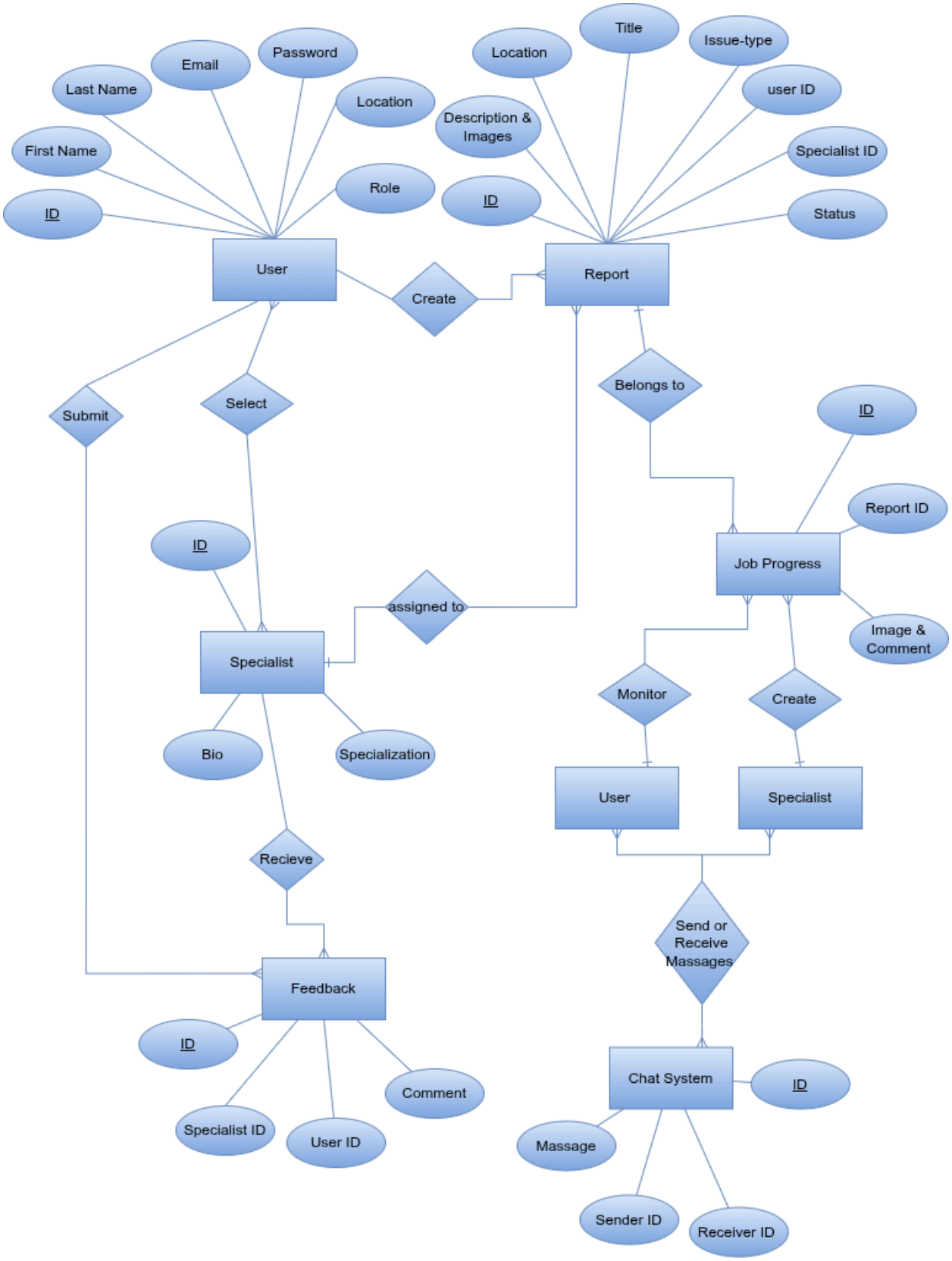

Figure 3 - ER-Diagram

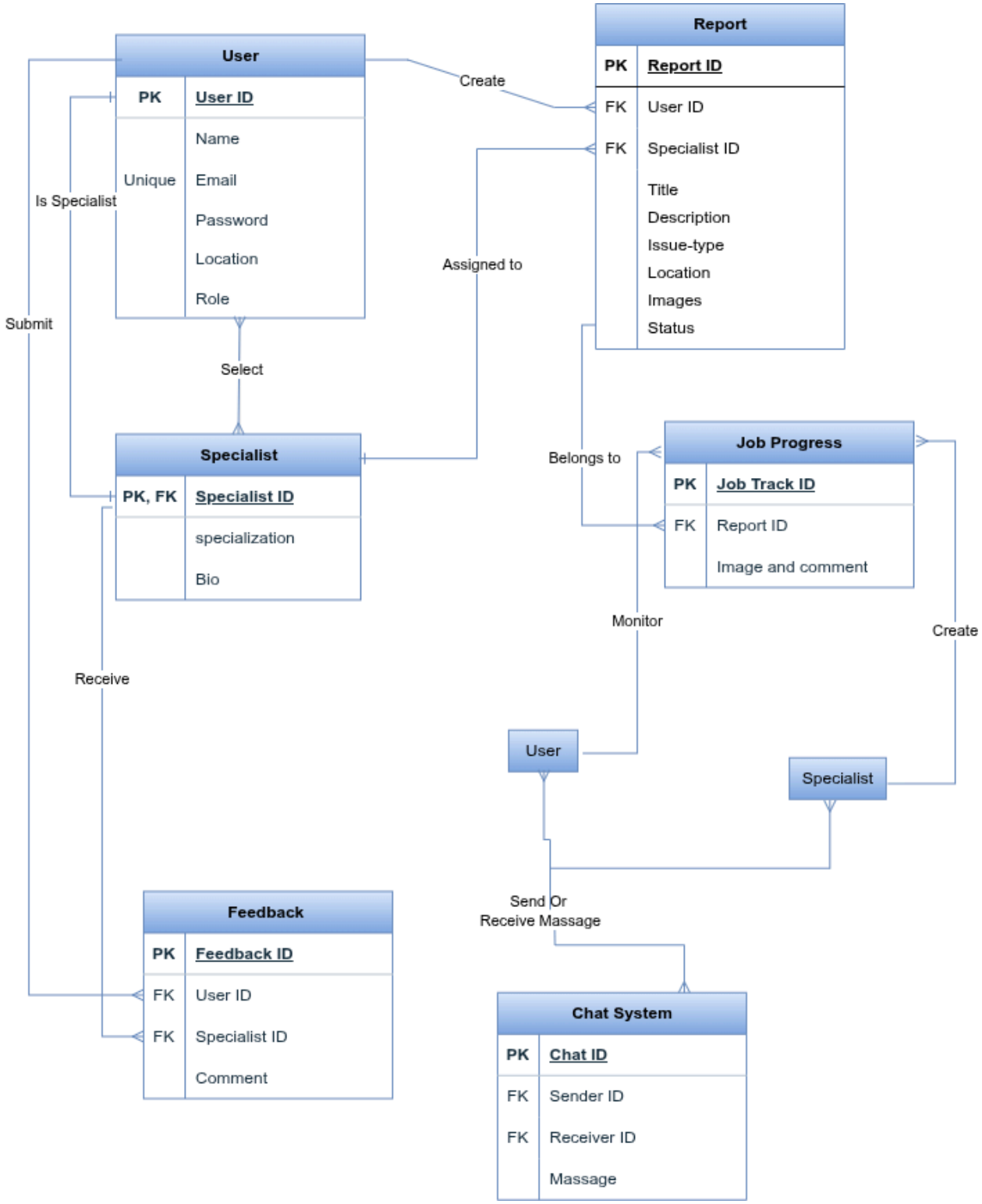

Figure 4 - Relational Diagram

Chapter 4

This chapter describes the implementation of the Repair Application. It begins with the technologies and tools used during development. Then, it explains the overall structure of the app, including the design of each screen and user interface component. The role of each screen, button, and feature is also explained in relation to its function within the system.

4.1-Introduction

Our repair app is designed to be easy to use and to help people quickly find repair specialists. We focused on making the app simple so that everyone can use it without any problems.

The app has a clean design, allowing users to easily report problems, check the progress of their repair, and communicate with specialists. We built the app using Flutter to provide a smooth mobile experience.

The backend uses Laravel to keep user information secure and to manage all the data. It also enables effective communication between the app and the server.

Next, we will present important screens and features of the app to explain how it works and helps users.

4.2-Technology Constraints:

Software needed: Visual Studio Code, Laravel Valet, Linux operating system, Insomnia (API testing tool), DataGrip (Database management tool).

Languages: Dart, PHP.

Database system: MySQL (managed via Laravel migrations and Eloquent ORM).

Frameworks & Libraries: Flutter, Laravel, Laravel Sanctum (for API authentication), REST API.

APIs & Services: Nominatim (OpenStreetMap - for location autocomplete and geocoding).

4.3-project’s Parts

This section presents the main parts of the mobile application through its user interface.
It is divided into three parts: the new user flow, regular user screens, and specialist screens.
Each screen is explained with its purpose, features, and role in the overall user experience.

4.3.1-New User Flow

This section includes the initial screens: splash, login, and registration. They guide new users into the app and handle secure account access.

4.3.1.1-Splash Screen

The splash screen shown in the *Figure 5 - Splash Screen* is the first screen users see; it introduces the app’s identity while loading.The splash screen shows an animated logo of a specialist and a house, symbolizing the app's purpose.It features the slogan **"Home or office, we've got your back"**.The color scheme (Indigo [900], orange, white) reflects trust, energy, and clarity.

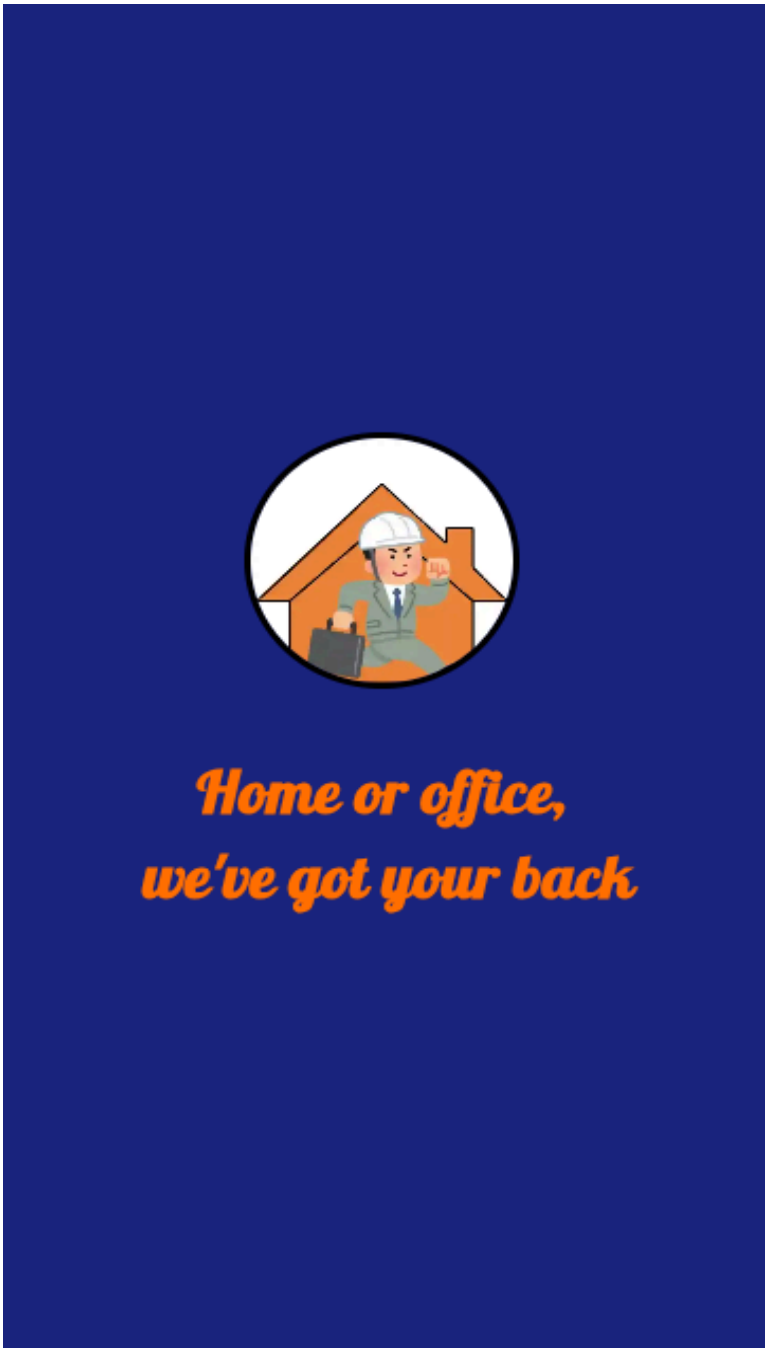

Figure 5 - Splash Screen

4.3.1.2-Login Screen

The login screen includes an app bar titled **"Repair App"**, the logo, and the slogan. It has fields for email and password, a login button, and snackbar messages for errors like invalid credentials or empty fields. A link at the bottom lets users navigate to the sign-up page if they don't have an account.

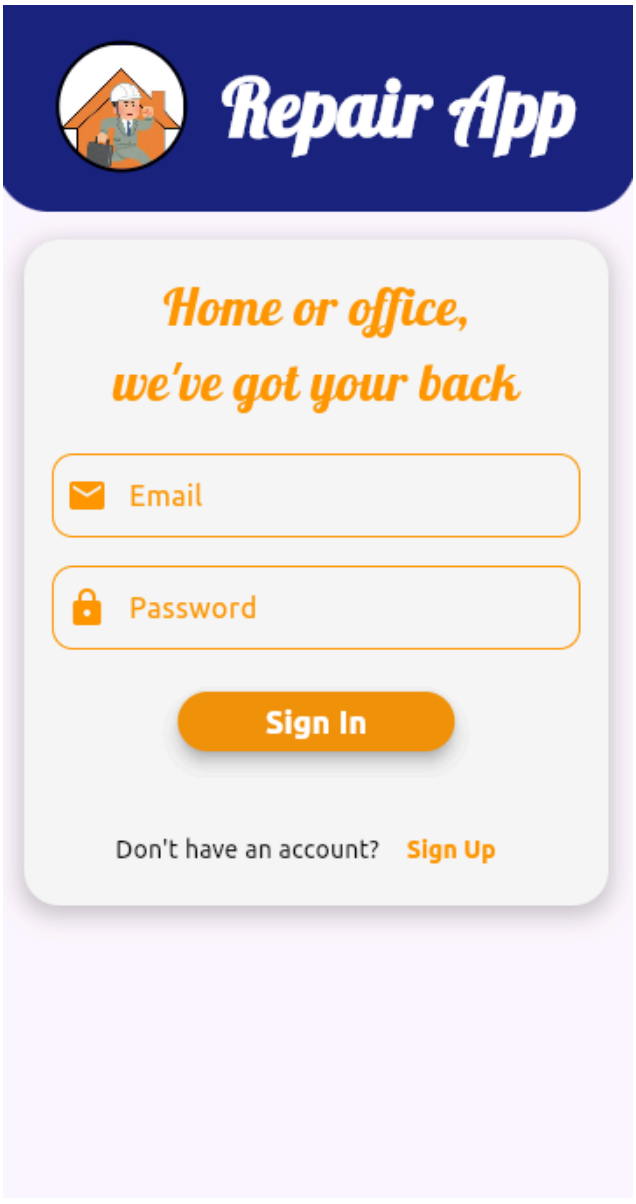

Figure 6 - Login Screen

4.3.1.3-Registration Screen

The registration screen displays the slogan and fields for first name, last name, email, password, and confirm password. Email must be valid, and passwords must be at least 8 characters with uppercase, lowercase, and a number or symbol. Users can toggle between user and specialist roles, and there's a link to return to the login page.

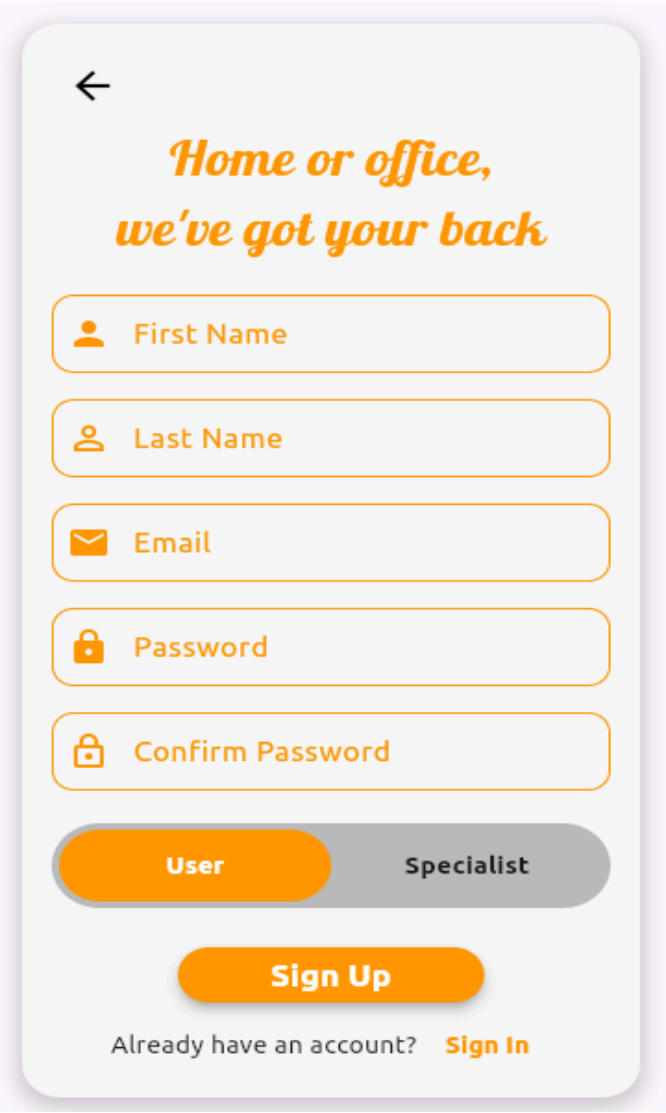

Figure 7 - Registration Screen

4.3.2 User Screens

This section covers the main screens available to regular users.

4.3.2.1-Home Screen

The home screen features an app bar titled **"Home"**, the slogan, and a description explaining the app's purpose. An auto-slider displays images representing the main services: electrician, plumbing, and handyman. At the bottom, a contact section provides the support email and a chat option for user assistance that is shown in the *Figure 9 - User Home Screen – Enrolled View*.

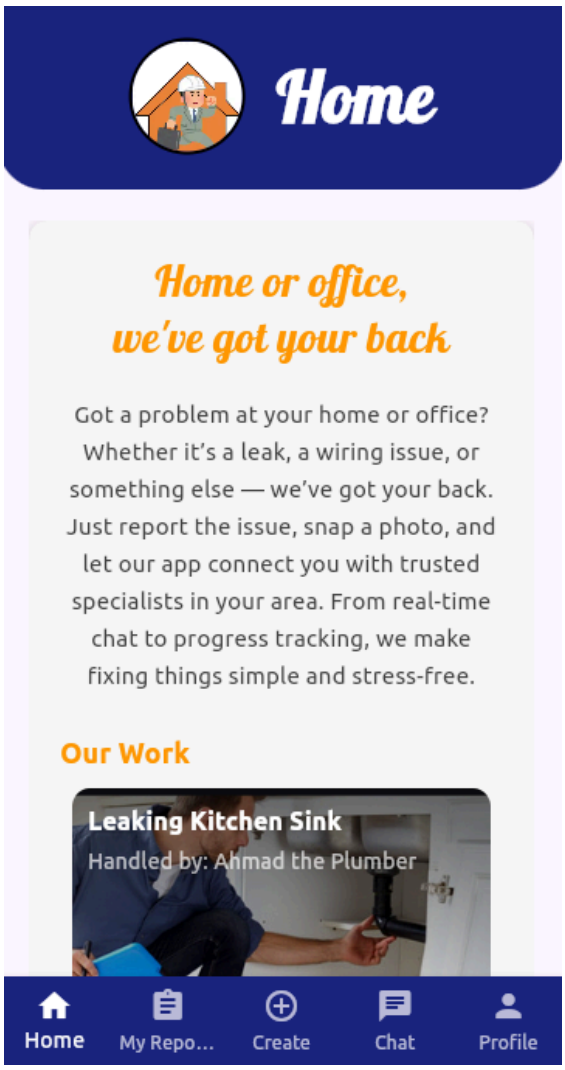

Figure 8 - User Home Screen

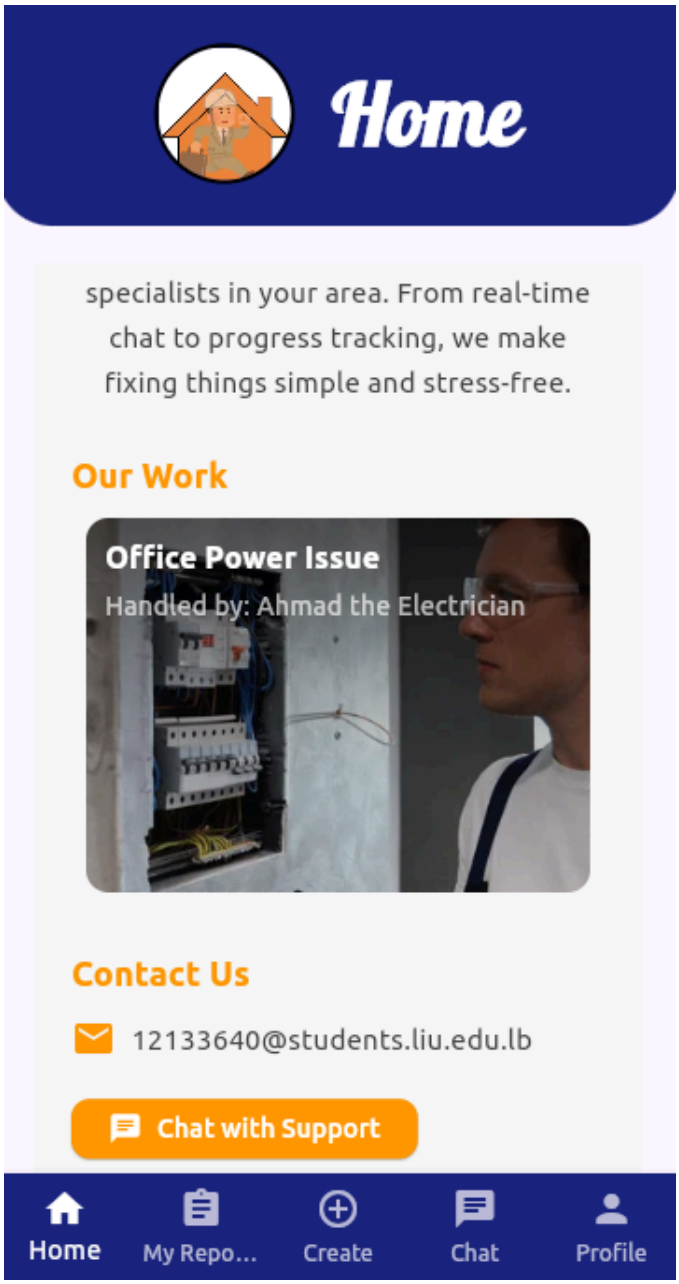

Figure 9 - User Home Screen – Enrolled View

4.3.2.2-Profile Screen

The profile screen has an app bar titled **"Profile"** and lets users view and edit their profile picture, name, and location (with autocomplete). There are buttons to save changes, change password, log out, and delete the account. Account deletion prompts a confirmation message: “Are you sure you want to delete the account?”

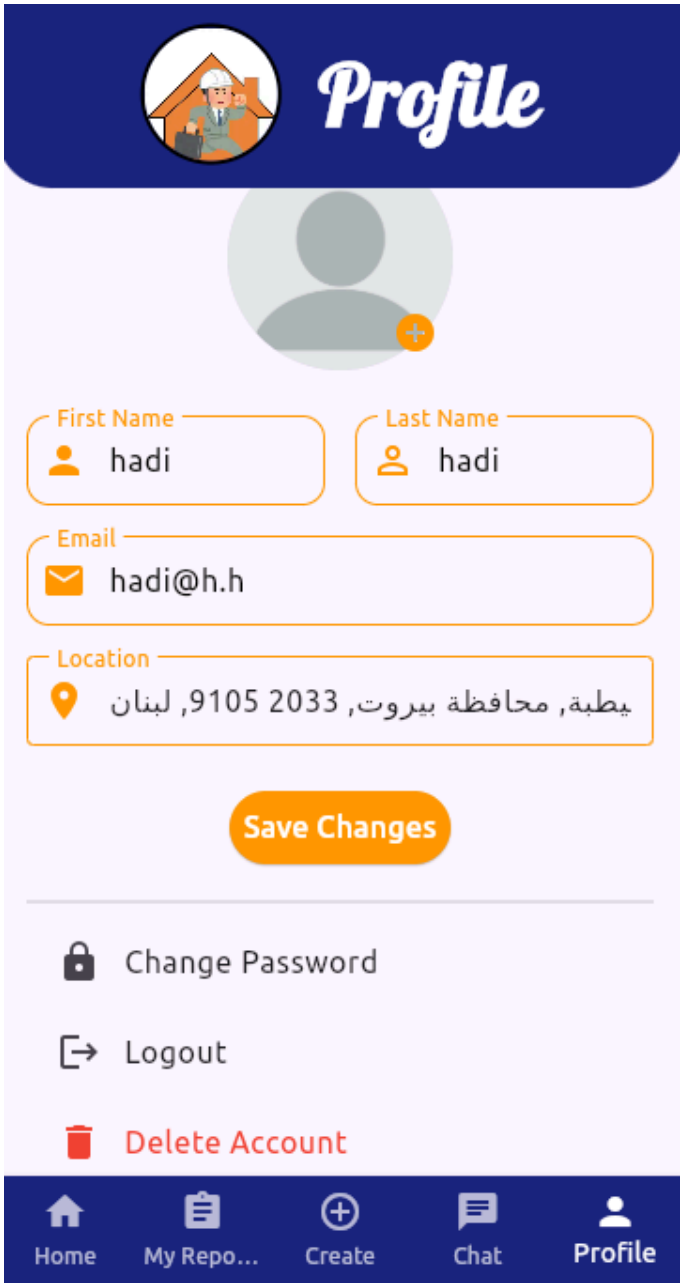

Figure 10 - User Profile Screen

4.3.2.3-Create Report Screen

The screen has an app bar titled **"Create Report"** and includes fields for title, description, and an issue type dropdown (contractor, electrician, plumber, handyman). Users can toggle between using their default location or entering a new one with autocomplete. They can also upload or take images and submit the report using the button at the bottom.

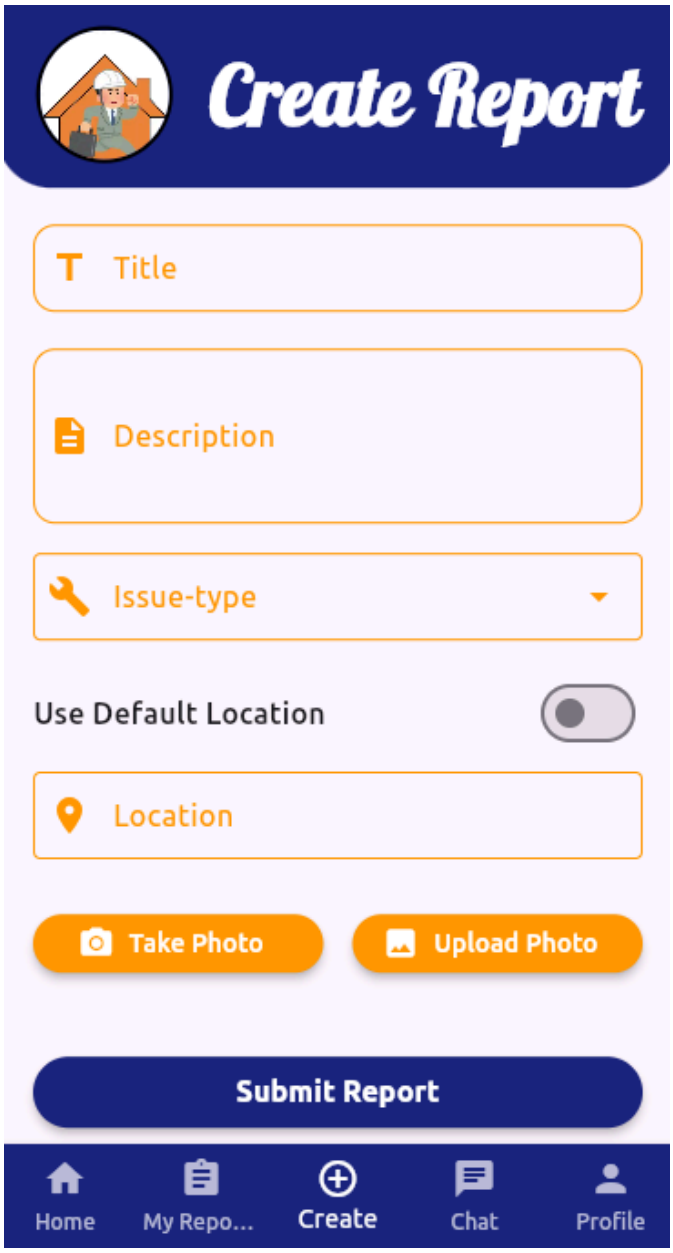

Figure 11 - User Create Report Screen

4.3.2.4-Search For Specialist’s Screen

This screen has an app bar titled **"Specialist List"** and displays specialists filtered by issue type and sorted by distance. Each list item shows the specialist’s name, profile picture, specialization, and location. Users can tap a specialist to view their full profile.A button is also available to navigate to the My Reports screen.

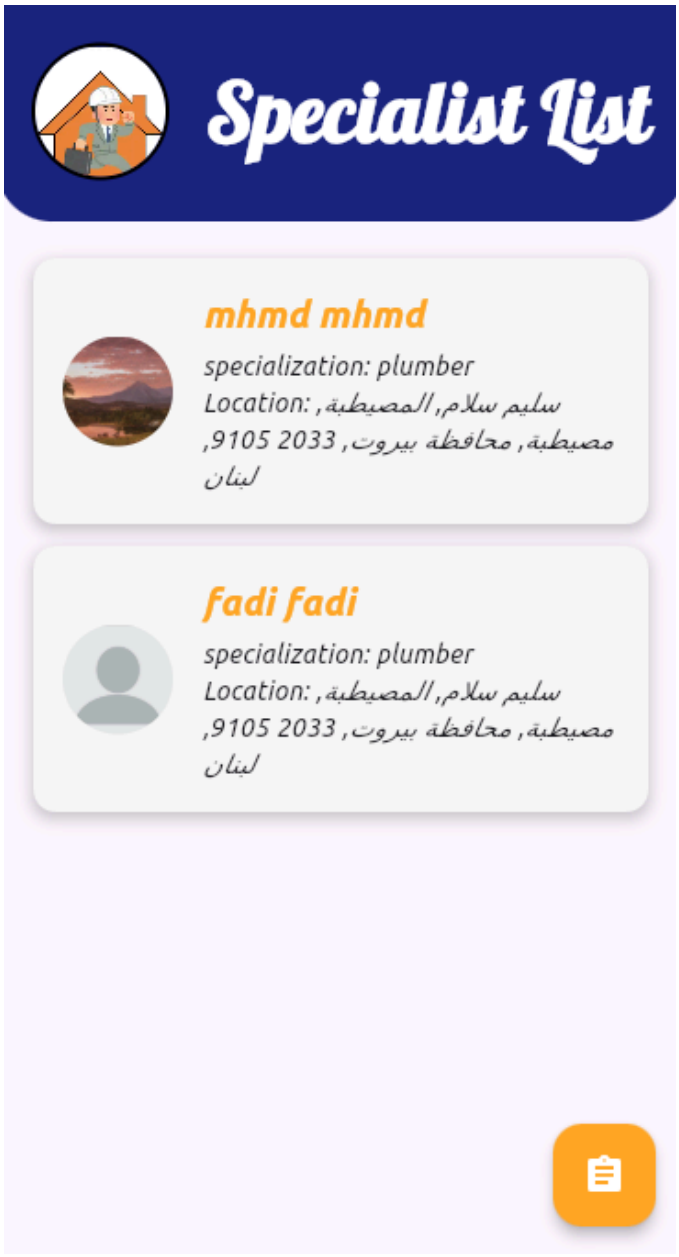

Figure 12 - Search For Specialist Screen

4.3.2.5-Specialist Profile View

Users can view the specialist's profile, including name, profile picture, specialization, email, bio, and feedback from other users. A **chat** button allows direct messaging, and a **select** button lets the user assign the job to the specialist. After selection, the user must wait for the specialist's response to confirm the assignment.

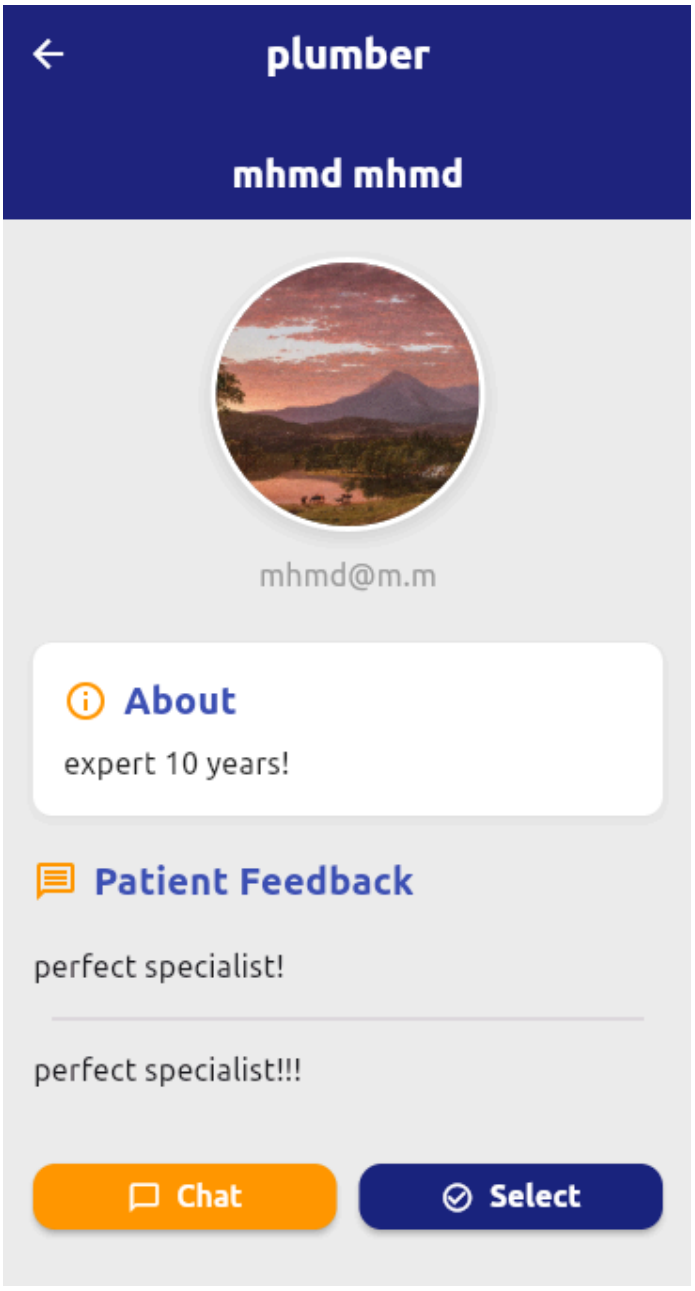

Figure 13 - Specialist Profile View

4.3.2.6-My Report Screen

This screen has an app bar titled **"My Reports"** and shows a list of the user's submitted reports.

Each item displays the report's title, date, description, and current status. Statuses include: **Waiting** – awaiting the specialist's response, **Rejected** – specialist declined the job, **In Progress** – specialist accepted and is working on the job, **Escalated** – specialist accepted but stopped without completing, **Completed** – the job was finished successfully.

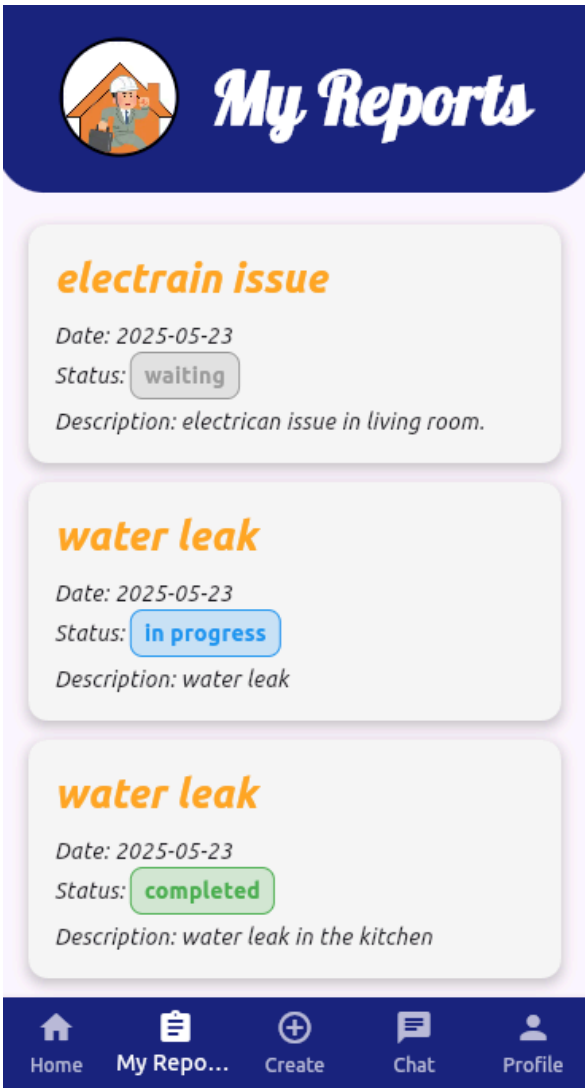

Figure 14 - My Reports Screen

4.3.2.7- Report Action Screen

This screen appears when the user opens a report with status **Waiting**, **Rejected**, or **Escalated**. It shows the report title, a **Delete Report** button with confirmation, and an option to assign the report to a new specialist. This allows the user to either remove the report or try again with another specialist.

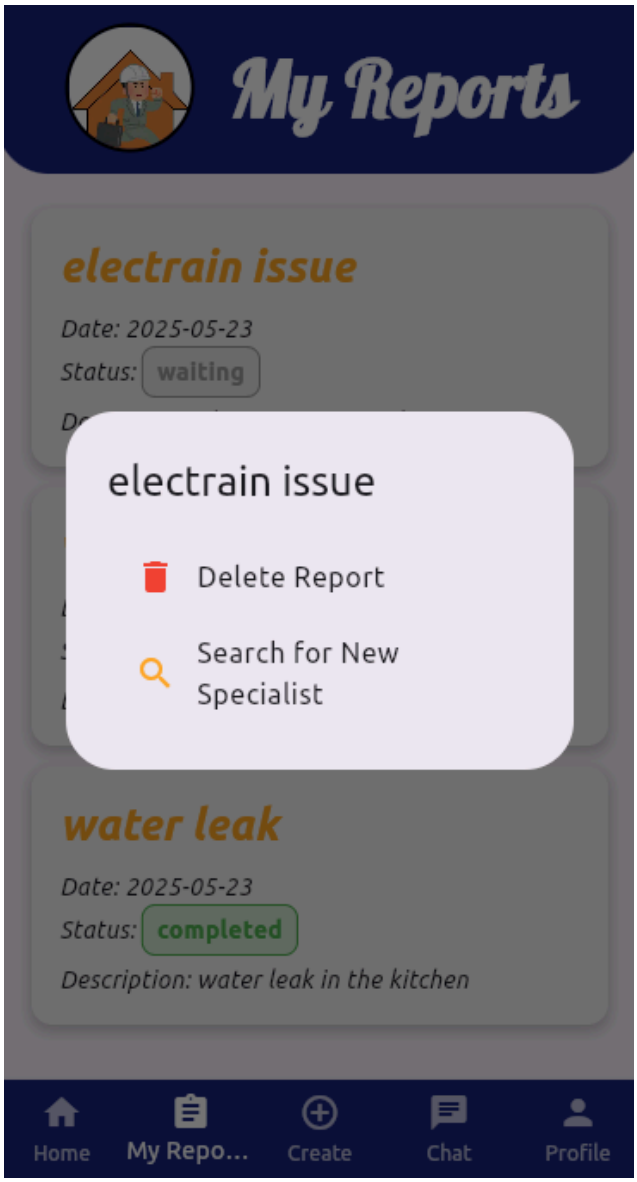

Figure 15 - Report Action Screen

4.3.2.8- Job Tracking Screen

This screen appears when the user opens a report with **In Progress** status. It shows the report details and displays job updates uploaded by the specialist. Each update includes an image and a comment to track the progress visually and descriptively.

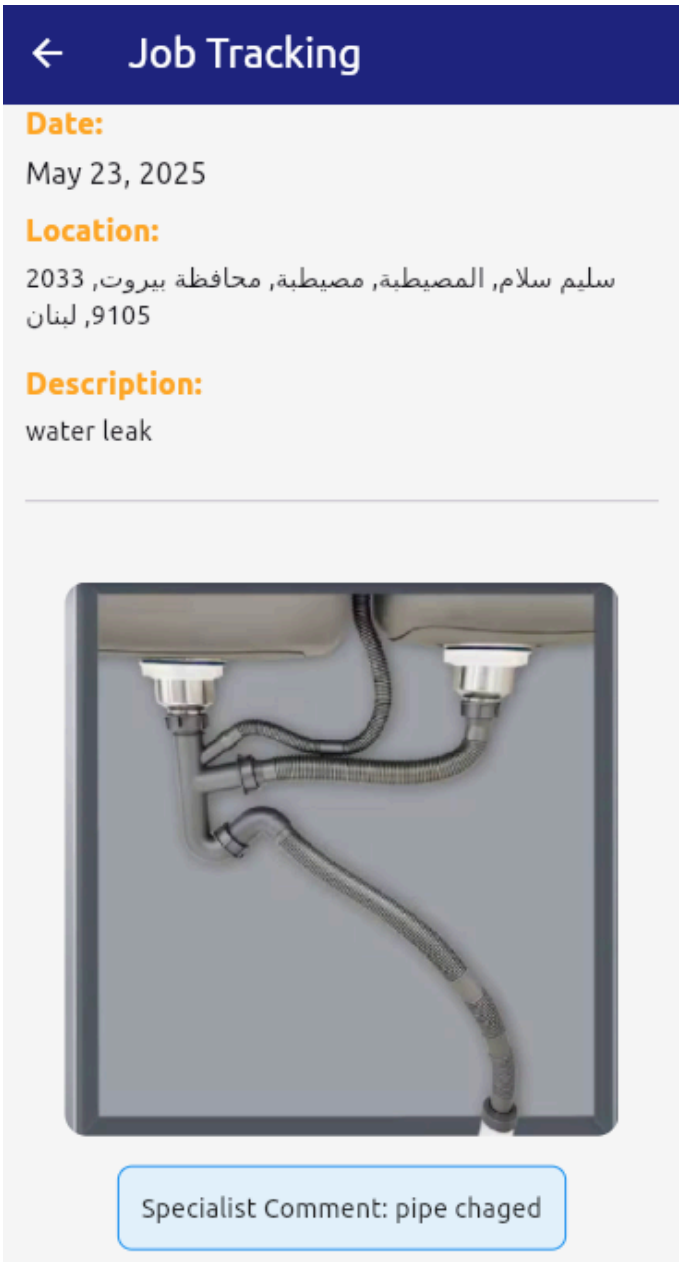

Figure 16 - Job Tracking Screen

4.3.2.9- Provide Feedback Screen

This screen appears when the user opens a report with **Completed** status. It allows the user to write and submit feedback for the specialist after the job is finished.

This helps maintain service quality and assist future users in choosing specialists.

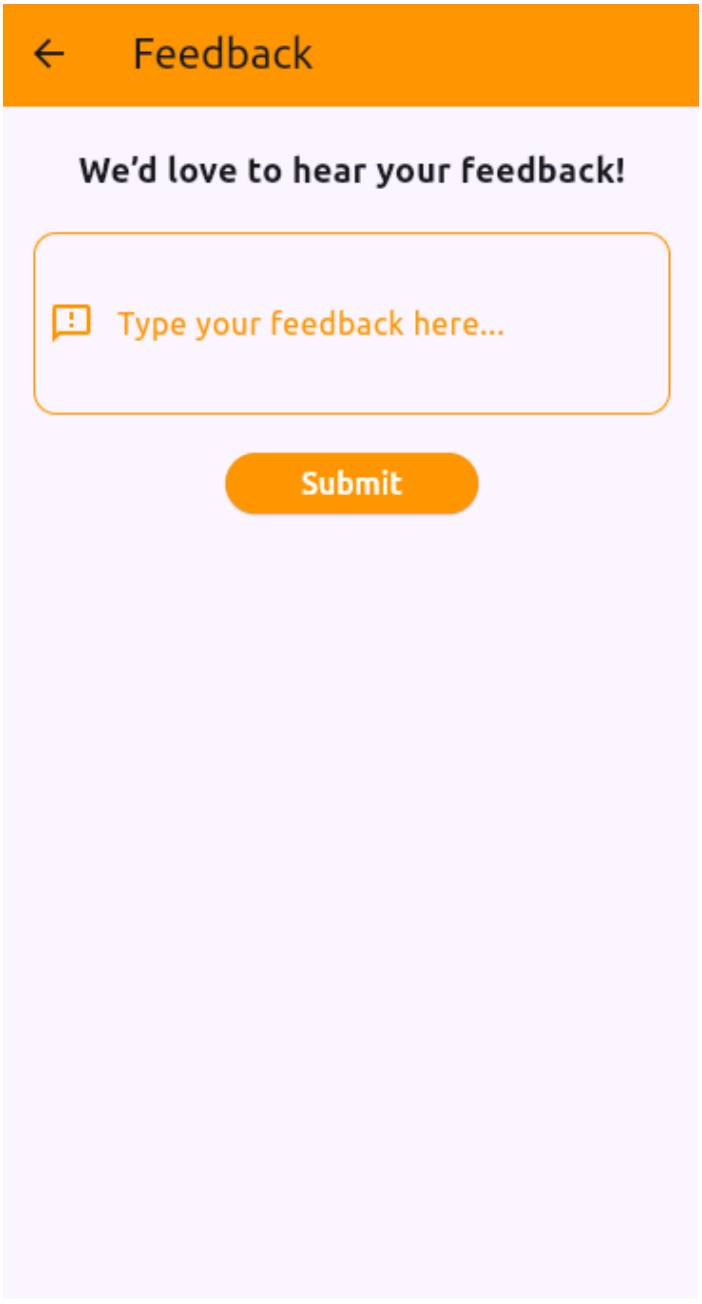

Figure 17 - Provide Feedback Screen

4.3.2.10- Chat List Screen

This screen displays a list of users the person has chatted with. Each item shows the user's name, profile picture, last message, and date. Tapping a user opens the chat to continue the conversation.

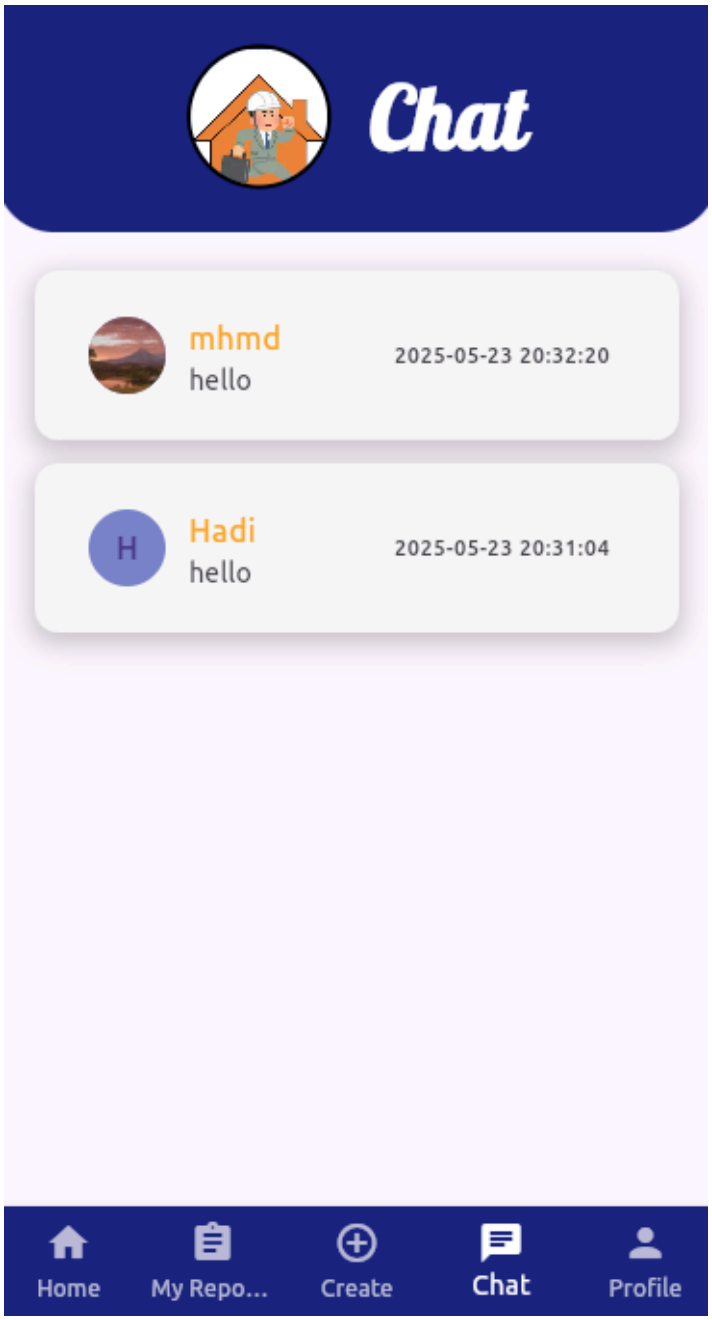

Figure 18 - Chat List Screen

4.3.2.11- Chat Screen

This screen shows the chat with a selected user, displaying their name and past conversation. Users can send new messages and see the date for each message. It supports ongoing communication between users and specialists.

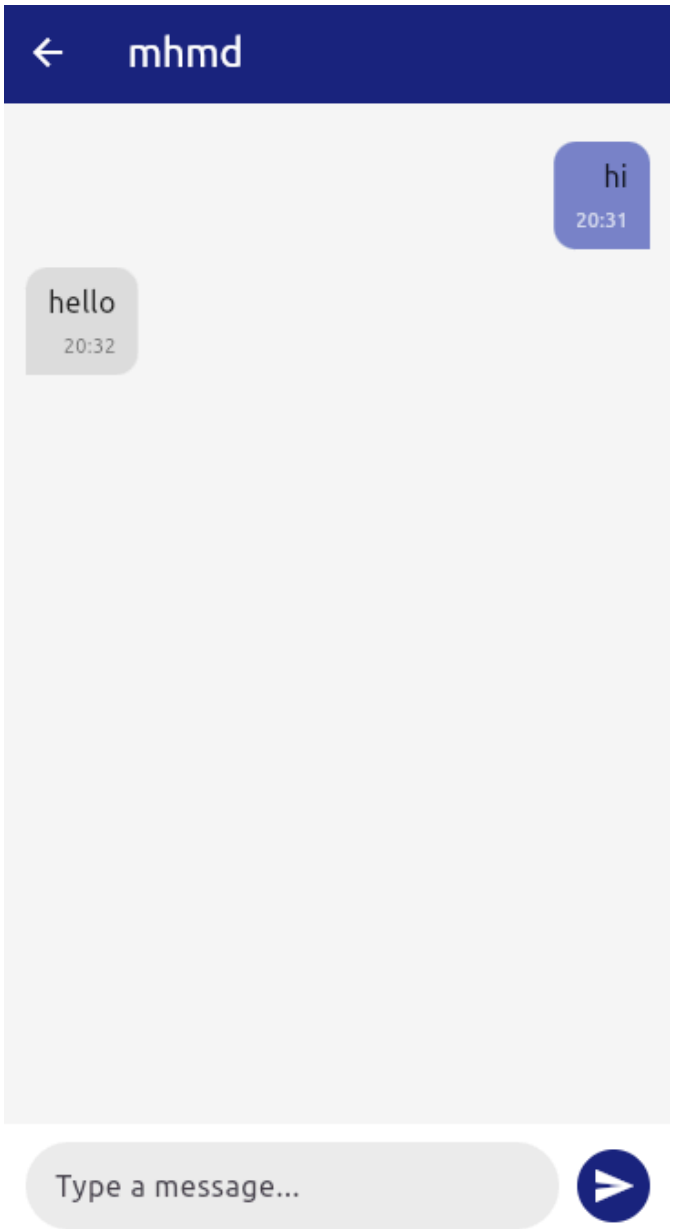

Figure 19 - Chat Screen

4.3.3 Specialist Screens

This section presents the screens designed for specialists.

4.3.3.1- Specialist Home Screen

This screen welcomes specialists with the motivational message: **"Keep me busy, I'm built for it."** It includes a description highlighting trust, honesty, dedication, and pride in their work ethics. A slider showcases services offered, and a button links to the contact us section for support.

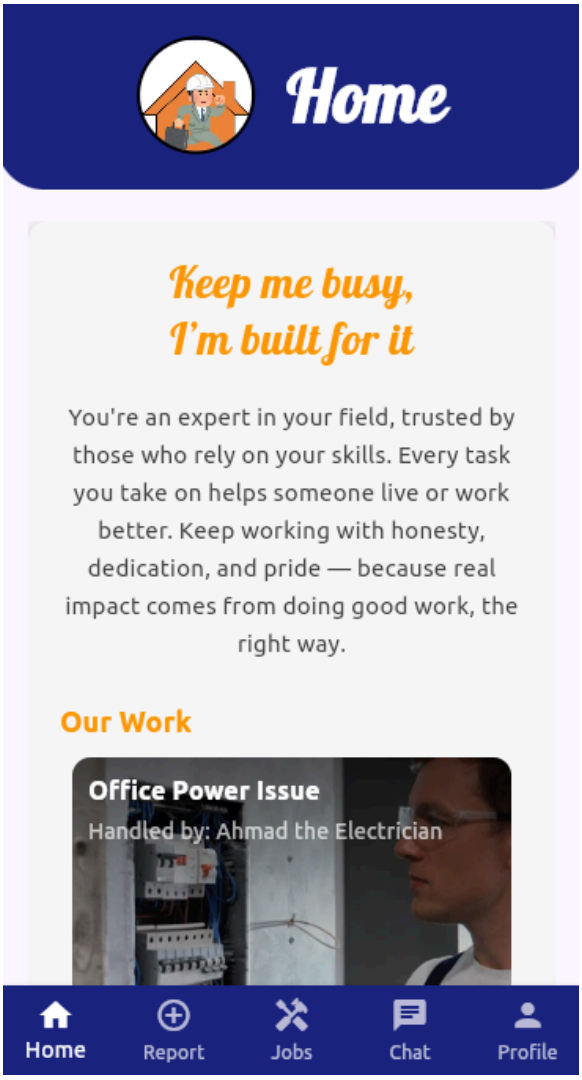

Figure 20 - Specialist Home Screen

4.3.3.2- Specialist Profile Screen

This screen allows the specialist to view and edit their profile details, similar to the user profile. It includes additional fields for **specialization** (contractor, electrician, plumber, handyman) and a **bio**. Specialists can also view feedback written about them by users.

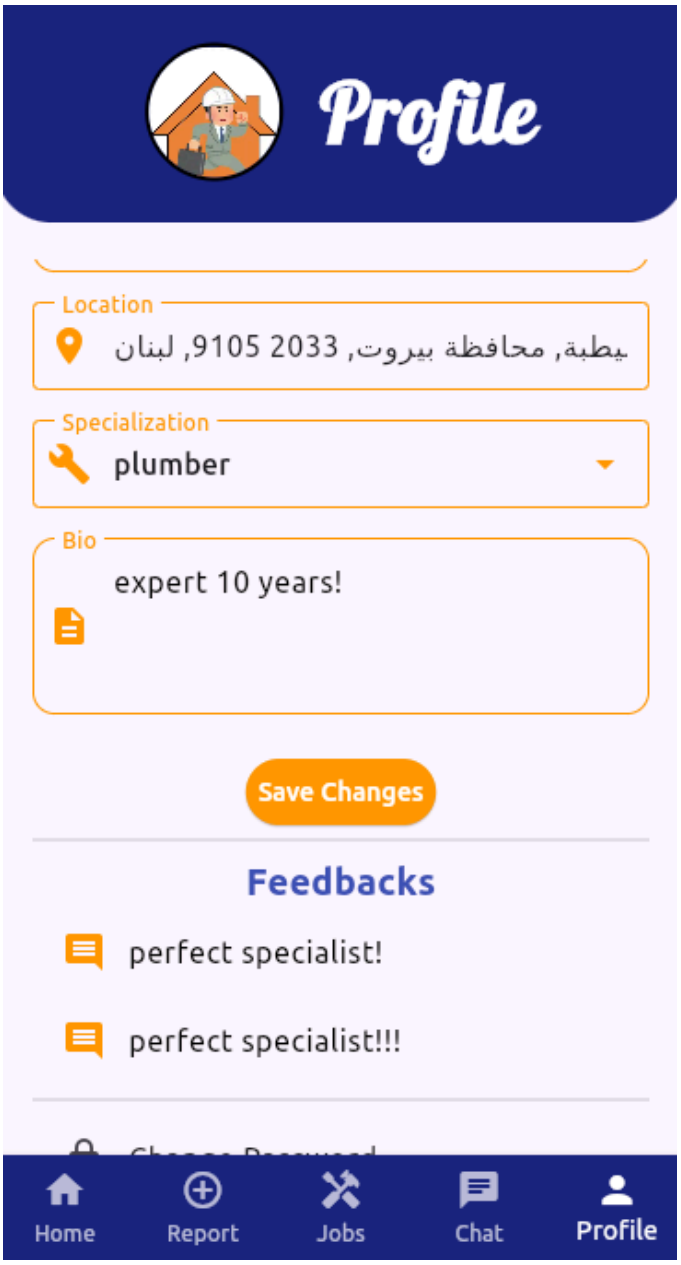

Figure 21- Specialist Profile Screen

4.3.3.3- Job's List Screen

This screen has an app bar titled **"Jobs"** and displays reports assigned to the specialist. Each item shows the report title, date, description, location, images, and status (**Waiting** or **In Progress**). Specialists can tap a report to view its details and update the status.

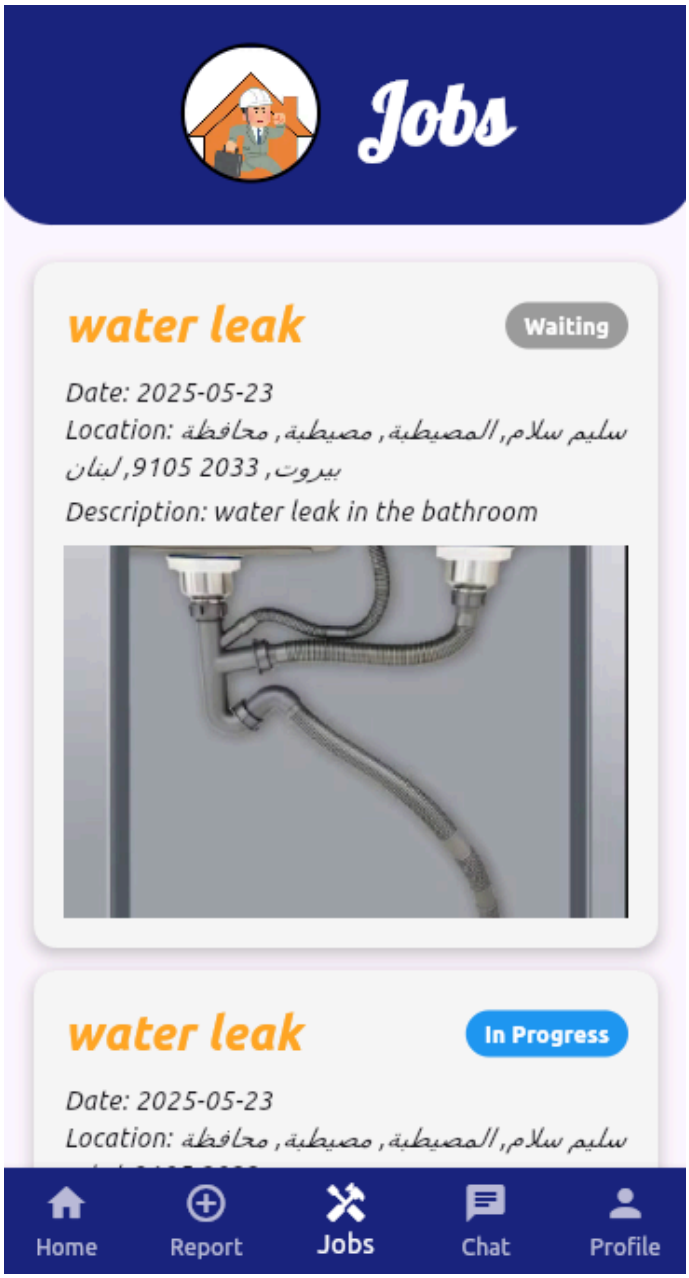

Figure 22 - Job's List Screen

4.3.3.4- Job Details Screen

This screen appears when a specialist opens a report with **Waiting** status. It shows the report's title, date, location, user's name and email, description, and images. The specialist can choose to **Accept** or **Reject** the job using the available buttons.

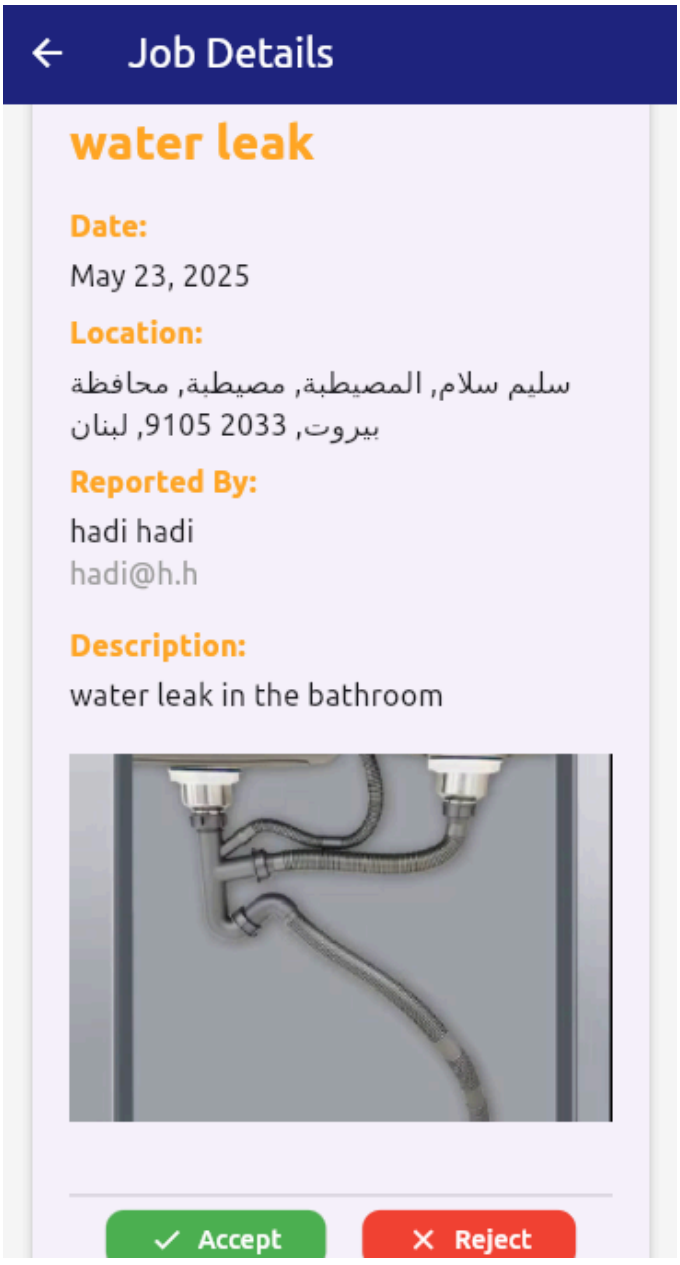

Figure 23 - Job Details Screen

4.3.3.5- Job Progress Screen

This screen appears when a specialist opens a report with **In Progress** status. It shows the report details and allows the specialist to upload progress images and comments. The specialist can also choose to **complete** the job or **stop** it without completing it.

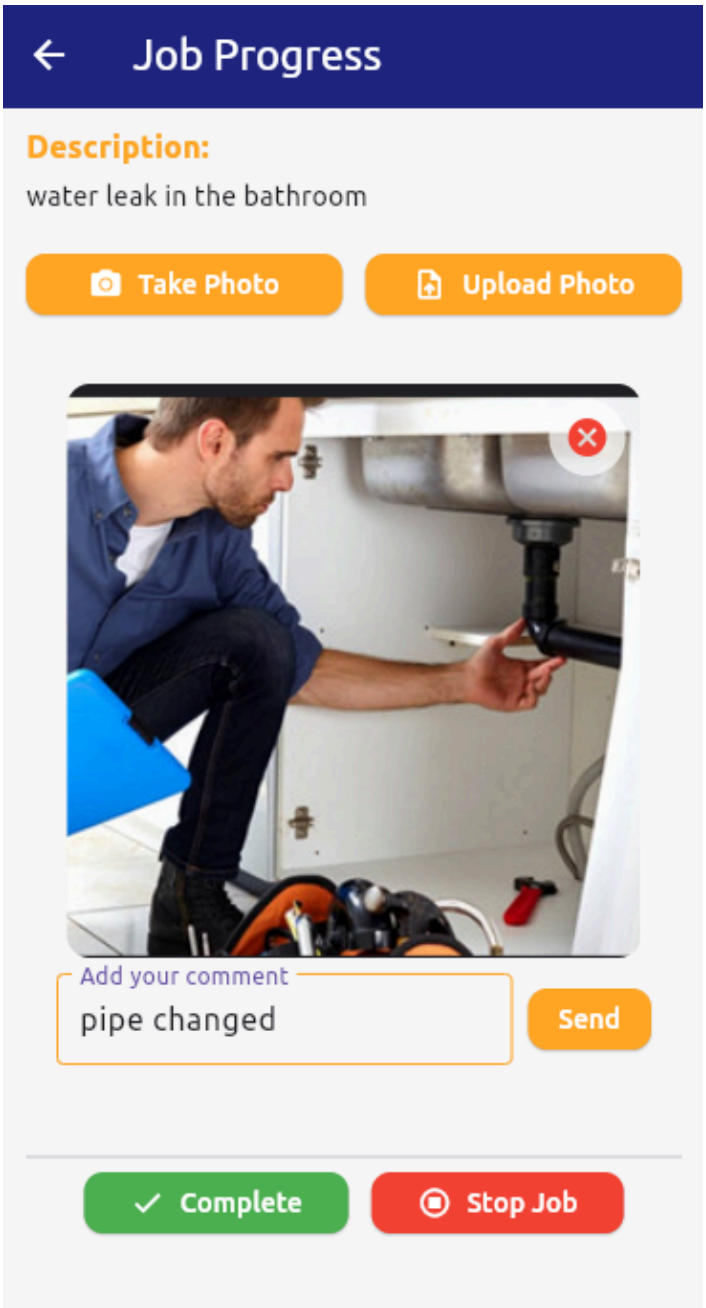

Figure 24 - Job Progress Screen

4.3.3.6- Report's Screen

This screen allows the specialist to act as a user by creating reports in the same way. It includes the report creation form and lets the specialist scroll to view their own reports. Reports are displayed with full details, just like in the **My Reports** screen for users.

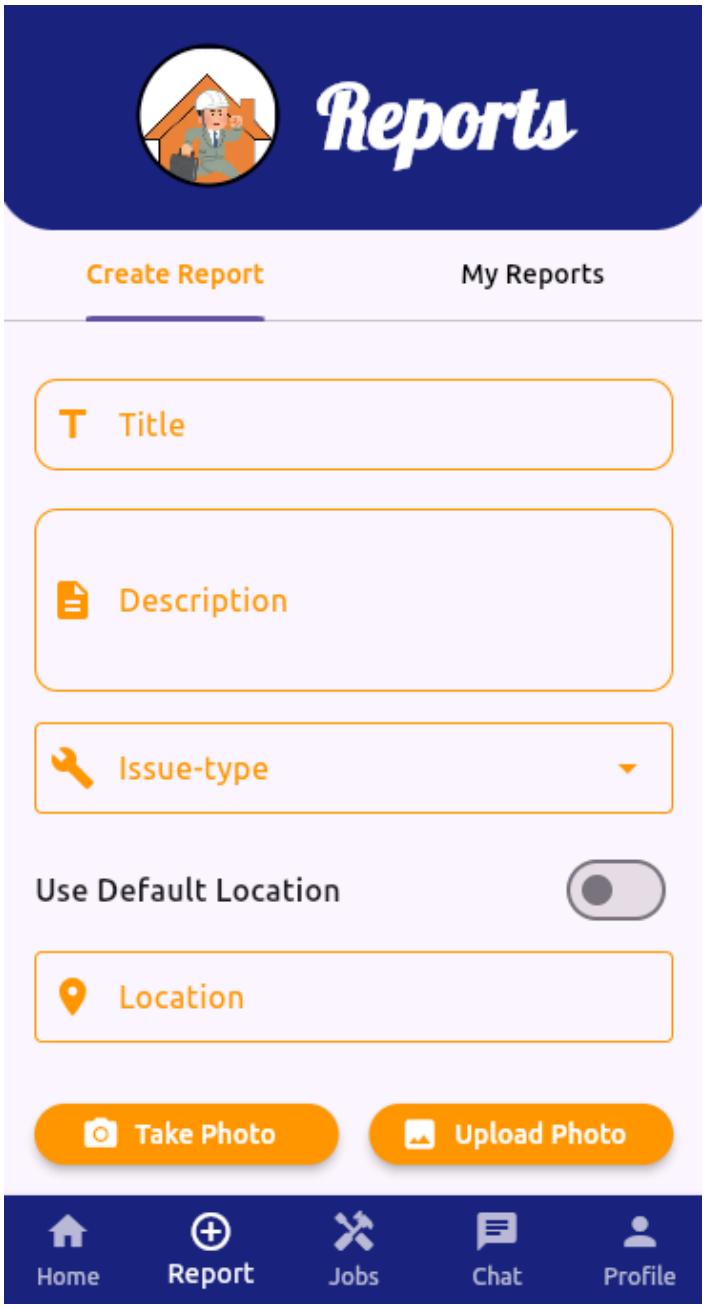

Figure 25 - Specialist Report's Screen

Chapter 5

5.1 Conclusion

This project successfully delivered a mobile application that connects users with trusted repair specialists for various home and office services.

It offers a smooth user experience with essential features such as report creation, specialist browsing, real-time job tracking, feedback, and communication.

By providing a reliable and user-friendly platform, the app addresses the challenge of finding skilled professionals quickly and efficiently, improving convenience and service quality for users and specialists alike.

5.2 Future Work:

If time and resources permit, several features can be added in future development to improve the application's functionality, trust, and management:

- **AI Report Classification**: Implement an AI assistant to automatically suggest the issue type by analyzing uploaded report images.

- **Admin Dashboard**: Develop an admin panel to monitor users, specialists, reports, and job activities.

- **Specialist Verification**: Require specialists to upload official government documents (e.g., a clean criminal record) for admin review and approval before account activation.

- **Specialist Subscription System**: Introduce a monthly payment system for specialists to access and receive job requests.

- **User Complaint Mechanism**: Allow users to report serious issues such as dishonesty about job duration or fees. Specialists who receive repeated verified complaints may be warned up to three times before account termination by the admin.

These enhancements would help strengthen platform reliability, streamline operations, and ensure quality service for all users.

5.3 Reference

- Flutter Documentation – https://docs.flutter.dev
- Laravel Documentation – https://laravel.com/docs
- Nominatim (OpenStreetMap) API – https://nominatim.org/release-docs/latest/api/Overview/
- Dart Language Documentation – https://dart.dev/guides