HelpMate

Home Service Provider App

A Cross-Platform Flutter Application

CSE 2216 - Application Development Lab

Batch : 29 / 2nd Year 2nd Semester 2024

Team Name :

Team MechaBytes

Submitted By

48 - Farhana Alam

59 - Jubair Ahammad Akter

11 - Md Shakin Alam Kabbo

57 - N. M Rashidujjaman Masum

Department of Computer Science and Engineering

University of Dhaka

Submitted On

July 18, 2025

Table of Contents

1 Introduction

1.1 About the App

The **Home Service Provider App** (HelpMate) is a cross-platform application that connects users with trusted home service professionals (such as cleaners, electricians, plumbers, and more) and allows seamless booking, management, and review of services.

1.2 Motivation

Finding reliable home service providers is often a challenge. Our motivation was to create a unified, easy-to-use platform that bridges the gap between customers and skilled professionals, ensuring convenience, trust, and efficiency for both parties.

1.3 Features

- Cross-platform : Available on Android, iOS, Web, Windows, macOS, and Linux (built with Flutter)
- User-friendly and modern UI
- Service provider and user registration
- Secure authentication and role-based access
- Service booking, scheduling, and order management
- Real-time notifications for service updates
- User reviews and ratings for providers
- Admin panel for managing users, providers, and categories
- Multi-user and multi-device support
- RESTful backend API (FastAPI, Python)
- SQLite/PostgreSQL database support

1.4 Tools, Technologies and Frameworks Used

- **Frontend :** Flutter (Dart)
- **Backend :** FastAPI (Python)
- **Database :** SQLite (development), PostgreSQL (production-ready)
- **Authentication :** JWT, FastAPI security
- **Other :** REST API, Provider (Flutter state management), Docker (optional), GitHub Actions (CI/CD)

1.5 Individual Responsibilities

1.5.1 Farhana Alam

- Led the overall project design and architecture
- Developed the complete backend API using FastAPI (Python)
- Implemented database models, migrations, and SQLAlchemy ORM
- Integrated JWT authentication and role-based access control
- Built RESTful APIs for user management, service booking, and notifications
- Integrated SSLCommerz payment gateway for secure transactions
- Implemented email notification system for service updates
- Coordinated team tasks and managed the GitHub repository

1.5.2 Jubair Ahammad Akter

- Designed and implemented the main user interface in Flutter
- Developed the home screen, categories screen, and navigation
- Integrated authentication flows and user registration screens
- Implemented service provider listing and search functionality
- Built the admin panel interface and dashboard
- Created order history and payment interface
- Assisted with frontend-backend API integration
- Wrote documentation and prepared the final report

1.5.3 N. M Rashidujjaman Masum

- Built the booking and order management screens
- Developed the profile management and settings UI
- Created user reviews and ratings interface
- Helped with responsive design for different screen sizes
- Implemented service status tracking and updates
- Implemented service category filtering and search

1.5.4 Md Shakin Alam Kabbo

- Developed notification screens and user messaging
- Built service provider registration and profile screens
- Implemented real-time order tracking and status updates
- Assisted with UI polish and bug fixing
- Created user feedback and support interface

2 Design and Implementation

2.1 UI Developed (Screenshots)

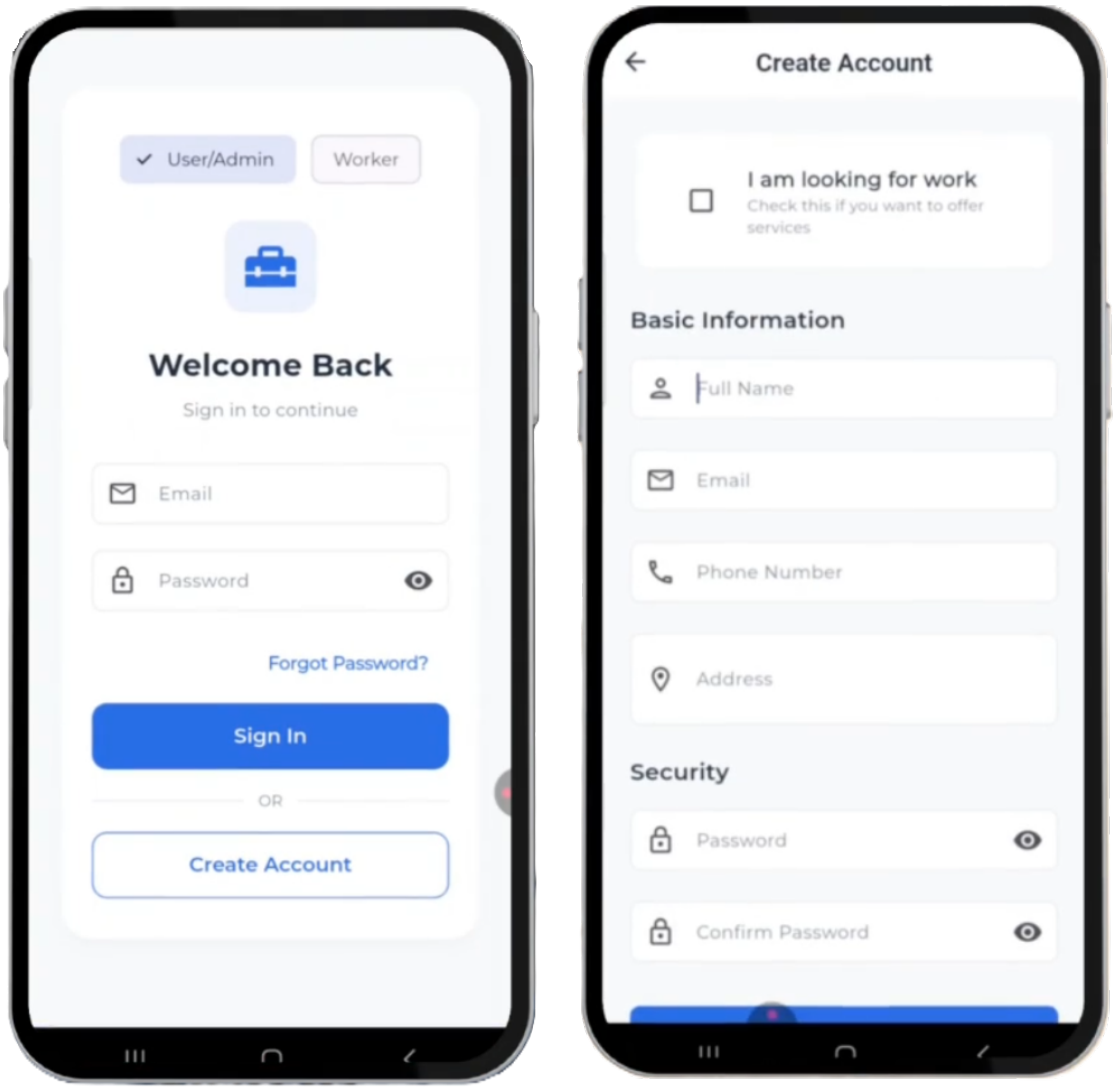

FIGURE 1 – Login / Registration Screen

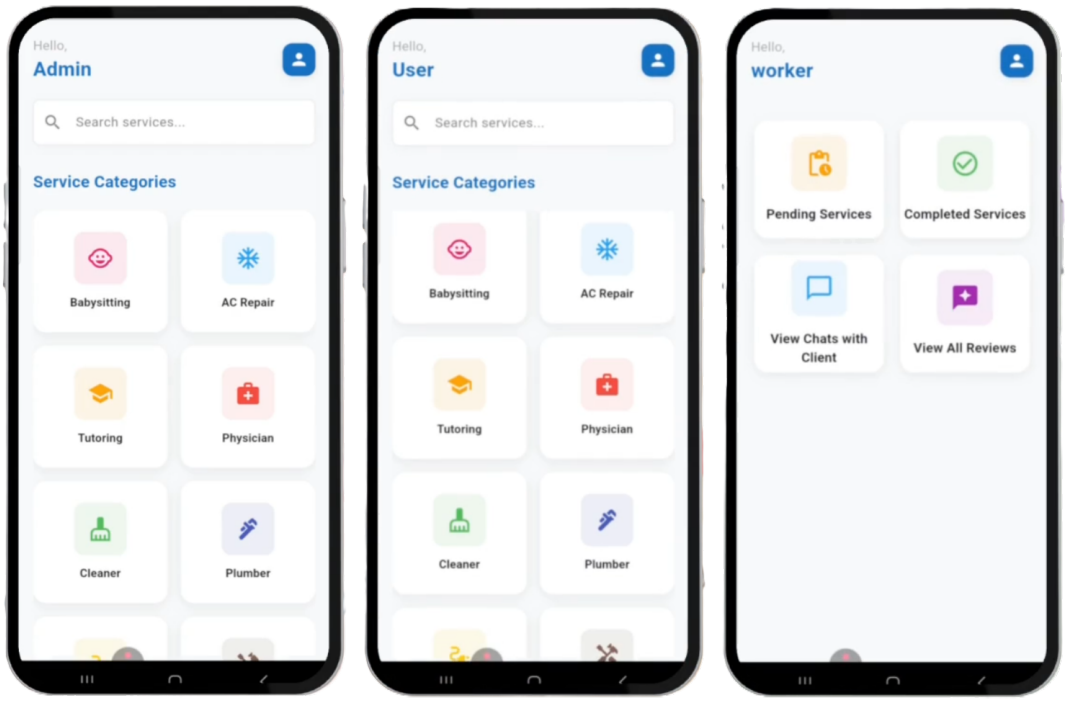

FIGURE 2 – Home Screen for Admin, User and Worker

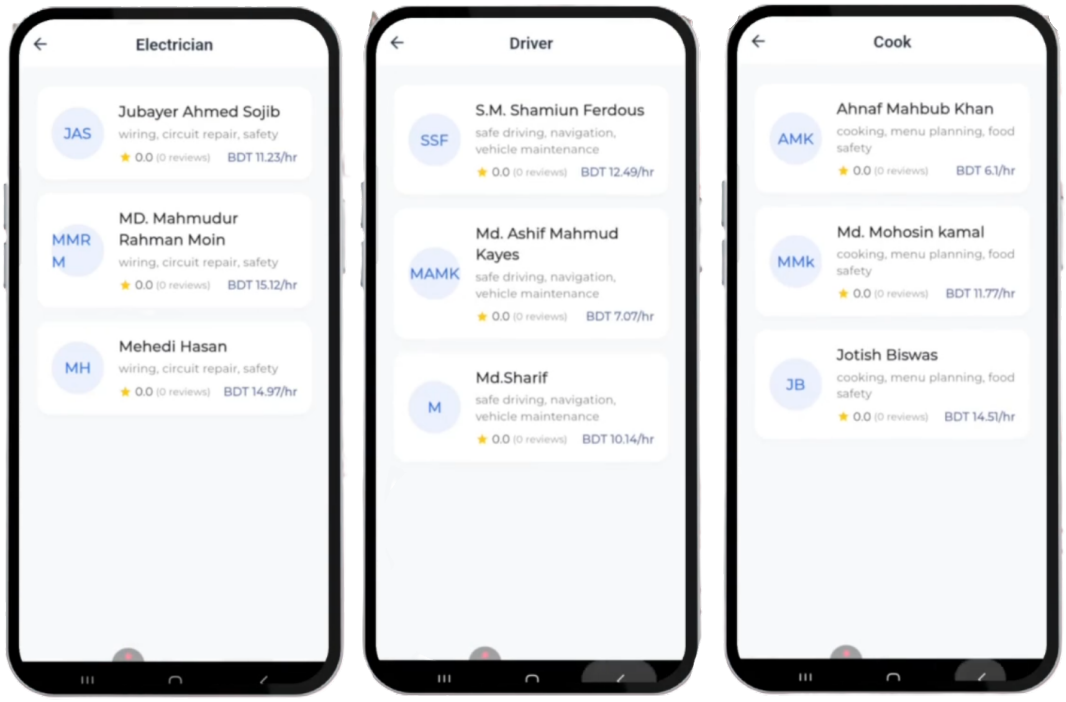

FIGURE 3 – Category wised Workers

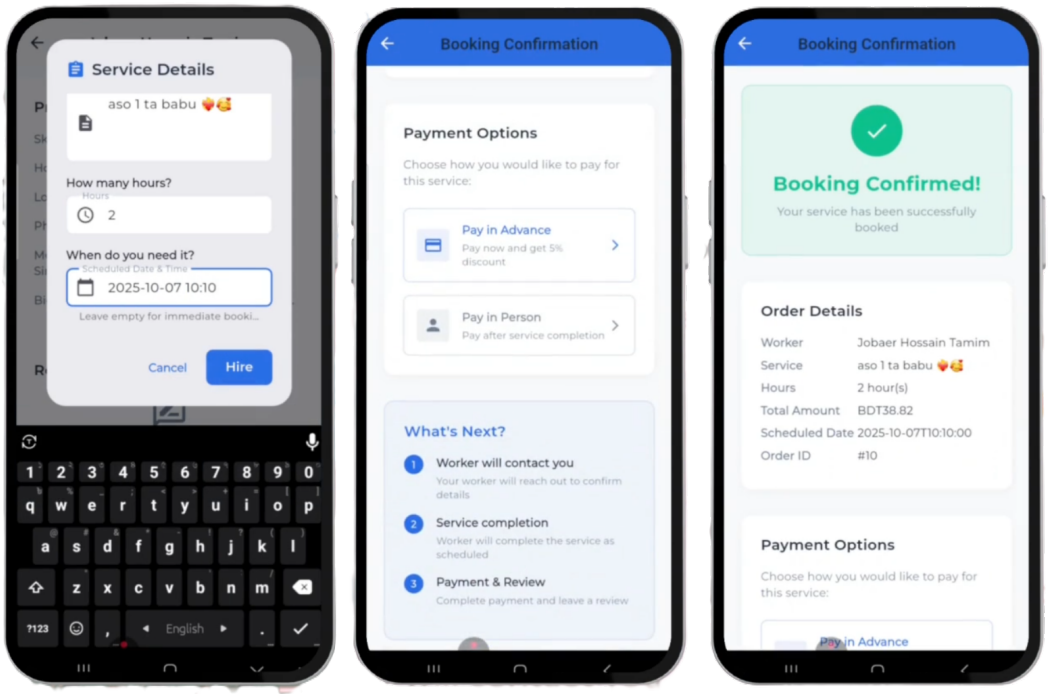

FIGURE 4 – Booking Screen

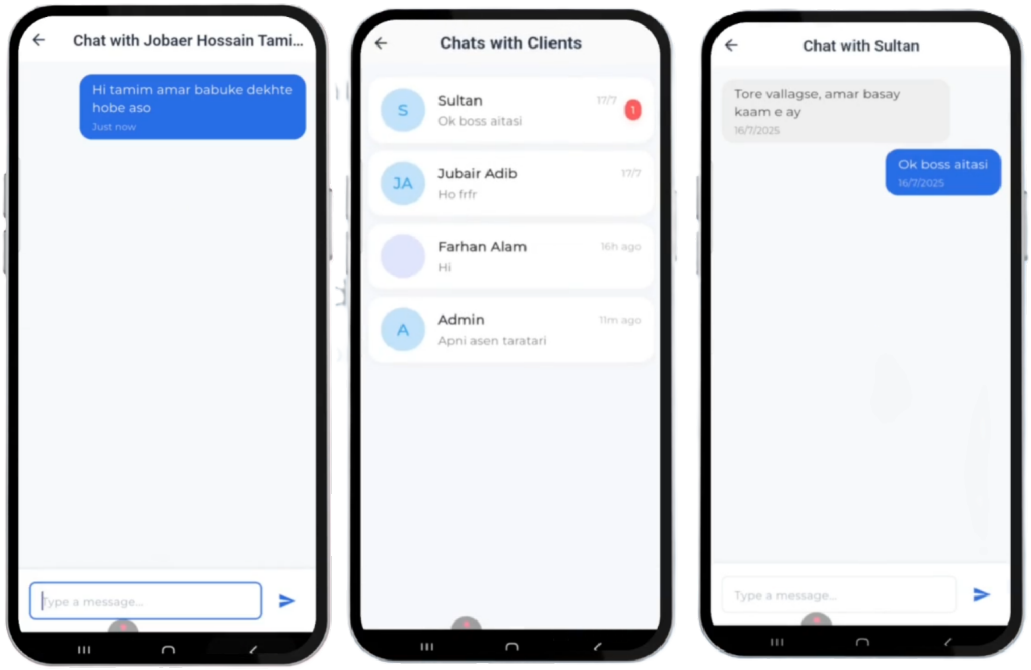

FIGURE 5 – Chat System for Admin, User or Worker

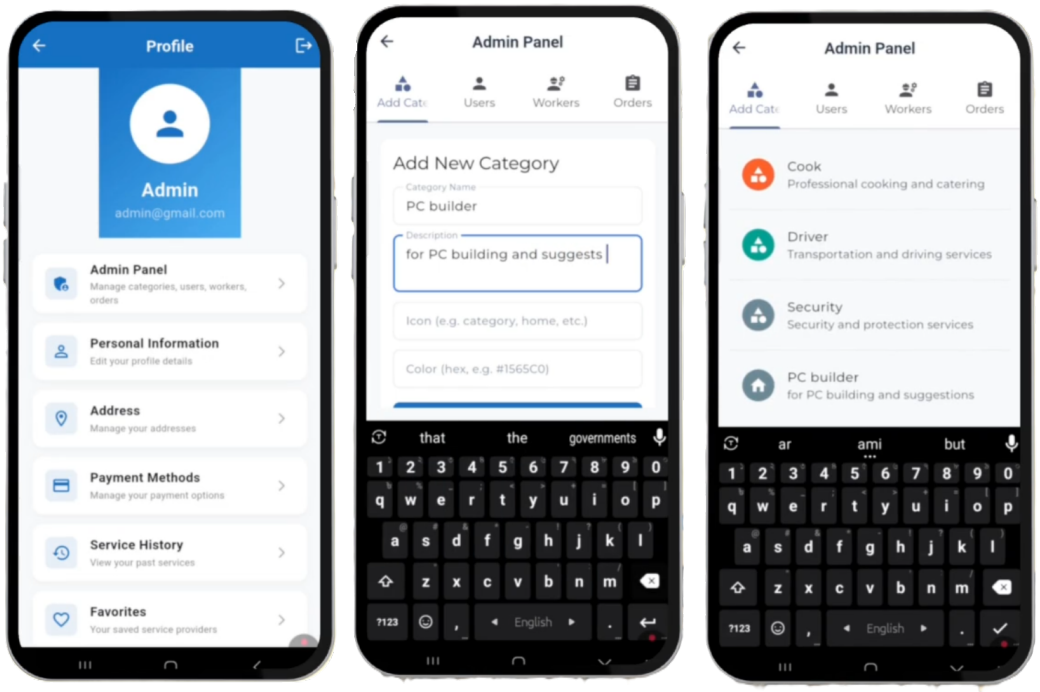

FIGURE 6 – Admin Profile, Admin Panel, Add Category

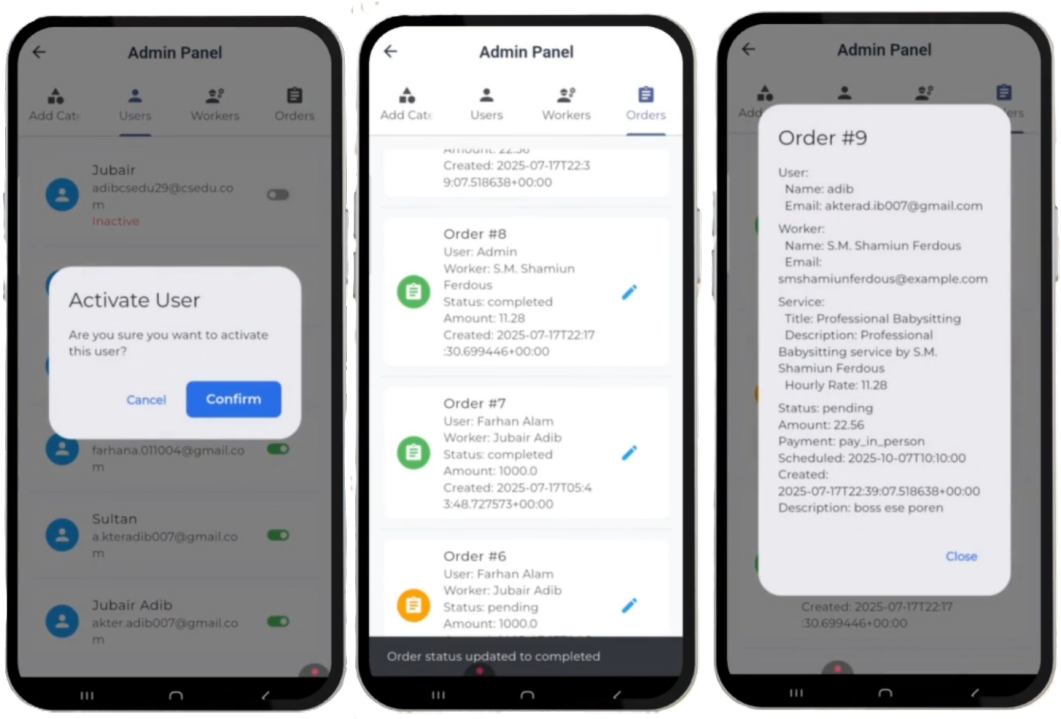

FIGURE 7 – Admin Panel- View Users and Workers, Activate and Deactivate them, View Orders

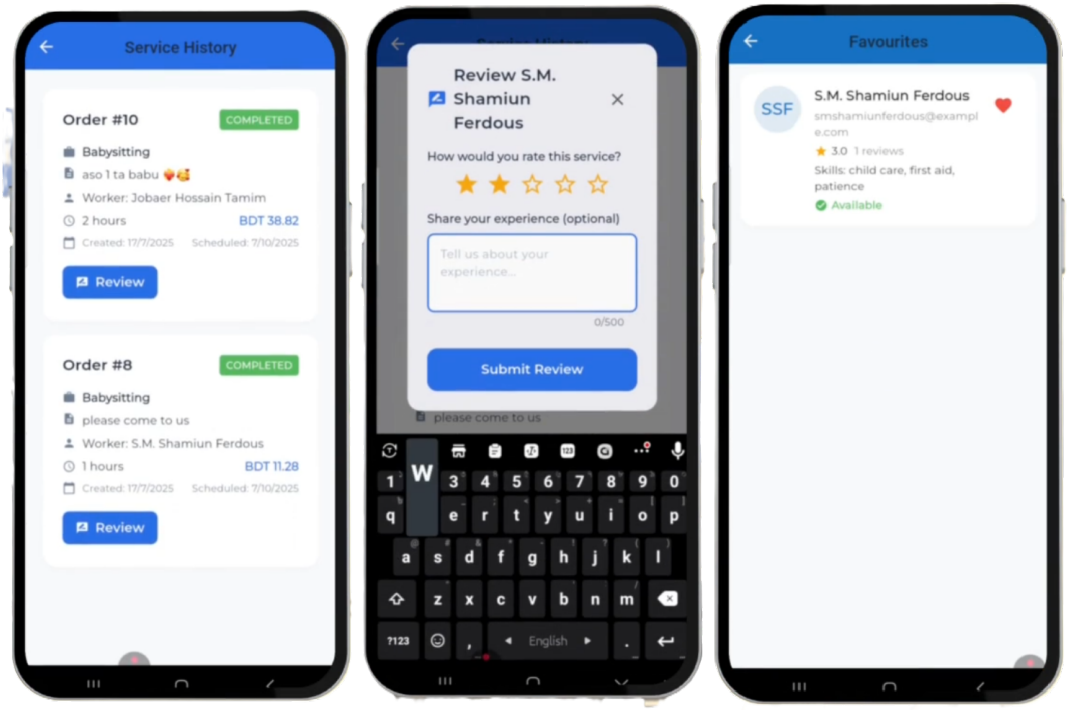

FIGURE 8 – Service History, Review of Workers and Favourites, workers

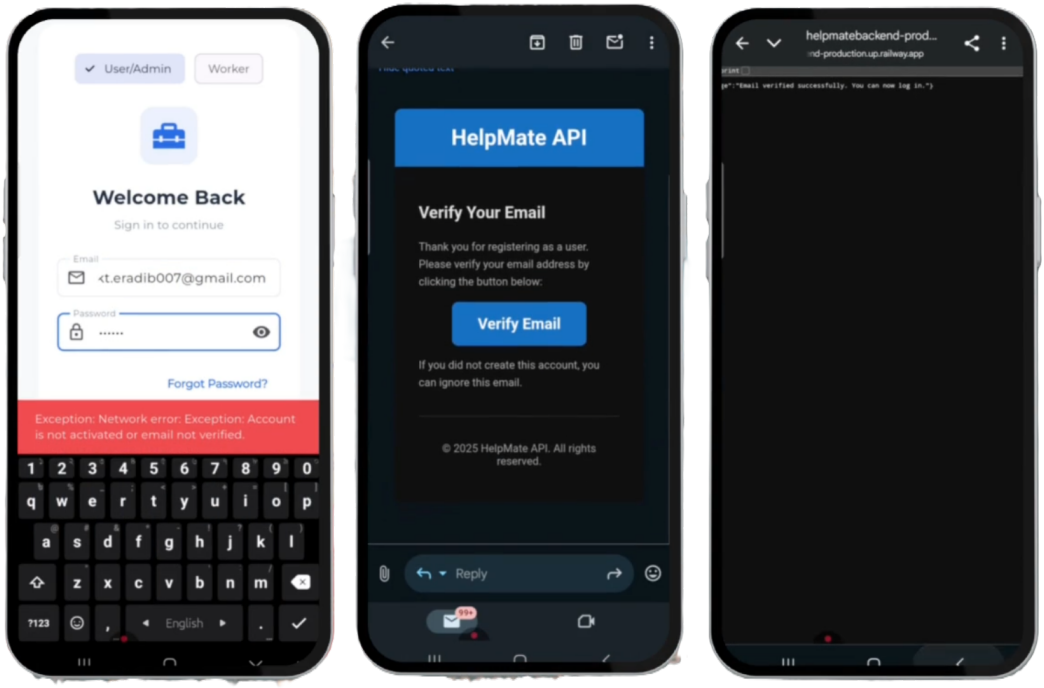

FIGURE 9 – E-mail Verification

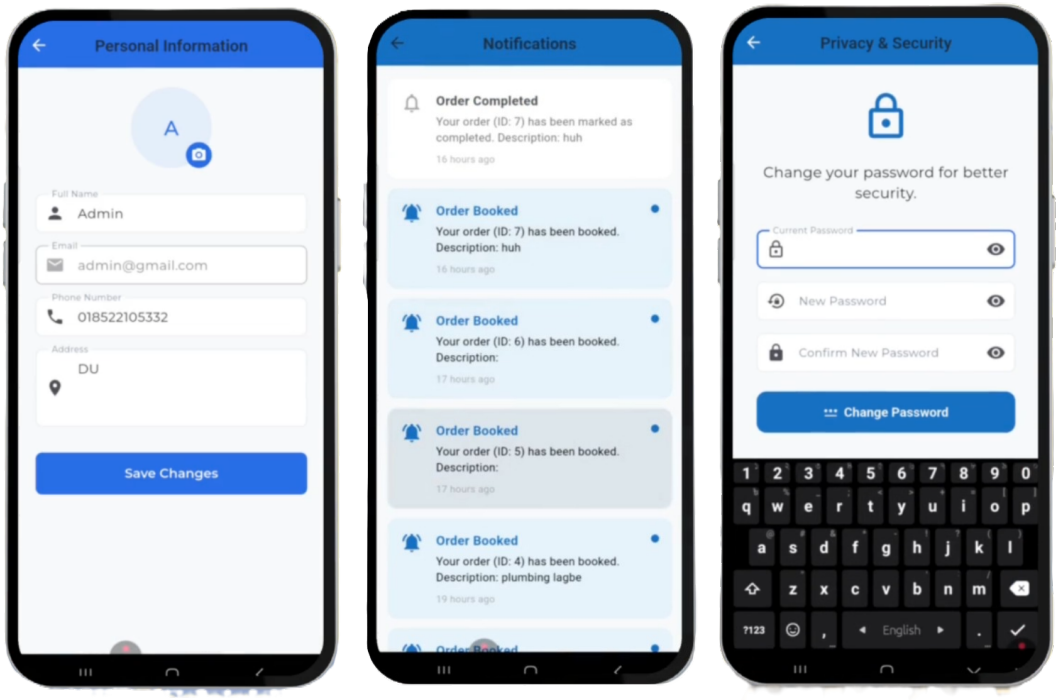

FIGURE 10 – Profile Update, Notification, update Password

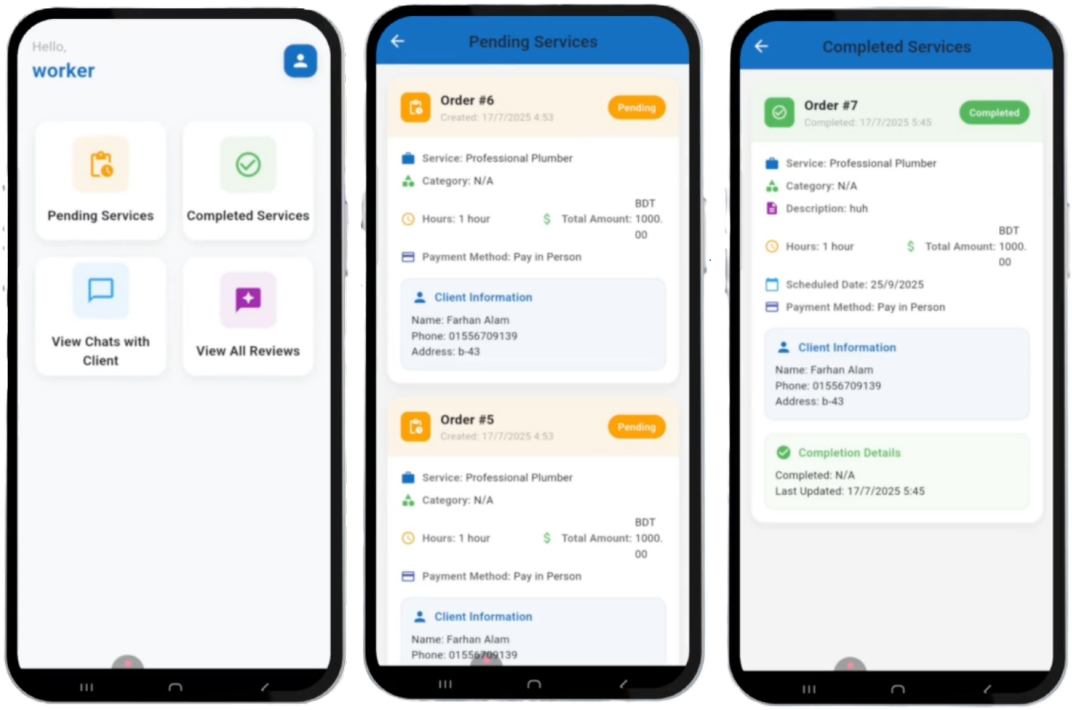

FIGURE 11 – Workers UI

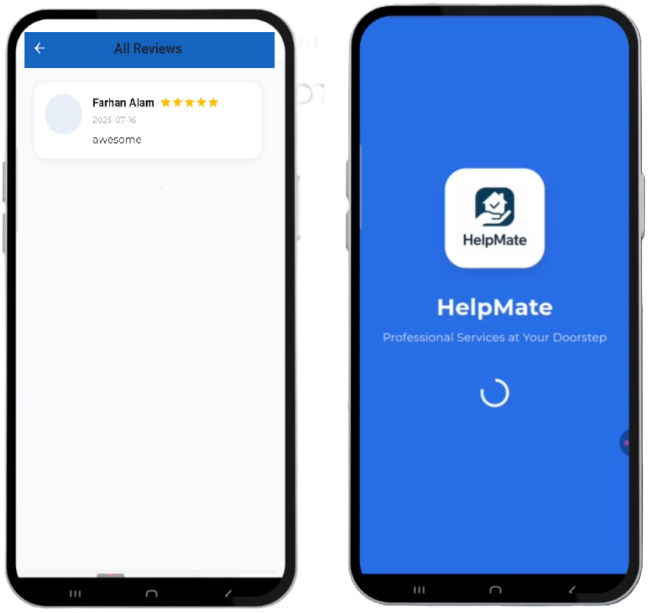

FIGURE 12 – Review of Worker and HelpMate Logo

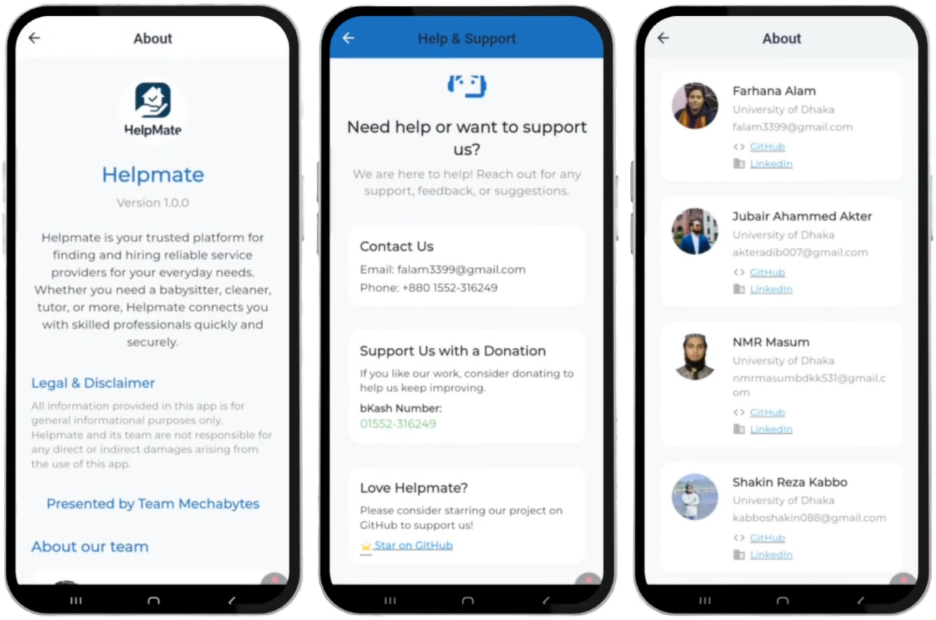

FIGURE 13 – About us

Note : The rest of the UI screens can be found at : UI Screens Repository

2.2 Database Schema

The backend database schema includes the following main tables :

- **Users :** Stores user and provider details (name, email, phone, role, etc.)
- **Categories :** List of service categories (e.g., Cleaning, Plumbing, etc.)
- **Workers :** Service provider profiles, skills, ratings, etc.
- **Orders :** Service booking and order management
- **Reviews :** User reviews and ratings for providers
- **Notifications :** Real-time notifications for users and providers

2.3 Code Repository Link

1 : The complete codebase and documentation are available at :
Click here for GitHub Repository
2 : The README includes a direct link to download the latest
Click here for APK

2.4 Video Demo Link

1 : A full video demonstration of the app is available at :
Click here for App Video Demo

3 Conclusion

3.1 Challenges and Solutions

- **Cross-platform UI Consistency :** Ensured a uniform look and feel across Android, iOS, and desktop using Flutter's responsive design features.
- **Authentication and Security :** Integrated secure JWT-based authentication and role management.
- **Real-time Updates :** Implemented real-time notifications and order status updates using FastAPI and WebSockets.
- **Database Integration :** Managed schema migrations and data consistency between SQLite (dev) and PostgreSQL (prod).
- **Team Collaboration :** Used GitHub Projects and Actions for effective team workflow and CI/CD.
- **Payment Integration :** Used SSLCOMMERZ for smooth payment system.

3.2 Lessons Learned

- Gained hands-on experience with Flutter for cross-platform development.
- Learned to design and implement RESTful APIs with FastAPI.
- Understood the importance of secure authentication and user management.
- Improved skills in team collaboration, version control, and project documentation.

3.3 Future Plan

The app currently includes all core features for a functional home service platform. In the future, we plan to :

- Enhance the UI/UX for even better usability
- Implement location-based service provider search and mapping
- Integrate AI/LLM for intelligent service matching and recommendations
- Add voice commands and natural language processing for booking
- Develop a more advanced admin dashboard with analytics
- Implement real-time chat support with AI assistance
- Add service scheduling optimization using machine learning

THE END