DECLARATION

This is to certify that the project entitled "**MessConnect**" has been carried out by Md. Azim Uddin in the Department of Computer Science and Engineering, Gopalganj Science and Technology University, Gopalganj-8100, Bangladesh. The above project or any part of this work has not been submitted anywhere for the award of any degree or diploma.

SUBMITTED BY

Md. Azim Uddin

ID: 21CSE031

MessConnect-App

Course Code: CSE278

Course Title: Project on Mobile Application Development

Submitted By:

Md. Azim Uddin

ID: 21CSE031

Submitted To:

Dr. Saleh Ahmed

Associate Professor

Department of Computer Science and Engineering,

Gopalganj Science and Technology University, Gopalganj, Bangladesh.

Gopalganj Science and Technology University, Gopalganj,

Bangladesh (GSTU).

16th July, 2025

Thesis Approval

This android app project is submitted to the Department of Computer Science and Engineering, GSTU, for the fulfillment of the requirements for the Degree of B.Sc. in Computer Science and Engineering.

Supervisor's Approval

_____

Dr. Saleh Ahmed

Associate Professor, Department of CSE,GSTU

Departmental Approval

_____

Associate Professor, Department of CSE, GSTU

Gopalganj Science and Technology University, Gopalganj (GSTU)

-

Instroduction: In urban and campus settings, students and young professionals often struggle to find mess seats or exchange used goods. Traditional methods such as relying on Facebook groups, physical notice boards, or personal networks are inefficient, fragmented, and slow.

MessConnect addresses this gap through a centralized digital solution — a mobile application that allows users to post and find available mess seats and second-hand items such as books, mobiles, and accessories. The system integrates interest tracking and real-time notifications to ensure immediate awareness when relevant posts appear. This ensures users never miss out on vital opportunities and allows mess owners and sellers to reach the right audience quickly.

The app simplifies and digitizes a process that has long been disorganized, time-consuming, and inefficient. Its benefits are especially prominent in university zones, PG accommodations, and urban settlements.

Problem Statement:

Students and working individuals often face problems when searching for mess accommodations or necessary used items like books and devices. Without a central platform, opportunities are often missed due to late communication. There's no way to automatically notify users when something they need becomes available.

The existing manual systems — asking friends, checking notice boards, or scrolling through social media — are not scalable or consistent. This problem becomes more critical during admission seasons or session changes when the demand is high.

MessConnect addresses this issue by offering:

- A central hub for posting and browsing offers
- Real-time interest tracking
- Instant push notifications to interested users

Goals & Objectives:

The major goals of this project are:

- To develop a user-friendly Android mobile application using Flutter
- To use Firebase Firestore to store user-generated ads and preferences
- To implement Firebase Cloud Messaging (FCM) v1 API for push notifications
- To allow ad posting with image uploads
- To make the app scalable and intuitive
- To introduce a system where users can mark their interests and get alerts accordingly

These objectives are achieved by focusing on usability, scalability, and real-time data communication.

Scope of the Project:

MessConnect focuses on simplifying peer-to-peer exchange for specific user needs:

- **Target Audience:** University students, mess owners, and urban renters
- **Location:** Focus on university-centric cities or neighborhoods
- **Supported Categories:** Mess Seats, Books, Mobiles, Keyboards
- **Scalability:** Can be expanded to include other items or services (e.g., bicycles, furniture)

The app is designed with modular features so that additional functionalities (chat, maps, user login) can be easily integrated in future versions.

Tools & Technologies Used:

- **Frontend:** Flutter (UI with Dart)
- **Backend:** Firebase Firestore (NoSQL database)
- **Push Notifications:** Firebase Cloud Messaging (v1 HTTP API)
- **Image Upload:** image_picker plugin
- **Networking:** http package, googleapis_auth for secure access
- **IDE:** Android Studio

- **Version Control:** GitHub

System Design:

App Architecture:

- UI Layer: Built using Flutter's widget system and Material UI
- Data Layer: Firebase Firestore handles CRUD operations
- Messaging Layer: FCM for real-time notification delivery

Database Design:

- **ads/** Collection: Each document stores an ad with fields like `title`, `description`, `price`, `type`, `created_at`
- **interests/** Collection: Stores user preferences (`wantsBooks`, `wantsMobile`, etc.) along with FCM token

User Flow:

- Login/Register (in future versions)
- View ads in feed by category
- Select interests from Interest Page
- Get real-time notifications for matching ads
- Post ads with image and description

Functional Requirements:

- Users must be able to select multiple interests
- Users can submit ads with title, price, type, description, and image
- Notification should only go to interested users
- All ads should be visible in reverse chronological order
- Users can navigate between pages easily using the bottom navigation bar

---

Non-Functional Requirements:

- App must be lightweight and fast-loading
- Push notification delivery should be reliable
- Firebase rules should ensure that only allowed users can write to the database
- App must handle device permissions for camera and storage gracefully
- The UI must follow basic accessibility standards

Implementation Details:

User Interface:

- HomeScreen contains bottom navigation bar (Ads, Interest, Post)
- Ad posting form is implemented using Flutter's `Form` widget
- Interest page uses `CheckboxListTile` to let users select options

Posting Ads:

- Ads are submitted via a form with validation
- image_picker is used to upload an image from the gallery or camera
- Data is pushed to Firestore with a timestamp

Subscribing to Interests:

- When a user selects an interest, it's saved in Firestore along with their FCM token
- This enables targeted notification delivery

Notification System:

- On ad posting, the backend function checks which users have that interest
- Notifications are sent using a POST request to FCM endpoint
- Service account is used for secure authentication

Testing & Validation:

Test Case	Result
Posting ad with image	✅ Successful
Selecting multiple interests	✅ Saved in Firestore
FCM token generation	✅ Works on real device
Receiving notifications	✅ Triggered upon ad post
Invalid form submission	❌ Rejected with error message
Offline usage	⚠️ Limited functionality

UI Screenshots:

Include the following screenshots with labels:

1. **Login Page**
2. **Home Page Navigation**
3. **Interest Page UI**
4. **Post Ad Form**
5. **Ad List with Price**
6. **Notification Screenshot**
7. **Firebase Console: ads/**
8. **Firebase Console: interests/**

Challenges Faced & Solutions:

Challenge	Solution
Real device not detected	Enabled USB Debugging, installed proper drivers
Camera not working	Added required permissions & FileProvider
Firestore denied access	Modified Firestore security rules
Notification not delivering	Used FCM v1 API with service account
Token mismatch	Regenerated FCM token and verified using logs

Limitations of Current Version:

- No login/authentication system
- No edit/delete functionality for ads

- Cannot filter by location
- Ads stay forever unless deleted manually
- UI not optimized for iOS

Future Enhancements:

- Add Firebase Authentication (Google, Email login)
- Include map support to display mess seat location
- Add real-time chat between buyer and seller
- Implement ad expiry system (auto-delete after 7 days)
- Admin dashboard to detect spam or abuse
- Add category-specific filters (e.g., price range for mobiles)

Conclusion:

MessConnect demonstrates how a targeted mobile app can solve a localized but common problem effectively. It provides a digital solution to a real-world need among students. Using modern tools like Flutter and Firebase, the project delivers a lightweight, scalable, and useful system for mess and second-hand item management. Its modular nature allows for future upgrades and integrations, making it a promising foundation for a fully developed marketplace app.

References:

- Flutter Documentation: https://docs.flutter.dev/
- Firebase Docs: https://firebase.google.com/docs
- Firebase Messaging: https://pub.dev/packages/firebase_messaging
- Image Picker Plugin: https://pub.dev/packages/image_picker
- Google Cloud Auth: https://pub.dev/packages/googleapis_auth

---

Appendix:

- Sample Firestore Rules
- Sample JSON Payload for Notification

- Screenshot of Firebase Console Ads
- Screenshot of Interests Collection
- AndroidManifest.xml configuration
- file_paths.xml for camera permission
