UNIVERSITY OF LEICESTER

School of Computing and Mathematical Sciences

CO7201 Individual Project

Interim Report

A Mobile App for Campus Navigation

Prerith Dsouza
pd245@student.le.ac.uk
249006686

Project Supervisor: Dr Kehinde Aruleba
Principal Marker: Anthony Conway

Word Count: 1208
Submission Date – 25/07/2025

DECLARATION

All sentences or passages quoted in this report, or computer code of any form whatsoever used and/or submitted at any stages, which are taken from other people's work have been specifically acknowledged by clear citation of the source, specifying author, work, date, and page(s). Any part of my own written work, or software coding, which is substantially based upon other people's work, is duly accompanied by clear citation of the source, specifying author, work, date, and page(s). I understand that failure to do these amounts to plagiarism and will be considered grounds for failure in this module and the degree examination as a whole.

Name: Prerith Dsouza
Date: 25/07/2025

Contents

1. Overview

The project is currently at 65–70% completion. The development has followed the structure outlined in the preliminary report. The user interface has been designed and implemented using Flutter, including core screens such as login, registration, home, map, and profile screen. Weather integration, voice search, Firebase authentication, and location-based redirection features have been partially implemented. This report details the tasks completed, tasks in progress, and tasks yet to be started, along with a revised timeline for project completion and submission.

2. Progress

2.1 Completed Tasks

- App development environment setup: The project environment was successfully configured with Flutter, Android Studio, Firebase, and GitLab. Testing has been conducted on physical Android device to ensure compatibility. This setup enables continuous integration and easier debugging throughout development.

- User authentication and session Flow: Firebase Authentication has been fully integrated, allowing secure login, registration, and logout. Session handling ensures users remain authenticated across app restarts, which forms the backbone for personalized features.

- Home screen UI: The home screen now features a dynamic greeting, the official university logo, and quick access buttons for key functions (Navigate, Timetable, Buses). This serves as the central hub of the application.

- App navigation logic between screens: Navigation flows have been implemented between all major sections, including Home to Timetable, Home to Notifications, Home to Profile, and Home to Map. This ensures a seamless user journey.

- Weather integration: Real-time weather updates are now functional via OpenWeatherMap API combined with Geolocator for location-based data. It enhances the app's contextual relevance for students navigating campus.

- Timetable and notification screens: Dedicated screens have been designed and populated with static test data to match expected UI/UX patterns. These act as placeholders for dynamic content integration.

- UI responsiveness: The app's layout adapts to different screen sizes and orientations. Early testing shows consistent behaviour, though a few secondary buttons still require connection to backend logic.

2.2 Ongoing Tasks

- Voice search integration: Speech-to-text functionality has been implemented with working microphone permission handling. The current focus is refining command matching so recognized speech can trigger relevant navigation or search actions in real time.

- Navigation feature (map): The redirection logic between search queries and map view has been built. Ongoing work includes embedding a live map, rendering campus routes, and enabling location-based search.

- Role-based UI and settings: The Profile screen now supports toggles for dark mode and notifications and is partially integrated with dynamic user data. Next steps include role-specific UI variations and saving preferences to Firebase.

2.3 Pending Tasks

- Firebase firestore integration: The backend database setup is essential for storing saved routes, user preferences, and dynamic campus data. The schema design and integration are planned for coming weeks.

- Bus API integration: Real-time Centre Bus schedule integration remains pending. This will allow students to check live bus timings within the app.

- Offline mode: Work is planned to enable cached campus maps and saved searches for scenarios where internet access is limited. The feature is currently under feasibility review.

- Location based trigger notifications: Context-aware alerts for events, bus arrivals, or building proximity will be implemented via geofencing APIs and background location services.

- Admin dashboard development: A separate admin interface for managing maps, schedules, and user analytics is still to be developed. This will support maintenance and future scalability of the system.

3. Updates to Initial Plan

- Indoor navigation: Originally included but deferred due to time and resource constraints and the complexity of indoor positioning systems. The focus has shifted to optimizing outdoor campus navigation.

- Gamified onboarding: Removed to prioritize core navigation, weather, and personalization features. This ensures delivery of a robust MVP within deadlines.

- Voice command expansion: Initially meant for simple searches, the feature is being

extended to recognize specific room or facility names and redirect users accordingly, improving usability.

4. Challenges

- Gradle and SDK issues: Early development faced delays due to Gradle configuration errors and Android SDK compatibility mismatches. Transitioning to Java 11 and aligning Gradle/AGP versions resolved most issues.

- Speech to text integration: The speech to text plugin required android gradle plugin compatibility adjustments, including setting a namespace and ensuring the environment used java 11+. Other features worked smoothly with Java 11, integrating voice input demanded additional dependency and SDK configurations.

- Firebase Android synchronization: Initially there were sync issues between firebase and the android app due to outdated google services plugin versions. These were resolved by updating build files and SDK paths.

5. Time plan

5.1 Updated time plan

Week	Start Date	End Date	Tasks to be completed	Description
1-2	16th June	27th June	Explore similar Apps - Completed	Conducted research on existing navigation apps; identified gaps in personalized campus features
			Set-up app development environment - Completed	Flutter, Android Studio, Firebase, and GitLab configured, physical device testing set up
			Preliminary Report submission (27th June) - Completed	Report Submitted
3-4	30th June	13th July	Wireframes and UI mock-ups - Completed	Designed Splash Screen, Login and Registration Screen, Home Screen, Map Screen and Profile screen flows.
			Begin frontend development – Partially Completed	Implemented core UI with navigation and static screens

5-6	14th July	25th July	Weather integration - Completed	Integrated OpenWeatherMap and Geolocator
			Timetable & notification screens - Completed	Static versions built and tested.
			Voice search integration -Partially Completed	Permissions and speech-to-text tested, currently refining commands.
			Navigation feature - Partially Completed	Map redirection logic is ready, live embedding in progress.

5.2 Upcoming tasks time plan

Week	Start Date	End Date	Tasks to be completed	Description
7-8	28th July	10th August	Complete voice search integration	Finalize speech-to-text with search redirection for buildings.
			Embed live map and campus routes	Integrate Google Maps, search, and navigation routes.
			Profile and role-based UI	Link dynamic user data, preferences, and admin-specific features.
9-10	11th August	25th August	Firebase Firestore integration	Dynamic saved routes, settings, and location data storage.
			Bus API integration	Real-time Centre Bus schedule API.
			Offline mode	Cached maps and saved searches for offline usage.
			Location-trigger notifications	Geofencing and background services for contextual alerts.
			Admin dashboard	Management interface for routes, schedules, and analytics.
11-12	26th August	5th September	Final testing and debugging	QA, bug fixes, and polishing UI/UX.
			Usability testing and feedback	User trials and final refinements.
			Final submission (6th September)	Submit final app and report.