CS5520 Final Project Proposal

Student: Sukhrobbek Ilyosbekov

Introduction

I would like to develop a mobile application for my logistics project that demonstrates advanced Kotlin development skills through Compose Multiplatform (CMP), real-time location tracking, and interactive data visualization. This application serves as a comprehensive driver companion app for freight logistics operations, enabling truck drivers to manage their assigned loads, track delivery progress, monitor earnings, and receive real-time notifications about load assignments and updates. The project leverages modern Android development practices including Kotlin Multiplatform Mobile (KMP) for shared business logic and UI, GPS tracking with proximity detection, and Firebase Cloud Messaging for push notifications. The application integrates with a backend API to synchronize load data, driver statistics, and location updates in real-time, while maintaining a clean architecture with MVVM pattern.

Why does this project interest me?

This project interests me because I have been developing a comprehensive logistics management system for trucking companies for a long time ago, with the backend API and web frontend applications already fully implemented. However, the driver mobile application was originally written in MAUI and was incomplete. I decided to rewrite the driver app from scratch using Kotlin Multiplatform to leverage native Android UI capabilities through Jetpack Compose, improve code maintainability, and gain hands-on experience with advanced Kotlin development practices.

What do I hope to learn?

- Kotlin Multiplatform Development: Learn the expect/actual pattern for platform-specific implementations, understand module structure in KMP projects, and learn effective strategies for sharing business logic while maintaining platform-specific features.
- Data Visualization: Integrate charting libraries for displaying financial and distance metrics, implement date range filtering, format data appropriately for different time periods, and create interactive charts.
- Location Services & Background Processing: Implement foreground services for continuous location tracking, handle location permissions properly, calculate

distance and proximity detection, and manage battery-efficient background operations.

What questions do I want to answer?

- How can I effectively structure a Kotlin Multiplatform project to maximize code sharing while keeping platform-specific code maintainable?
- How do I properly handle authentication flows with JWT tokens, including token refresh, tenant isolation, and secure storage in a mobile application?

Objectives

Primary

- Rewrite MAUI app to Kotlin
- Using Material3 components instead of Syncfusion from MAUI app
- Login screen
- Authentication: Login with backend
- Update load status
- User data screen
- Build APIs and connect with the backend

Secondary

- Driver statistics with charts
- Push notification
- Real-time location tracking and reporting to backend
- Add google map to the main screen
- Shared Compose UI instead of Android only

Tertiary

- Support for trips API
- Biometric authentication
- Direct login without redirecting to Identity website
- iOS implementation
- Upload load invoice after delivering
- Native Mapbox map layer instead of Google Map's HTML iframe
- Chat with dispatcher within app
- Support for multiple languages

- Settings Screen

Resource Statement

Required Resources

- Android Studio
- Physical Android device for testing (I will use my personal phone)

Partnership Statement

I will work on this project alone without any partners.

Intellectual Property Statement

I will develop this project as a portfolio item and a learning exercise. Therefore, I do not require any special licenses. I will create it as an open repository on GitHub under the standard MIT license.