System Design Document

Navi

Multi-Platform Travel Planner

An EECS3311 F25 Project by Group AK:

Klodiana Kamberi
Tjioe Andrew Elvio Febrian
Akash Deep
Trung Hieu Nguyen
Trong Duc Nguyen

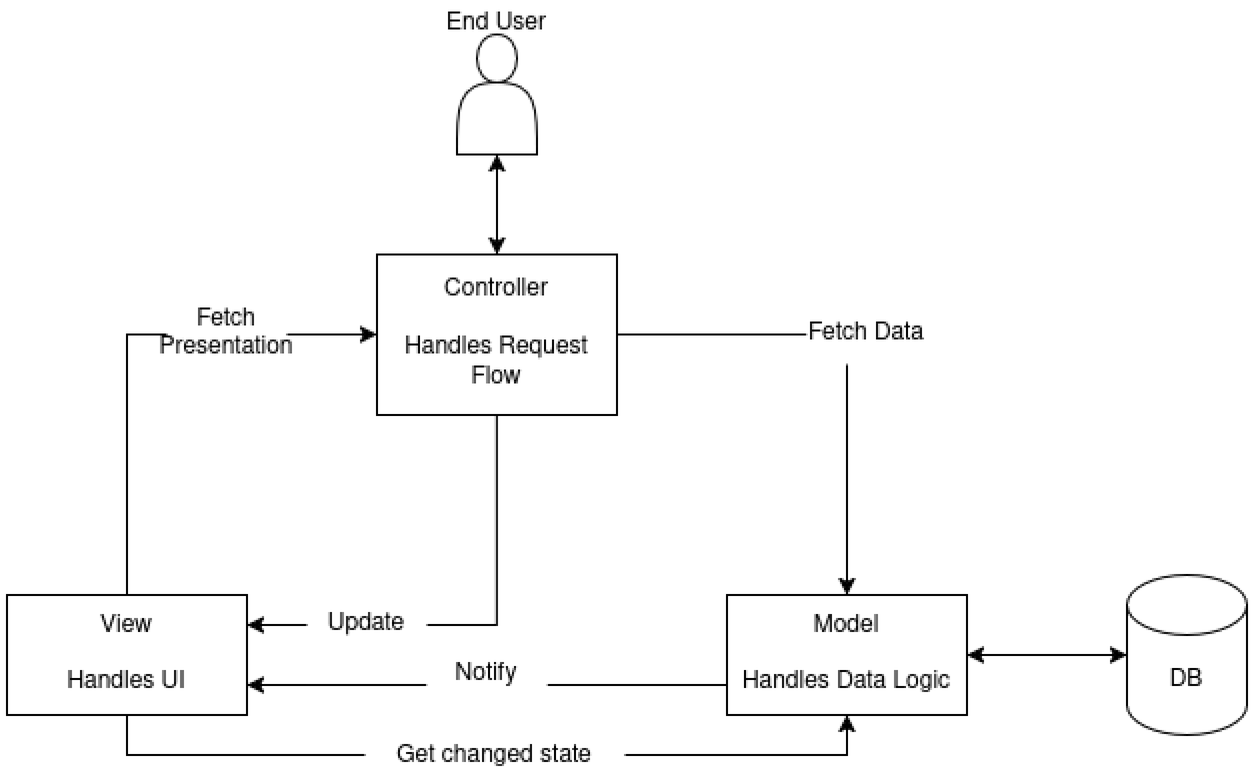
Table of Contents

Overview

Navi is a travel planning application designed to make trip organization collaborative and enjoyable. With Navi, users can seamlessly coordinate their travel experiences by managing events within trips, organizing schedules through its itinerary features, and visualizing event locations interactively on a map. The intuitive interface ensures users can easily add, modify, and review trip details, making group planning both efficient and engaging.

Navi is built with a robust technical foundation focused on delivering performance and flexibility across multiple platforms. Our development leverages Kotlin Multiplatform paired with Compose Multiplatform to simultaneously target iOS, Android, web and JVM environments. The application is backed by a Ktor-based server and a PostgreSQL database for reliable storage and API services. Additionally, a separate Python server is integrated to deliver advanced AI-driven features, ensuring a comprehensive travel planning solution.

Software Architecture Diagram



CRC Cards

Class Name: **User**	
Parent Class: - Subclasses: -	
Responsibilities: - Knows name - Knows userID - Knows email - Knows list of trips - Makes trips - Edits trips - Manage account info	Collaborators: ● **Trips**

Class Name: **Trip**	
Parent Class: - Subclasses: -	
Responsibilities: - Knows trip title - Knows list of events - Knows list of users - Knows trip duration - Knows header image URL - Make and edit events	Collaborators: - Events - Duration

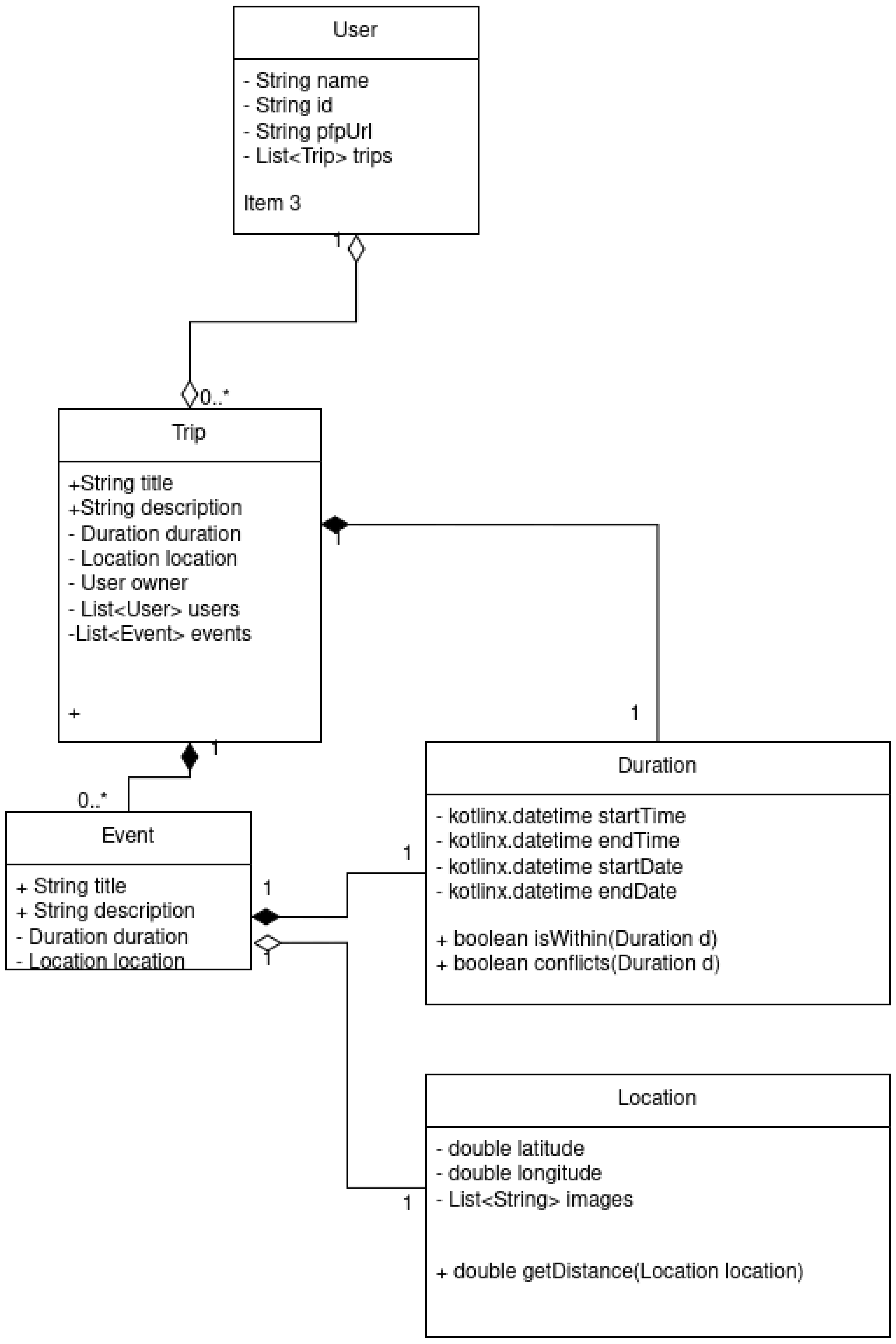
Class Name: **Event**	
Parent Class: - Subclasses: -	
Responsibilities: - Knows event title - Knows duration - Knows event description - Knows event location	Collaborators: - Location - Duration

Class Name: **Duration**
Parent Class: Subclasses:

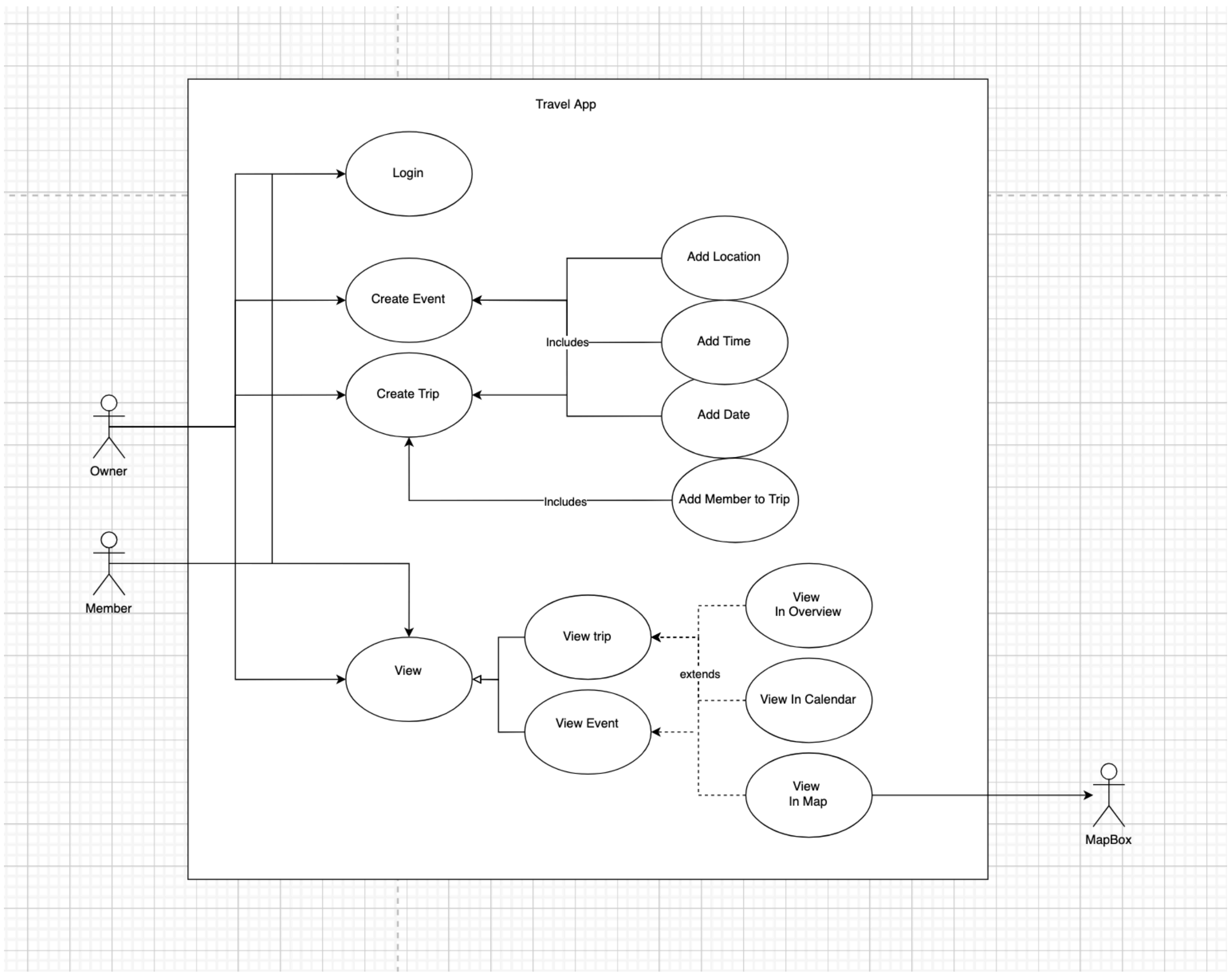
Vars: - localDate startDate - localTime startTime - localDate endDate - localTime endTime	
Responsibilities: - Knows start and end time and date - Provides conflict check between durations	Collaborators:

Class Name: **Location**	
Parent Class: - Subclasses: -	
Responsibilities: - Knows title - Knows GPS coordinate - Knows image of location if any	Collaborators:

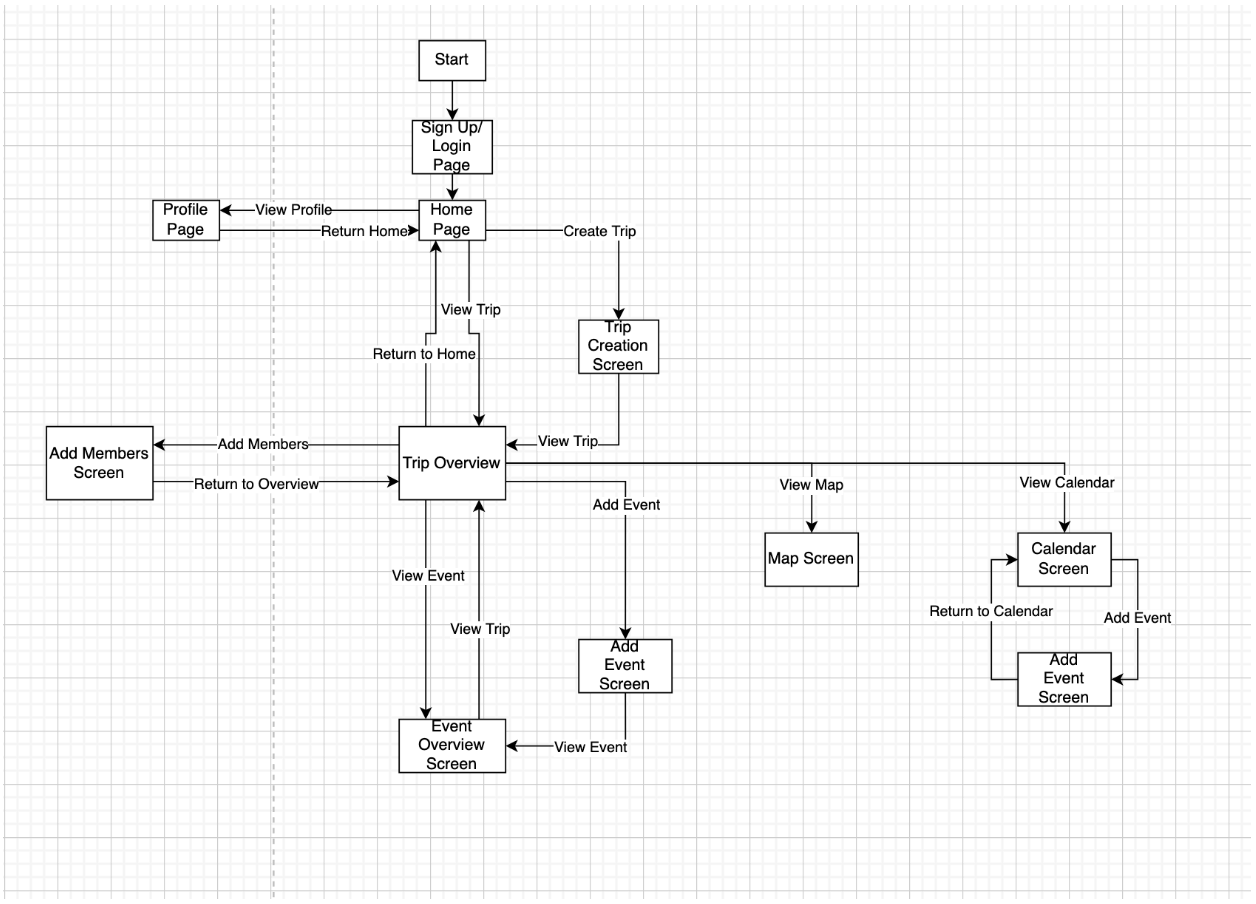
Class Diagram



Use-Case Diagram



User-Flow


Start
Sign Up/ Login Page
Home Page
View Profile
Profile Page
Return Home
Create Trip
View Trip
Return to Home
Trip Creation Screen
View Trip
Add Members
Add Members Screen
Return to Overview
Trip Overview
Add Event
View Map
View Calendar
Map Screen
Calendar Screen
Return to Calendar
Add Event
Add Event Screen
View Event
View Trip
Add Event Screen
Event Overview Screen
View Event