

**Faculty of Engineering & Applied Science**

**SOFE 4640U Mobile Application Development**

**Mobile Calendar Application**

**Deadline date: October 27, 2022**

**Group Number*:* 12**

**Course Instructor: *Anwar Abdalbari***

| **Student Name** | **Student Id** |
| --- | --- |
| Preet Patel | *100708239* |
| [Tiwaloluwa Ojo](mailto:tiwaloluwa.ojo@ontariotechu.net) | *100700622* |
| [Aaditya Rajput](mailto:aaditya.rajput@ontariotechu.net) | *100622434* |

## 

## Problem Description and Statement

Maintaining a schedule is a difficult task. Gone are the days of composing and managing multiple events on a singular paper copy of a calendar. They do not notify you of your event nor are they easy to view. Poor user interface often results in missed events/deadlines. Our application will be focusing on designing a mobile calendar application for managing and maintaining personal schedules. It will provide a user-friendly interface that can be tailored to personal preferences.

## Requirements

* View Events in calendar view - Monthly, Weekly, Daily
* View Events in a list view
* Filter Events in all views
* Create, Delete, and Edit Events - Type, Name, Description, Date, Time
* Send reminders to user prior to event
* Sync google calendar events with the application
* Search events by name or date
* Create, Delete, Edit notes

## Project Stack

* Flutter- Cross-platform (Android and IOS) application development
* Google Calendar API - Events synchronizations with the Google Calendar API
* SQLite- Events store for locally created events on our mobile calendar app
* Shared Preferences - Saving user themes, and app personalization settings

## Project will Include

* User Interface and Multiple Activities - Events Calendar and List view, Events Filter, Event Search, Log in, Event creation, Event views, Event filter, User Settings, etc.
* Intents - Switching between activities
* Save state mechanism - Shared Preferences and Local Database
* The usage of internet resources (i.e. JSON) - Google Calendar API
* Local database transactions - SQLite