

FUNCTIONAL SPECIFICATION DOCUMENT

“TMANAGE”

INFM 600, TEAM - 4

*Prof. Ping Wang
Pranav Adiraju
Steicy Singh
Ushasri Bhogaraju
Sravya
Animesh Nandan
Amit Broca
Ritika*

Table of Contents

1. Introduction

- A. Purpose**
- B. Scope**
- C. Intended Stakeholders/Audience**
- D. Document Conventions**

2. System Overview

- E. Use-case scenarios**
- F. Use-case diagrams**
- G. System Organization (ERD/DFD)**
- H. User Class/Characteristics**

3. System Features

- I. Functional Requirements**
- J. User Interfaces**
- K. Hardware Interfaces**
- L. Software Interfaces**
- M. Communication Interfaces**

4. Design Overview

- N. Design Prototype**
- O. Design Constraints**
- P. Design Goals/Guidelines**

5. Non-Functional Requirements

- Q. Performance Requirements**
- R. Software Quality Attributes**

1. Introduction

A. Purpose :

To create a Mobile App “**TManage**” that provides a unified view of time commitments for a student, that may be distributed across different calendars and the ELMS system, to validate and accept time slot requests from students.

B. Scope:

The app will be available for students who are Android users for the time-being, and we could venture into multiple spaces in the future to support all available platforms.

C. Intended Stakeholders/Audience :

UMD students

D. Document Conventions

The document uses the following conventions:

ERD	Entity Relationship Diagram
DFD	Data Flow Diagram
DB	Databases
UI	User Interface

2. System Overview

E. Use-case Scenarios & Diagrams

- Scenario 1: Log in to TManage (Student)

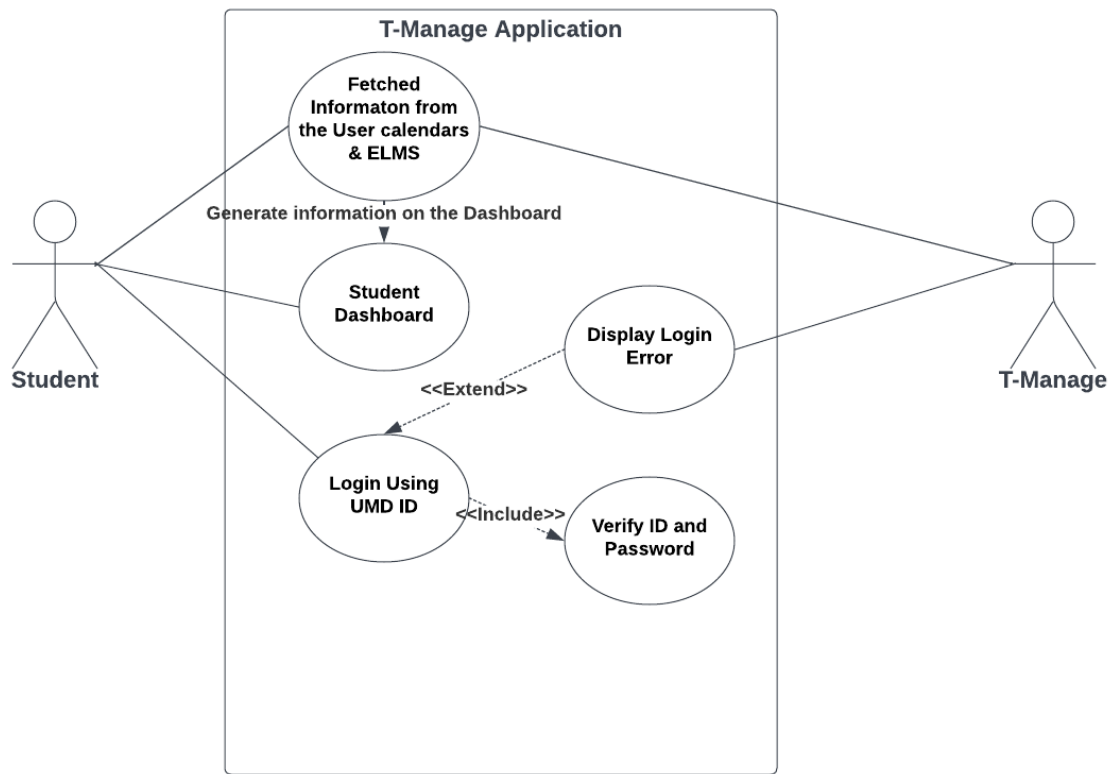
Use-case ID	U_ID_1
Name	Student_Name
User	UMD_Student
Pre-conditions	Use valid UMD email and credentials
Post-conditions	Successfully logged in to TManage App

Related conditions	Verify if credentials are right
Stakeholders	Student

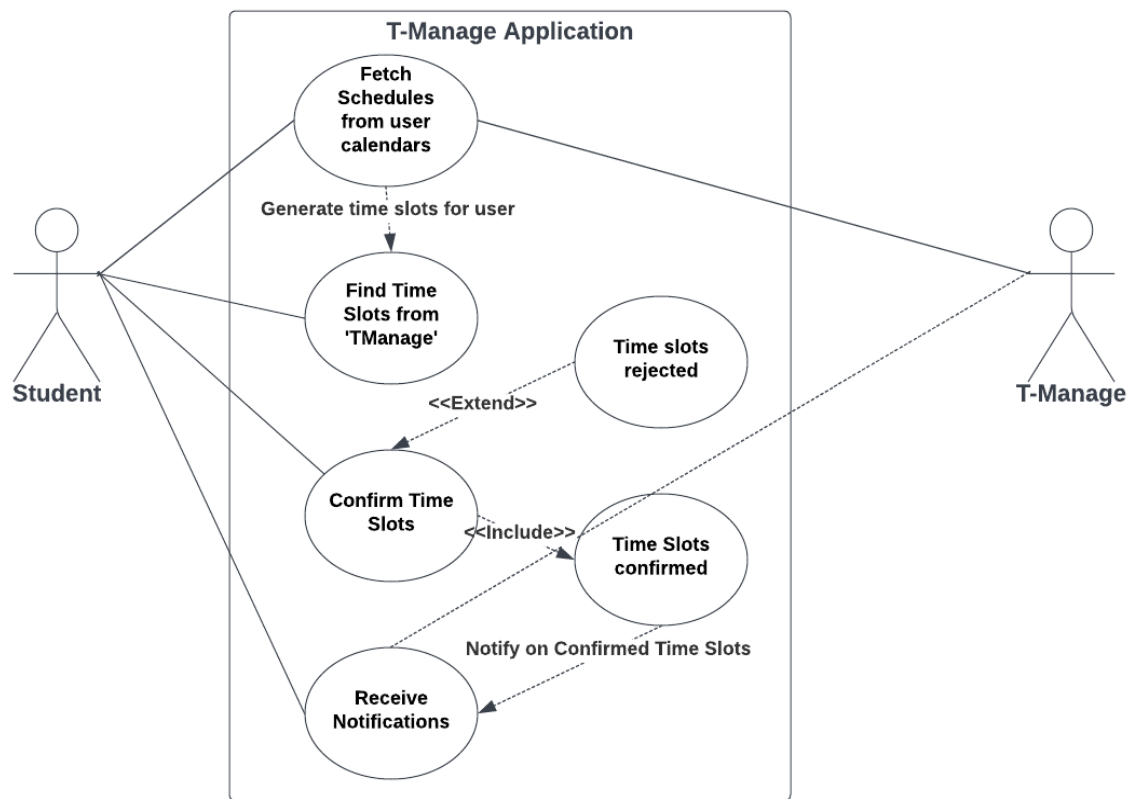
- **Scenario 2: Validate time slot availability based on data collected from student calendars**

Use-case ID	U_ID_2
Name	Student_Name
User	UMD_Student
Actor	TManage App
Pre-conditions	Check all data collected from Calendars and ELMS
Post_conditions	Accept Student input time slot request
Related conditions	Validate time slot availability and confirm to student
Stakeholders	Student

- **Scenario 1:
Log in to TManage (Student)**



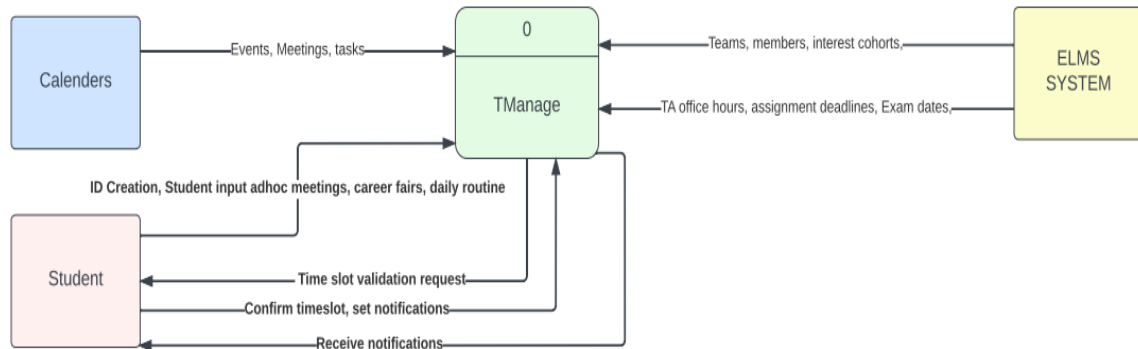
- **Scenario 2:**
Validate time slot availability based on data collected from student calendars



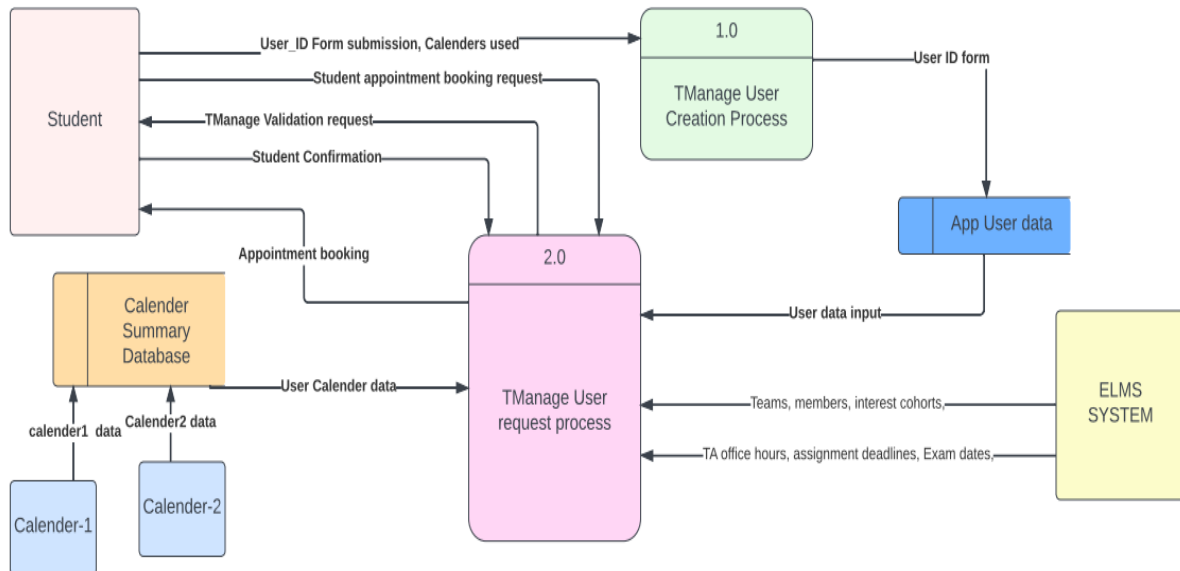
F. System Organization:

- **Data-Flow Diagram:**

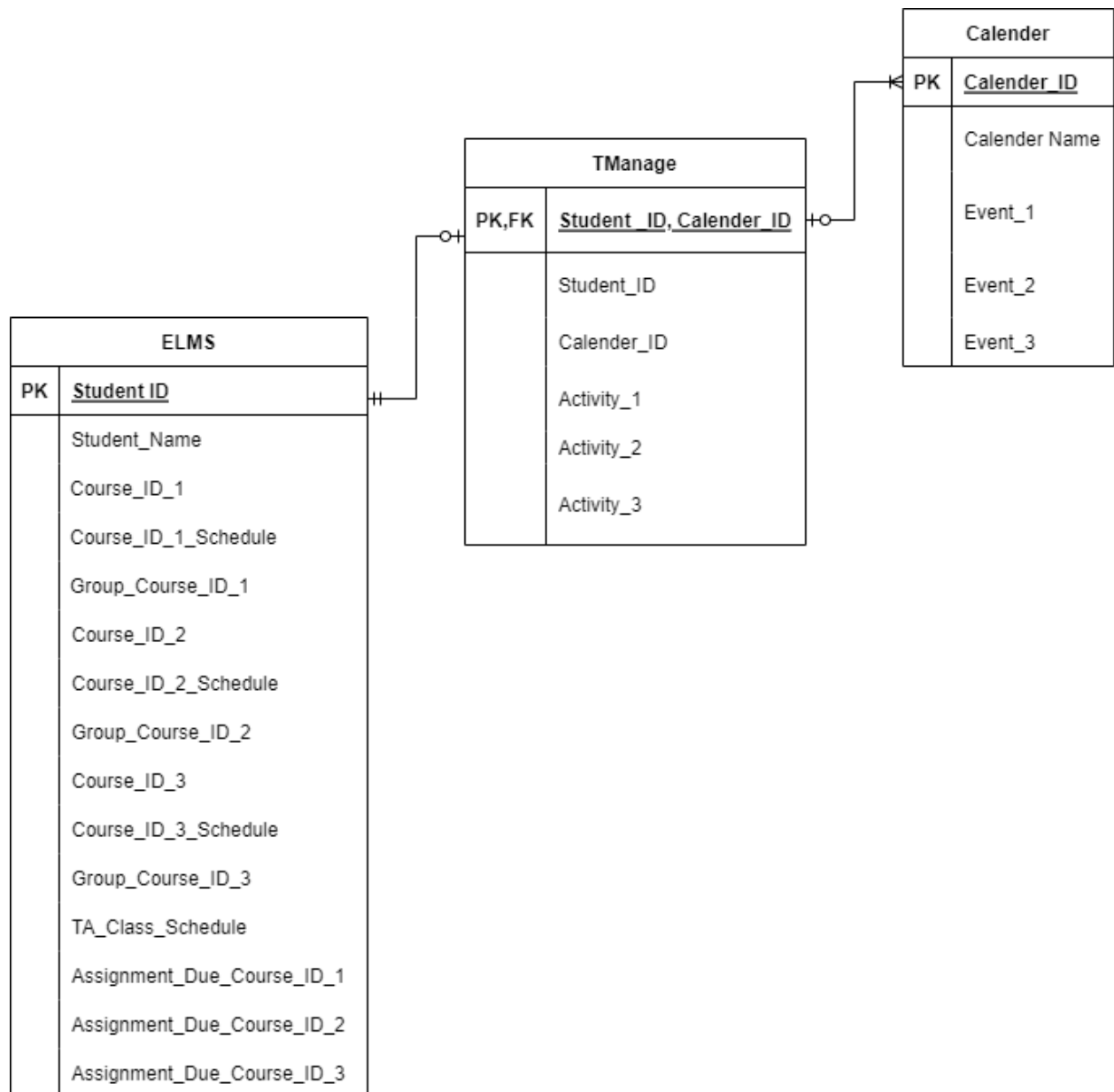
INFM 600 PROJECT PROPOSAL, TEAM-4
 CONTEXT DIAGRAM -
 TMANAGE APP- UNIFIED CALENDER-SCHEDULER



LEVEL "0" DIAGRAM TMANAGE - UNIFIED CALENDER-SCHEDULER



Entity-Relationship Diagram:



User/Class Characteristics:

The following functionalities of TManage are:

<Student> Functionalities:

- ☐ Create/Edit student profile
- ☐ View time commitments already made in Calender and ELMS
- ☐ Input required time for new event / activity
- ☐ Set notification.

<Admin> Functionalities:

- ☐ Execute periodical syncing of different calendars
- ☐ Execute syncing of ELMS database with APP
- ☐ Verify new request for time and validate against commitments
- ☐ Provide a timeslot

3. System Features

G. Functional Requirements:

Functional ID	Description
FR_ID_1	The system must allow the user to log in using UMD ID
FR_ID_2	The system must show all the prior time commitments for a student in different calendars
FR_ID_3	System must show ELMS Course IDs, Course times, Assignment due dates, Group No for each course, Group members names, TA commitments
FR_ID_4	When a fresh time slot is requested by student, system must be able to validate it against prior commitments

H. User Interfaces:

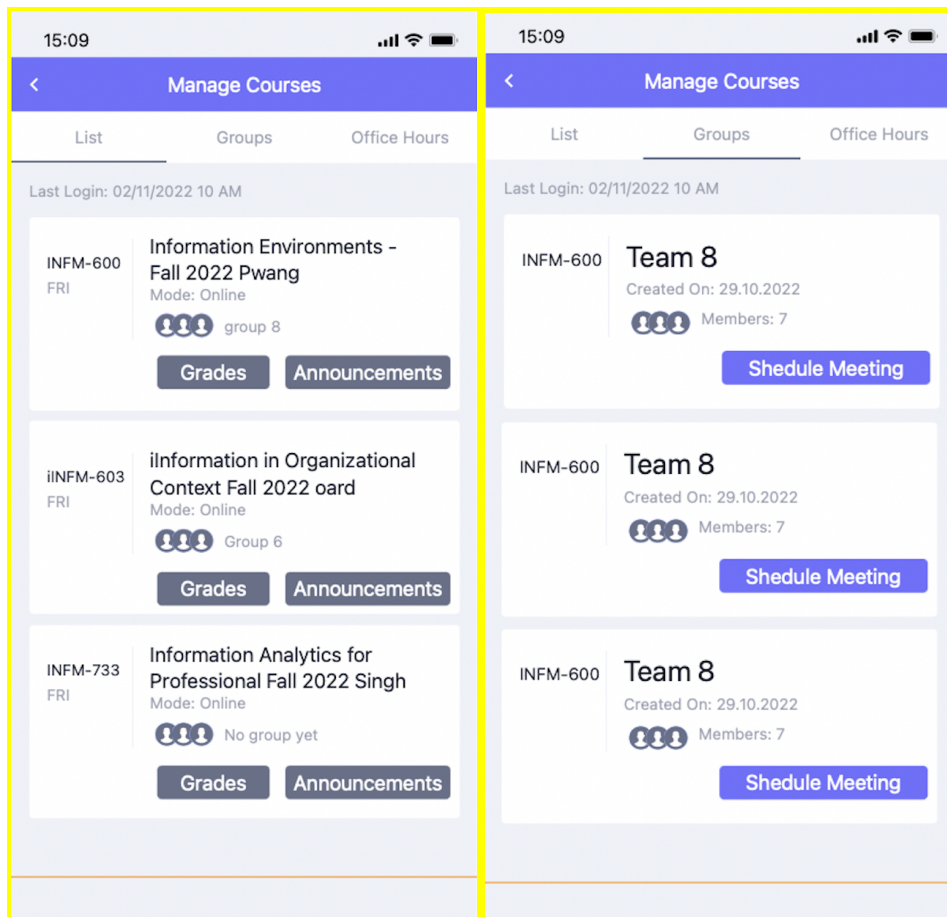
- **FRONT-END:** Java & XML
- **BACK-END:** No Back-end will be required for our prototype.

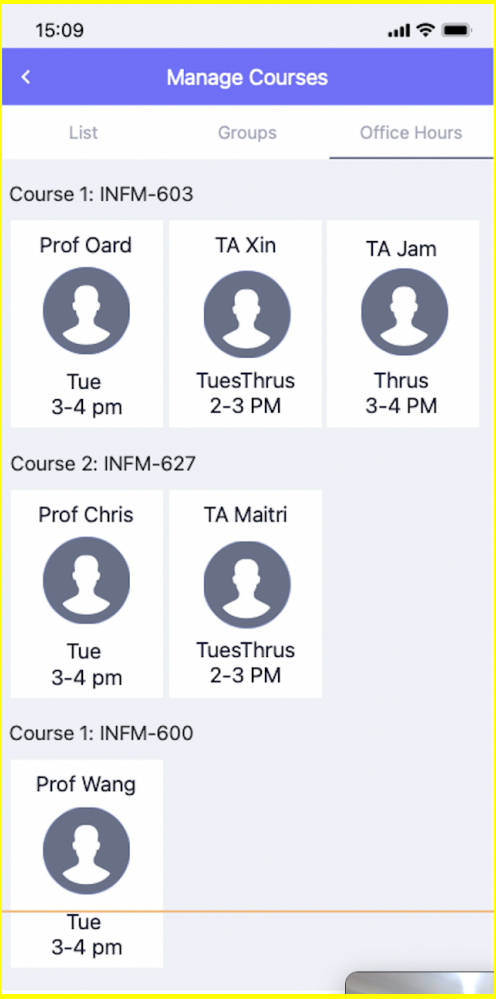
I. Software Interfaces:

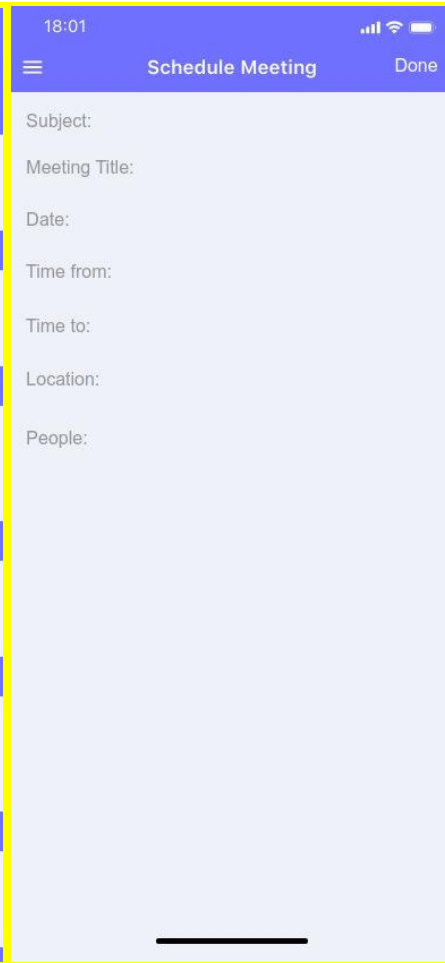
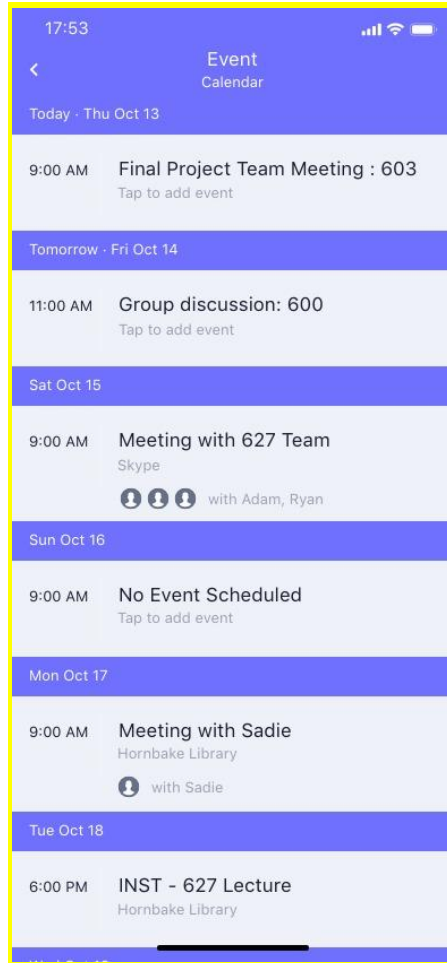
Software Used	Description
Operating Systems	Android Mobile Operating System
Databases	No Databases will be used for the Prototype

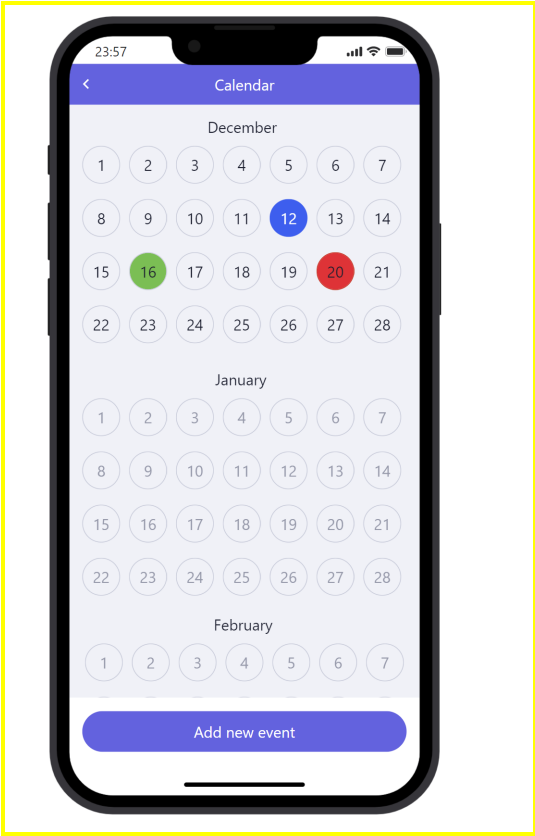
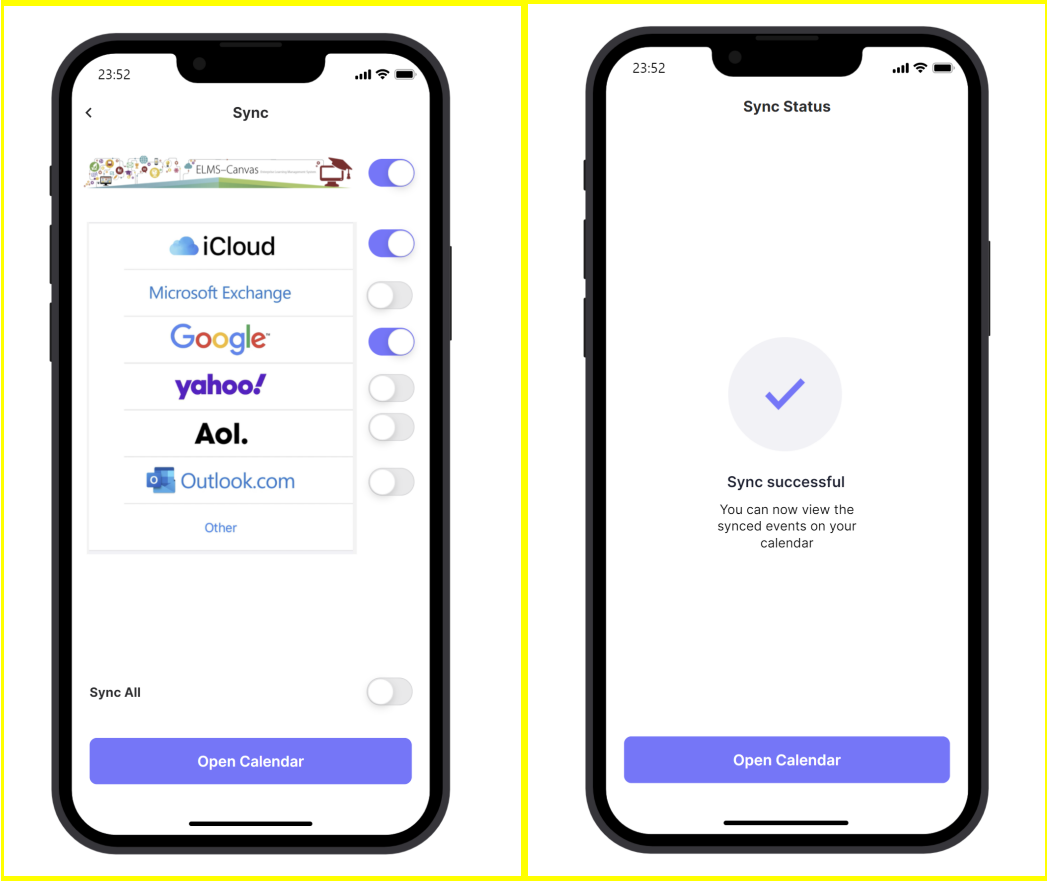
4. Design Overview

J. Prototype Designs :









K. Privacy Constraints:

For syncing time commitments of users, access permissions from various calendars and ELMS is to be obtained which may prove difficult in some cases

L. Design Goals/Guidelines:

- **Reliability:** The App will be tested with different student IDs to ensure that calendar data and ELMS data is being correctly fetched
- **Usability:** The usability will be extended to Apple users once it stabilizes for Android.
- **Portability:** Portable from Android version 12 and above

5. Non-Functional Requirements:

M. Performance Requirements:

- **Time Constraints:** Adherence to committed time schedules by all team members amidst other course requirements

N. Software Quality Attributes:

- **Correctness:** The system should not allow users to read the information that is not intended for that person.
- **Maintainability:** The system should use continuous integration in order to fix the bug-fixes without any downtime.
- **Safety:** Safety of student information to be implemented with adequate firewalls.
- **Privacy:** Student information should not be utilized for other purposes.

O. Adherence to timelines:

Name	Work entrusted	Timeline
Ritika	PPT, Coded Prototype	November 25, 2022

Sravya	A sample Coded Prototype	November 20, 2022
Steicy	Android UI Designs	November 5, 2022
Amit	Android UI Designs	November 5, 2022
Animesh	PPT, Android UI Designs	November 25, 2022
Pranav	PRD document, PPT	20th october, 2022
Usha	ER, DFD, PRD Document	20th october, 2022

S. Testing Overview

Testing Scope

Test Item	Explanation
Design	Test that the layout actually follows the design intended
Backend Components	Test that the backend components work in unison, fetching the required information
Accessibility	Test if the application is accessible to the target users
Complexity	Test if the application is simple enough for the users
Performance	Evaluate the performance of the application, making sure that it runs as intended under different circumstances

THE-END