Machine Problem 2

UI Composition and Flutter Widgets

CS 1635/2035 Fall 2025

Overview

Your goal for this assignment is to build a working interface using a wireframe as guidance. You are not required to do any back-end coding or data transfer for this assignment; you are only recreating the given wireframe. The purpose of this assignment is to give you practice implementing a visually complex UI in Flutter, and the deliverable will be your completed project in the form of a .zip file.

Requirements

THIS WILL BE A MOBILE APP, if you do not already have a mobile android or iOS simulator set up as a Flutter build target on your computer, you must do so for this project.

For this assignment, you will be implementing a basic productivity app, capable of receiving a variety of different messages (tasks, events, and memos). The wireframes for the app are provided at the end of this document, but do not worry about perfectly recreating them. The point of this assignment is the *composition* of the app; details such as font, color, and assets should follow the theming framework that you have chosen for your app (e.g., Android or iOS). If Android, you should use Material and for iOS you should use Cupertino themes.

IF YOU ARE TAKING THIS CLASS FOR GRADUATE CREDIT (CS 2035), YOU MUST IMPLEMENT YOUR INTERFACE FOR <u>BOTH</u> Android AND iOS.

Your finished interface must:

1. Contain sample emails that a supposed user of the app has received.
    - These sample message should contain relevant data indicating:
        - Who the message is from
        - The subject/description of the message
        - The time the message was received
        - The contents of the message
    - Each message should fall into one of three categories, which will determine the contents of the email:
        - Memos – contain a written message
        - Events – contain an invitation to a scheduled event, which has its own data (i.e. Title, Date and time, etc.)
        - Tasks – contain a task that the user has been assigned
2. Display a home/default "Inbox" screen that includes a list of the message and previews the information contained in each one.
    - The preview should display an icon indicating which category (memo, event, or task) the email falls into.

3. When a message from the list is selected, open the contents of the message in a separate detail section of the interface, where all the information contained in the message is displayed.
    - Memos should display the full message received
    - Events should display the details of the event, and give the user the option to accept or decline the invitation
    - Tasks should display the name of the task, and give the user the option to mark the task as complete or add it to their to-do list

4. When the device (simulator) is turned horizontal, rotate the display in the app as well.
    - Additionally, when the device is horizontal and an message is selected, rather than opening it's contents in a separate screen, it will be displayed on half of the screen, while the inbox remains on the other half.

You are expected to implement your code using a *clean* Model View View Model (MVVM) design paradigm.

You are expected to use the data represented in the linked JSON file. You do not need to maintain the data structure, but the data values should not be altered. Please refer to the Dart documentation on JSON deserialization to populate your models.

Deliverable

- Compress your project folder into a .zip file and submit it through canvas

Grading Rubric (150 pts total)

1. **Inbox Composition (25 pts)**

- (10 pts) Inbox displays list of sample message
- (10 pts) Previews of message contain all necessary data, including properly formatted dates and times (note, you should use date parsers to make this clean!)
- (5 pts) Message categories are indicated by icon in preview

2. **Message Composition (25 pts)**

- (10 pts) Opened message displays all associated data
- (5 pts x3) Opening an message displays different content depending on message category (memo, event, task)

3. **Functionality (50 pts)**

- (20 pts) Displays all the content from the JSON file in chronological order (depending

or ascending is implementors choice)

- (10 pts) User can open message contents by selecting a preview from the inbox and can navigate back to the inbox from opened message contents
- (10 pts) Rotating the device rotates the app display and adjusts content spacing to fit screen
- (10 pts) Message contents display changes based on device orientation (entire screen while vertical, half of screen while horizontal)

4. **MVVM Framework Adherence (30 pts)**

- (20 pts) Separation of Concerns - No business logic or data parsing in Widgets; Views are declarative only. ViewModels hold state/logic; Models are dumb data objects.
- (5 pts) ViewModels expose immutable UI state and explicit commands (e.g., openMessage(id), toggleFavorite()), not raw mutable models.
- (5 pts) JSON access isolated behind a service interface. ViewModels depend on abstractions;.

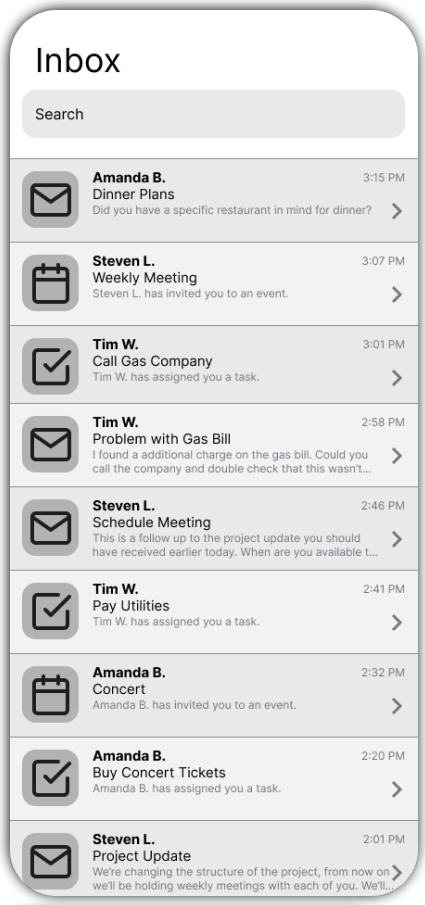
5. **Proper Display of Theming (20 pts)**

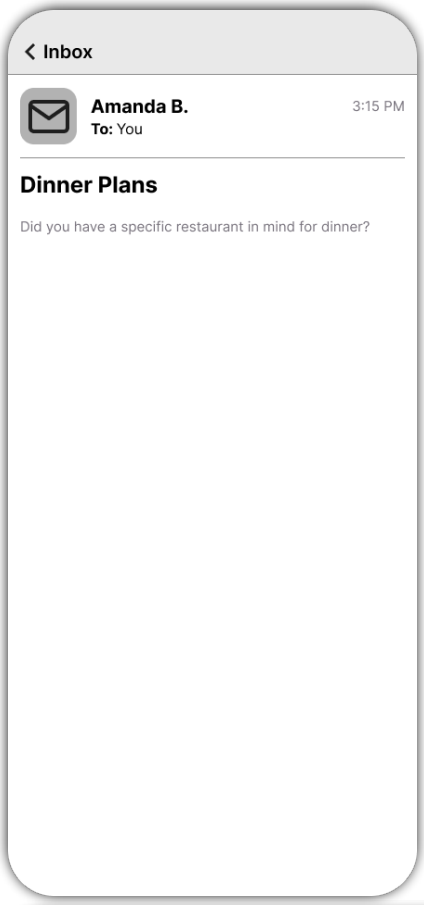
- (10 pts) App should follow all theming conventions and reflect the "UI feel" of a native app

6. **Structure and Legibility (20 pts)**

- (10 pts) App preserves structure of provided wireframe
- (10 pts) Components of app are well organized and legible

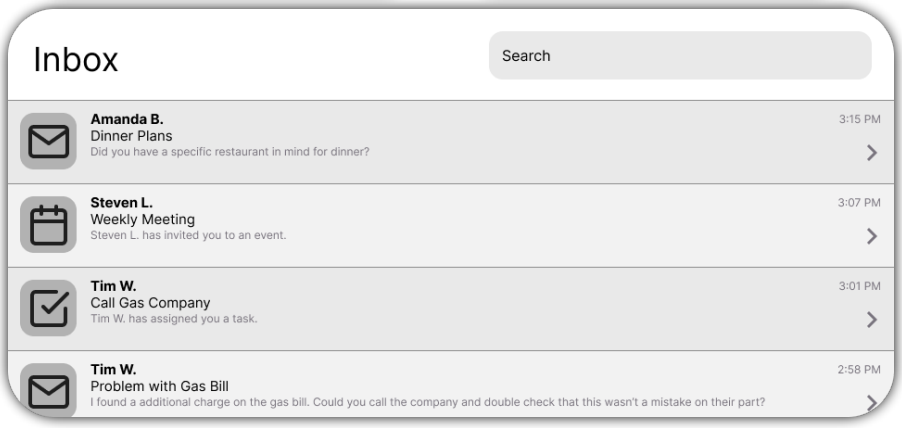
Wireframes

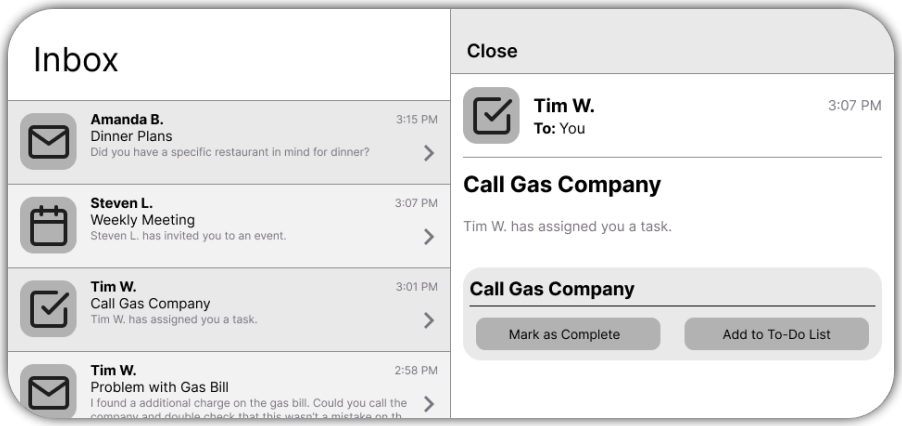

Inbox
Search
Amanda B.
Dinner Plans
Did you have a specific restaurant in mind for dinner?
3:15 PM
Steven L.
Weekly Meeting
Steven L. has invited you to an event.
3:07 PM
Tim W.
Call Gas Company
Tim W. has assigned you a task.
3:01 PM
Tim W.
Problem with Gas Bill
I found a additional charge on the gas bill. Could you call the company and double check that this wasn't...
2:58 PM
Steven L.
Schedule Meeting
This is a follow up to the project update you should have received earlier today. When are you available t...
2:46 PM
Tim W.
Pay Utilities
Tim W. has assigned you a task.
2:41 PM
Amanda B.
Concert
Amanda B. has invited you to an event.
2:32 PM
Amanda B.
Buy Concert Tickets
Amanda B. has assigned you a task.
2:20 PM
Steven L.
Project Update
We're changing the structure of the project, from now on we'll be holding weekly meetings with each of you. We'll...
2:01 PM


Inbox
Amanda B.
To: You
3:15 PM
Dinner Plans
Did you have a specific restaurant in mind for dinner?


Inbox
Steven L.
To: You
3:07 PM
Weekly Meeting
Steven L. has invited you to an event.
Weekly Meeting
Date: Tuesday, October 7th, 2025
Time: 2:15 PM
Repeats: Weekly
Accept
Decline


Inbox
Tim W.
To: You
3:07 PM
Call Gas Company
Tim W. has assigned you a task.
Call Gas Company
Mark as Complete
Add to To-Do List


Inbox
Search
Amanda B.
Dinner Plans
Did you have a specific restaurant in mind for dinner?
3:15 PM
Steven L.
Weekly Meeting
Steven L. has invited you to an event.
3:07 PM
Tim W.
Call Gas Company
Tim W. has assigned you a task.
3:01 PM
Tim W.
Problem with Gas Bill
I found a additional charge on the gas bill. Could you call the company and double check that this wasn't a mistake on their part?
2:58 PM


Inbox
Amanda B.
Dinner Plans
Did you have a specific restaurant in mind for dinner?
3:15 PM
Steven L.
Weekly Meeting
Steven L. has invited you to an event.
3:07 PM
Tim W.
Call Gas Company
Tim W. has assigned you a task.
3:01 PM
Tim W.
Problem with Gas Bill
2:58 PM
Close
Tim W.
To: You
3:07 PM
Call Gas Company
Tim W. has assigned you a task.
Call Gas Company
Mark as Complete
Add to To-Do List