Team name: Turtle and Rabbit Inc.
Names and PIDs: Chung Chan (cchan02), Angel Perez (angelpg), Pranesh Ambokar (pambokar), John Lynch(jwlynch4)

Part 1: Requirements Workshop (3%)
Preliminary project ideas: TODO Application

Here are some suggestions:
[Discussion Sign-Up 2023 - Google Sheets]
[Course/Project at main · CS3704-VT/Course (github.com)]

1. Provide an example of five hypothetical non-functional requirements for this system. Be sure to include the specific type of requirement discussed in class, with each requirement coming from a unique category.

- The user design will be intuitive
- The application should start up in less than five seconds
- Design of the app must include an online color scheme
- App should be easily accessible from all types of devices
- App should have a Light mode and Dark mode option

2. Provide an example of five hypothetical functional requirements for this system.

- Display the tasks
- Schedule the task
- Create a prioritize system, ex: Stars system (3 stars means most important, 0 means daily task, etc.)
- Remove the finished tasks
- Edit the task

3. Think of a specific task required to complete each of the functional requirements and non-functional requirements mentioned above. Estimate the amount of effort needed to complete this task using function points (i.e., using the values here). Briefly explain your answer.
Non-functional:

1. The user design will be intuitive
   a. Amount of time to take will be 35. It will take some time making the display concise but overall it should not be the most difficult task
2. The application should start up in less than five seconds
   a. Amount of time to take will be 75. Different devices have different capabilities so reviewing the code to make it run as smooth as possible and as efficiently as possible will be important.
3. Design of the app must include an online color scheme
   a. Amount of time to take will be 35. Taking advantage of online sources to create a color scheme which will be appealing to the user will mean looking up online sources and checking which color schemes are best for this kind of app.

4. App should be easily accessible from all types of devices
    a. Amount of time taken to complete this will be 100. You would have to coordinate through some kind of cloud or 3rd party database to be able to access data from multiple devices.
5. App should have a Light mode and Dark mode option
    a. Amount of time taken to complete this will be 55. Since the device will ideally be available on multiple platforms, this means designing a Light and Dark mode for each kind of device. iOS won't take long as Swift has a fairly simple way to install Dark mode on creator's app, however Android may take longer as their app-builders may take more time to design the UI/UX behind it.

Functional

- Display tasks -> 13
    - This is a must function, having the user read what task needs to be completed. The rating is 13 because the values needs to be displayed well.
- Schedule the task -> 40
    - It is one of the core functionalities of the app, so I believe it takes a lot more effort than others because you need to make sure this functionality works to avoid bugs.
- Create a prioritized system -> 100
    - It is one of the core functionalities of the app, so I believe it takes a lot more effort than others because you need to make sure this functionality works to avoid bugs.
- Remove the task -> 13
    - I think this functionality be done by using a similar code with "schedule", so wouldn't need that much effort.
- Edit the task -> 30

    The time to code the ability to edit tasks will be 30. This means changing the time scheduled, the task name, and the priority system. Change values shouldn't be complicated if the system is implemented well from the start

4. Write three user stories from the perspective of at least two different actors. Provide the acceptance criteria for these stories.

Actor 1

1. John Smith is a software engineer at Lockheed Martin. In order for the app to be successful for him, it will have to incorporate a reminder system to get his attention for projects that are approaching their deadline.
2. Jane Doe is a low-level systems programmer at Verizon who struggles with bad eyesight, which makes it hard for her to notice alerts. This means that for the app to be successful for her, it must incorporate an attention grabbing, but tasteful color scheme.

Actor 2

1. Jermaine Cole is a sophomore mechanical engineering student at Virginia. For this app to be successful for him, he asked that the app offers an option to give the user reminders early.

5. Provide two examples of risk that could potentially impact this project. Explain how you would mitigate these risks if you were implementing your project as a software system.

- **Risk 1**: Recently, unity made a new runtime fee that affects lots of game companies. If some popular coding editor like VSCode does a similar thing, it will impact the development.
    - Be familiar with other coding editors so you can be prepared for
- **Risk 2:** There is a risk that we run out of money from designing UI/UX for both iOS and Android. We could mitigate this risk by maybe using a language like Flutter, which claims to develop apps in both languages at once.

6. Describe which process your team would use for requirements elicitation from clients or customers, and explain why.

Strawman/Prototype will be good in this case since a prototype shows the steps and UI interface of our design. TODO applications can be prototypes to look like mobile apps and websites. There are software to implement prototype such as a TODO application. The customer will have the ability to see our prototype. This means if any mistakes are made during our prototype, we as designers can revise and fix our problems. This will follow a iterative process since prototypes require revision.

Part 2: Requirements Analysis (5%)

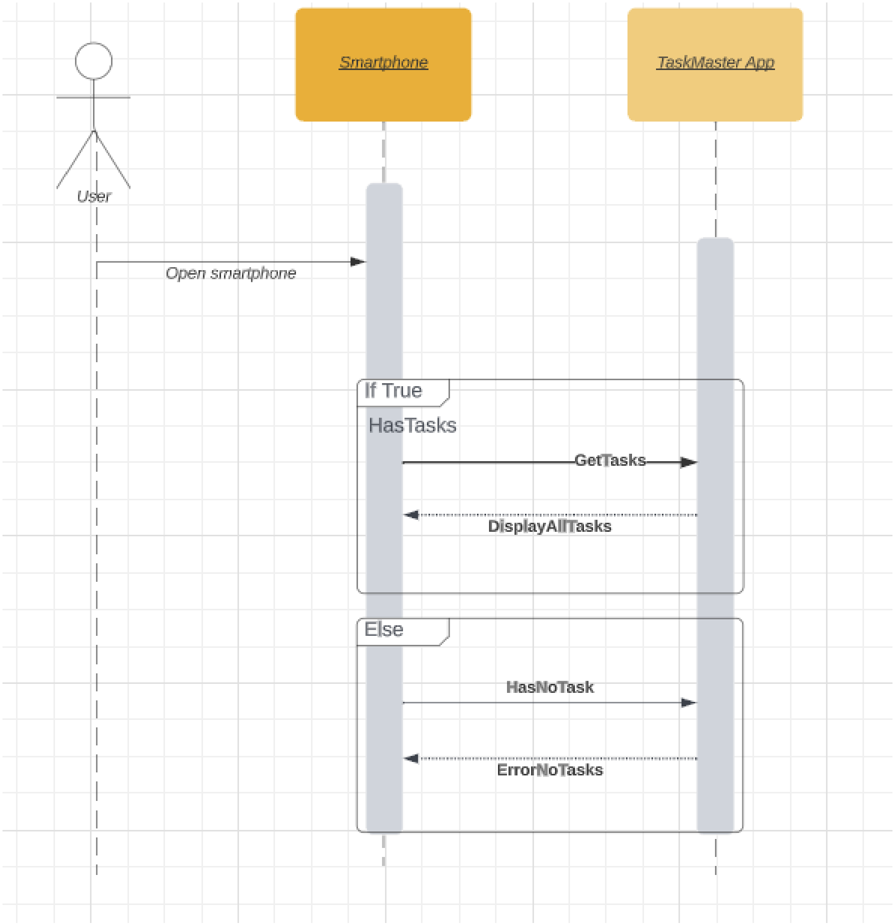
Use Case 1: Display Tasks List

Preconditions
User must be registered and logged into the "TaskMaster" app.
User has previously created tasks on the app.

Main Flow
User navigates to the main dashboard of the app [S1].
The system displays all the tasks created by the user in order of their creation date or priority [S2].

Subflows
[S1] User can access the main dashboard by clicking on the "Tasks" tab located at the bottom of the screen.
[S2] Tasks are displayed with their titles, priority stars, and due dates. The user can scroll through the list to view all tasks.

Alternative Flows
[E1] If no tasks have been created, the system displays a message prompting the user to add their first task.
[E2] If there's a sync issue, the user is informed about it and asked to try refreshing the task list.



Use Case 2: Schedule a Task

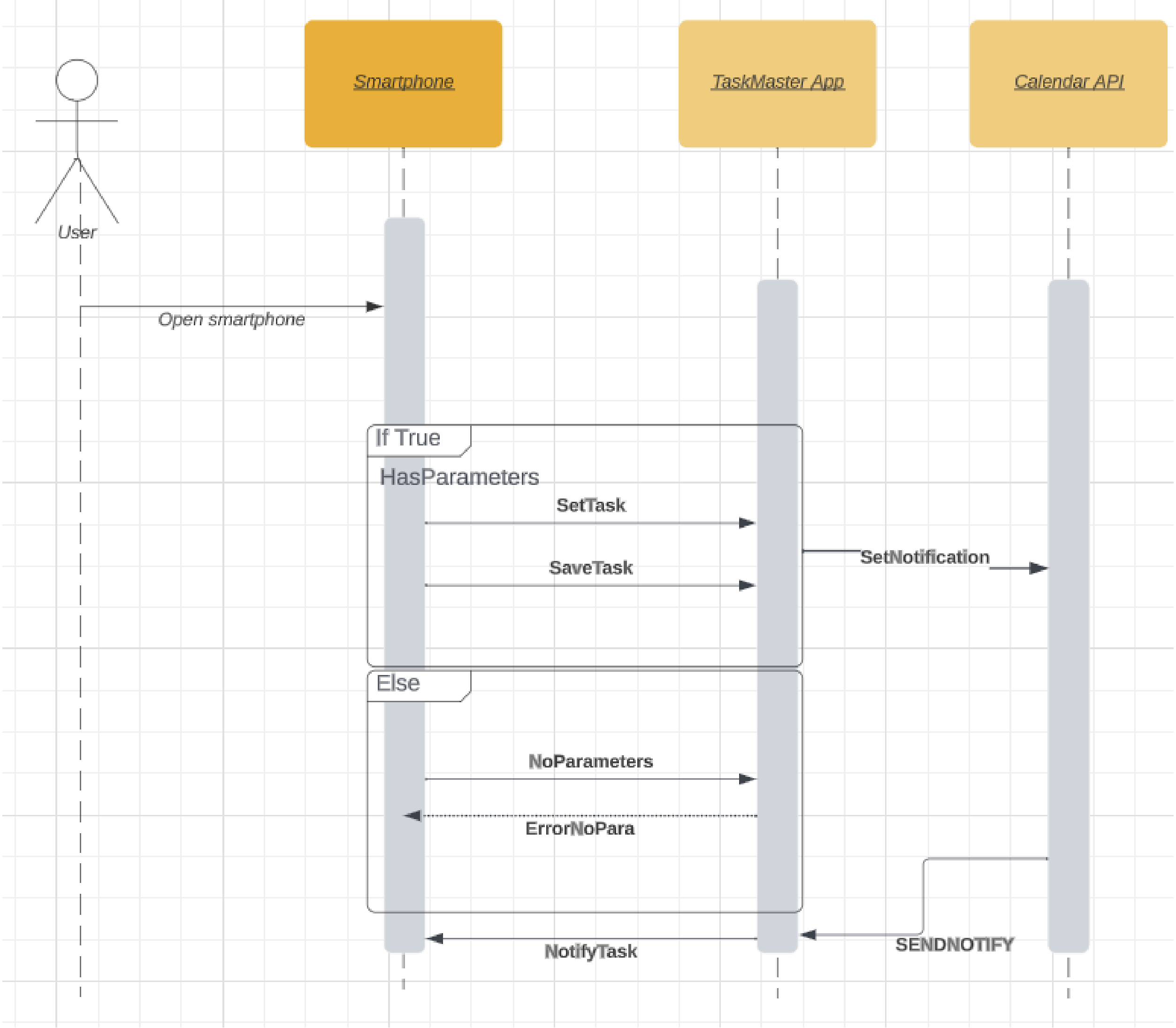
Preconditions
User must be registered and logged into the "TaskMaster" app.

Main Flow
User clicks on the "Add Task" button [S1].
User provides the task title, sets the due date, and clicks on "Save" [S2].
The system validates the input and adds the task to the scheduled tasks list [S3].

Subflows
[S1] The "Add Task" button is prominently displayed on the main dashboard.
[S2] The system provides a calendar widget to help the user select the due date easily.
[S3] A notification or a toast message confirms the task's addition to the list.

Alternative Flows
[E1] If the user does not provide a title or a due date, the system prompts an error message requesting the missing details.

[E2] If there's a technical issue during the task creation, the user is informed and asked to try again.



Use Case 3: Prioritize Tasks using Stars System

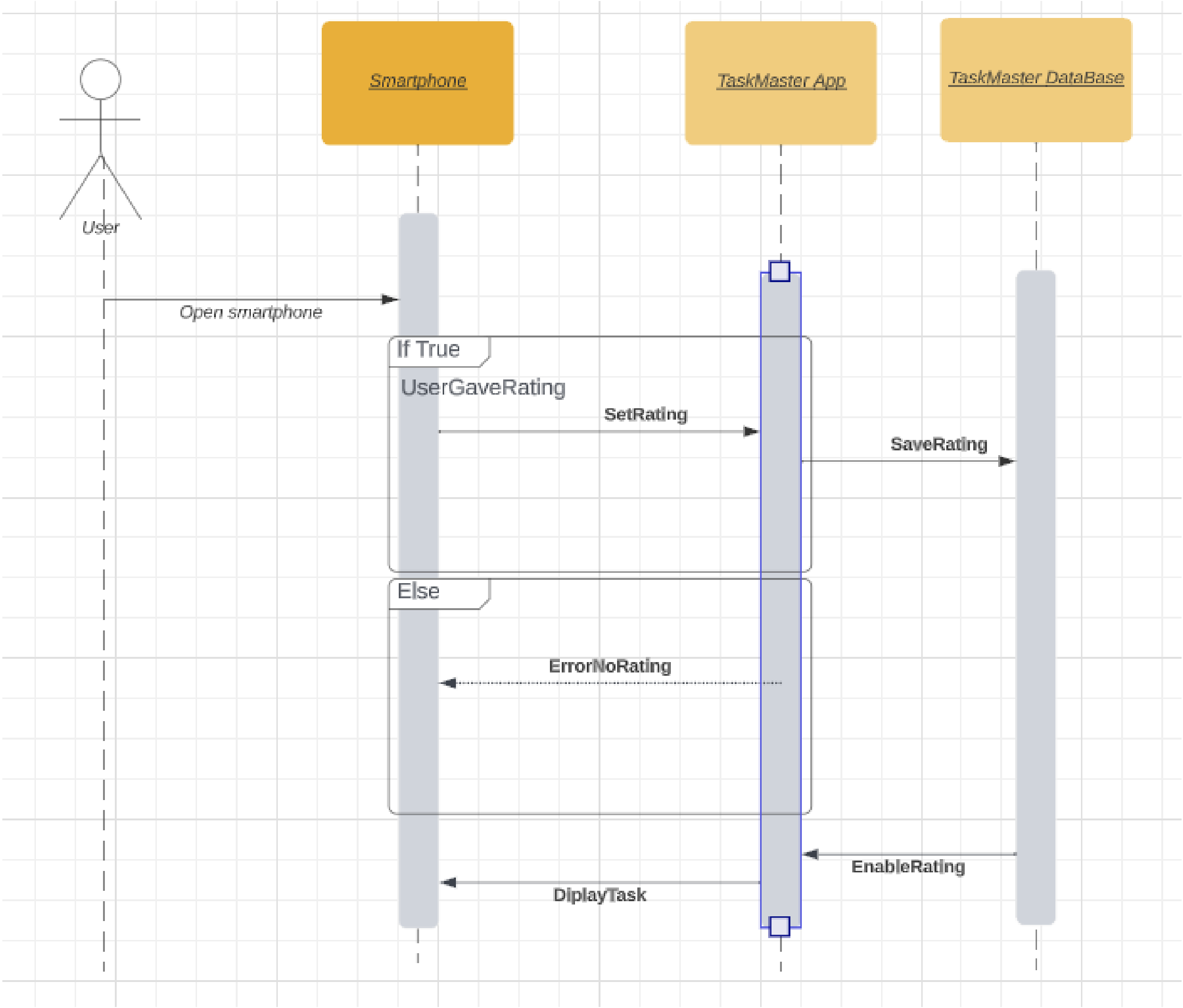
Preconditions
User must be registered and logged into the "TaskMaster" app.
User is viewing an existing task or creating a new one.

Main Flow
User assigns a star rating to the task ranging from 0 to 3 stars [S1].
Upon submission, the system updates the task's priority based on the star rating [S2].

Subflows
[S1] A set of star icons is presented to the user, allowing them to tap the desired star rating.

[S2] The task list on the main dashboard is reordered based on the priority given through the star system.

Alternative Flows
[E1] If the user doesn't select any stars, the task is set to the default priority (0 stars).
[E2] If there's a technical issue during the priority update, the user is informed and asked to try again.



Use Case 4: Remove Finished Tasks

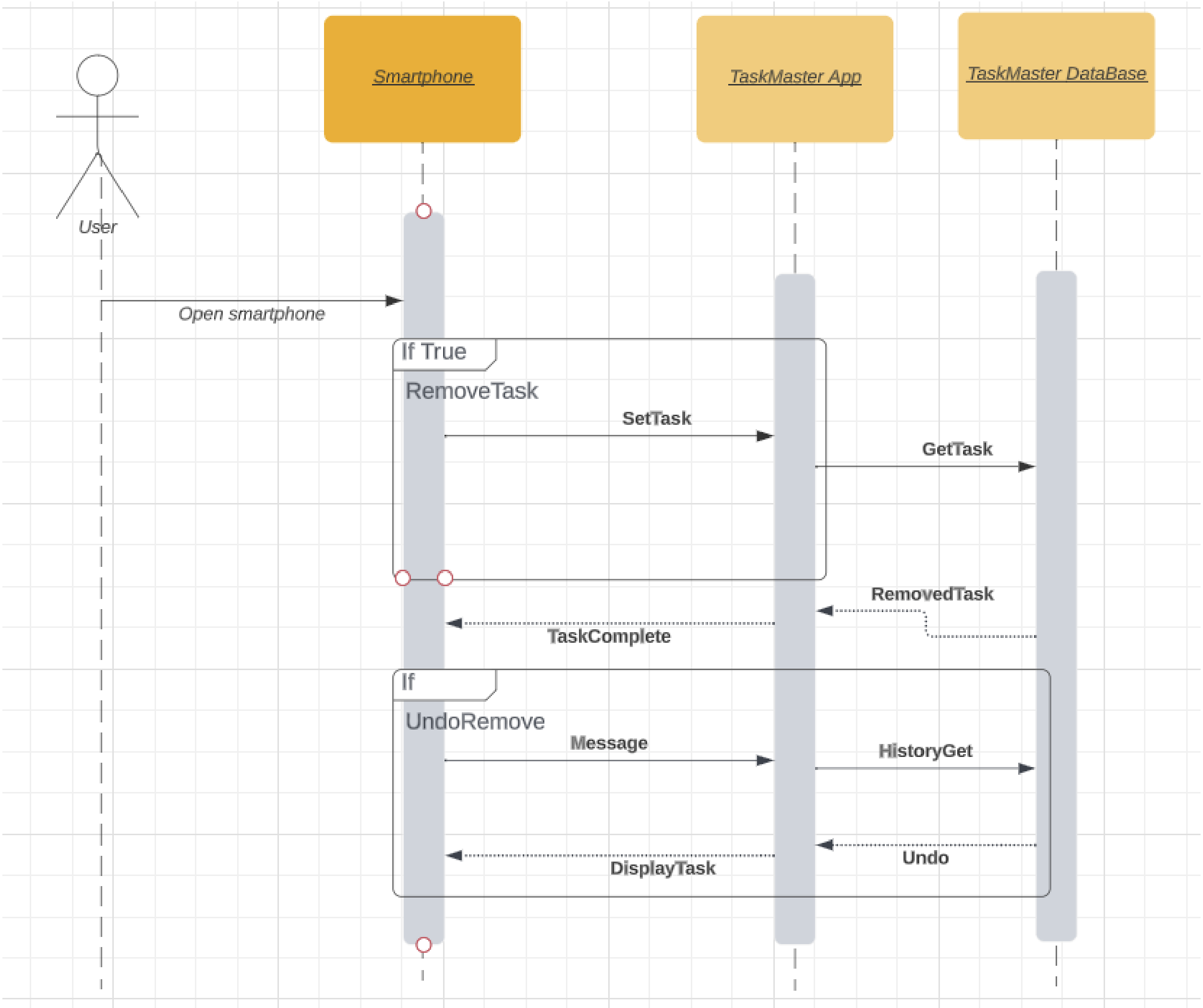
Preconditions
User must be registered and logged into the "TaskMaster" app.
User has tasks marked as completed.

Main Flow

User selects a task and clicks on the "Mark as Complete" button [S1].
The system prompts a confirmation message, and upon user confirmation, the task is removed from the active task list [S2].

Subflows
[S1] Each task has an associated checkbox or a completion button.
[S2] Removed tasks may be archived in a "Completed Tasks" section for future reference.

Alternative Flows
[E1] If the user mistakenly marks a task as complete, there's an option to undo the action immediately.
[E2] If there's a technical issue during task removal, the user is informed and asked to try again.



Use Case 5: Edit a Task

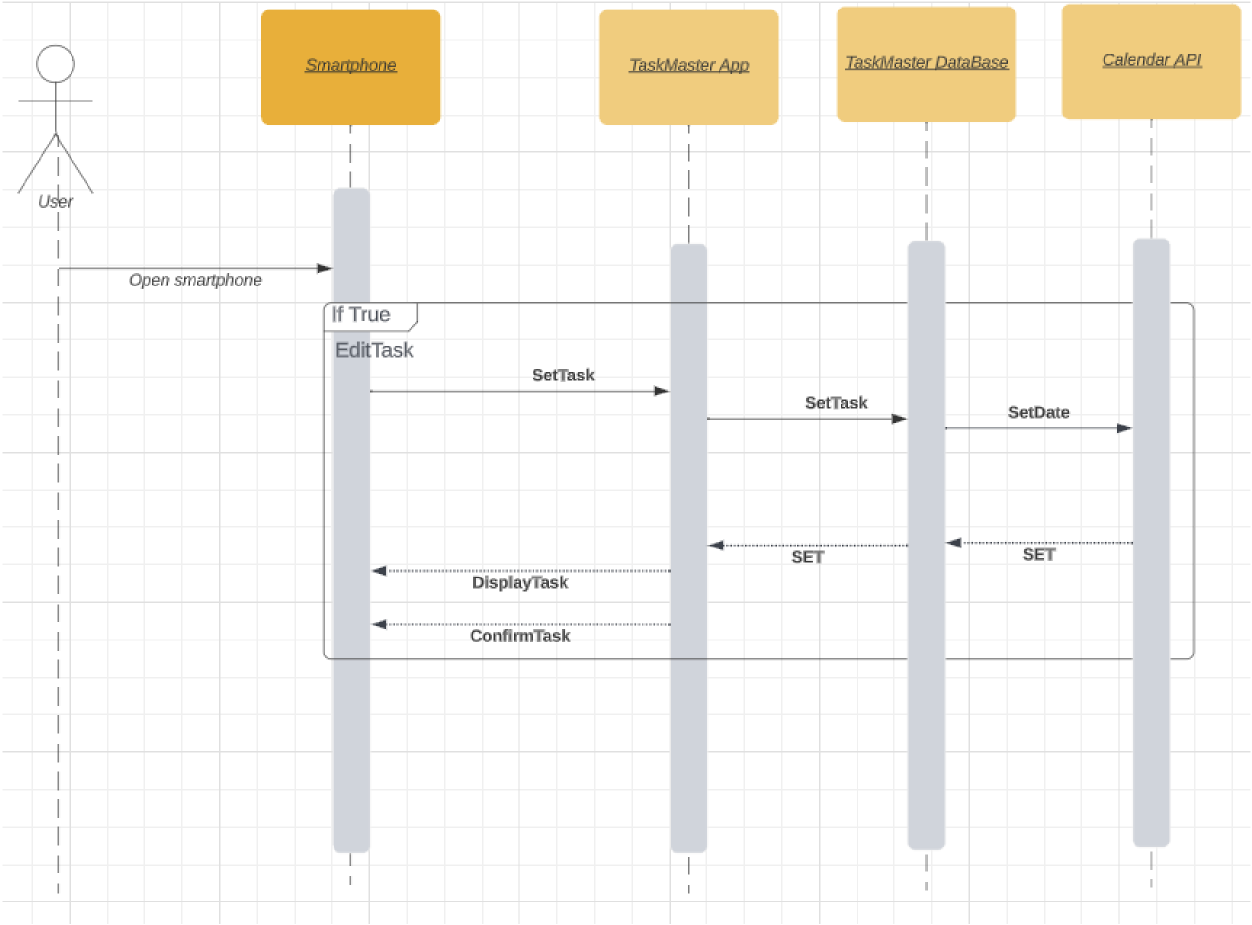
Preconditions
User must be registered and logged into the "TaskMaster" app.

User wishes to modify an existing task.

Main Flow
User selects a task from the list and clicks on the "Edit" button [S1].
User modifies the task details such as the title, due date, or star rating and then clicks "Save" [S2].
The system updates the task with the new details and confirms the changes to the user [S3].

Subflows
[S1] The "Edit" button is accessible from a task's options or by tapping on the task itself.
[S2] The editing interface resembles the task creation interface, but with pre-filled details of the task.
[S3] A notification or toast message confirms the successful modification of the task.

Alternative Flows
[E1] If the user attempts to save the task without making any changes, the system might display a message indicating no changes detected.
[E2] If there's a technical issue during the task update, the user is informed and asked to try again.

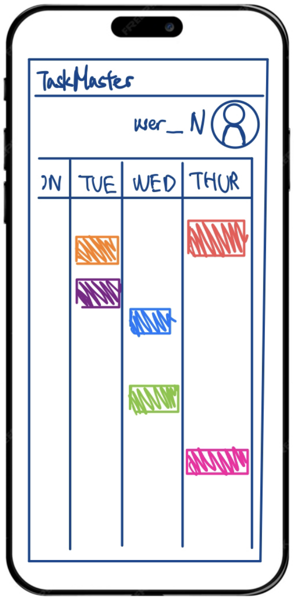


Process Deliverable (2%): Prototyping

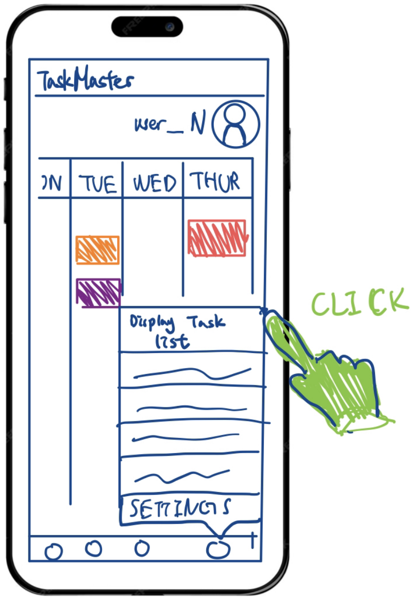
USE CASE 1:

DISPLAY TASKS LIST

1

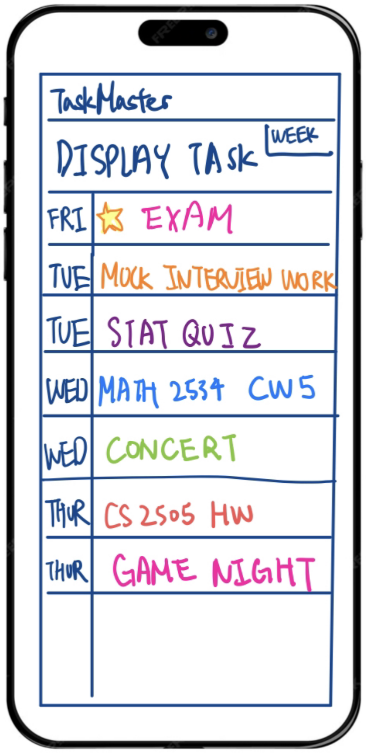


2



User must be registered and login
Has previously created tasks on the app

3



TaskMaster

DISPLAY TASK

WEEK

FRI	☆ EXAM
TUE	MOCK INTERVIEW WORK
TUE	STAT QUIZ
WED	MATH 2534 CW5
WED	CONCERT
THUR	CS 2505 HW
THUR	GAME NIGHT

User can choose to display it weekly, monthly, yearly
The list will show in sorted order (by priority star/ date/ time)
The text box will be filled in the same colour as it is shown on the main dashboard