Requirements Workshop (3%)
We will have a requirements workshop in class on September 15. This workshop should be completed and turned in with the remaining materials for PM1.

Completed in PM1

Requirements Analysis (5%)
Requirements analysis is the process of understanding the requirements for a software application. This deliverable will include 5:

Fully dressed use cases:

USE CASE 1: *Personal Productivity and Daily Planning.*
Scenario: Sean is a busy professional who wants to manage his daily tasks efficiently.
Use Case: He enters his daily activities into the TaskMaster and prioritizes them according to importance and urgency. He marks each assignment off as he completes it. In order to prevent him from forgetting anything, the app also sends reminders for high-priority activities. If a task is missed or delayed, he can reschedule it in the future with the TaskMaster.
Stakeholders: Sean.
Preconditions: Sean has the to-do app installed and an account set up.
Postconditions: Tasks for the day are listed, prioritized, and checked off upon completion.
Step-by-step Scenario:
Sean launched the to-do app.
He adds tasks for the day.
Prioritizes tasks based on urgency and importance.
Receives reminders for high-priority tasks.
Checks off tasks as completed.
If a task is delayed, Sean reschedules it using the app.
Requirements: Push notifications enabled for reminders.

USE CASE 2: Collaborative Project Management.
Scenario: A team of developers and designers are working on launching a new website.
Use Case: The team assigns tasks, sets deadlines, and keeps track of progress using the TaskMaster. They can label tasks according to the project phase, provide comments, and receive notifications when a deadline for a high-priority task is approaching. This cooperative strategy guarantees effective communication and keeps the team on track.
Stakeholders:
Developers: Need to clearly understand, keep track of assigned tasks, and complete them on time.
Clients: Efficient working project that adheres to the deadline.

Preconditions: Team members have the TaskMaster installed and accounts set up.
Postconditions: Tasks are assigned, monitored, prioritized, and checked off upon completion.
Step-by-step Scenario:
The team leader creates tasks.
Assign tasks to respective team members.
Team members update task status and add comments.
If a team member is unavailable, tasks are reassigned.
Deadlines are tracked and adhered to.
Project completion is achieved.
Requirements:
Push notifications are enabled for reminders.

USE CASE 3: Event Planning.
Scenario: Akshath is planning a wedding and needs to keep track of numerous tasks, from booking venues to sending invitations and decorating the hall.
Use Case: Akshath makes separate lists for each aspect of the event, such as the location, catering, dress, guest list, invitation list, flower decor, etc. He sets deadlines and reminders for paying deposits and getting confirmation from vendors. He also prioritizes the tasks in order of importance set by his boss.
Stakeholders:
Akshath: Make sure that the wedding is well organized with no hiccups.
Guests: Expect a well-organized wedding with.
Preconditions: Akshath has the to-do app installed and an account set up.
Postconditions: Tasks for the wedding are listed, prioritized, and checked off upon completion.
Step-by-step Scenario:
Akshath creates lists for different wedding aspects (venue, catering, decor, invitations, etc.).
Add tasks, and deadlines, and set reminders.
Monitors task progress and adjusts as needed.
If a vendor cancels, search for alternatives and update the respective list with new information.
Completes all preparations by the wedding date.
Requirements: Push notifications enabled for reminders.

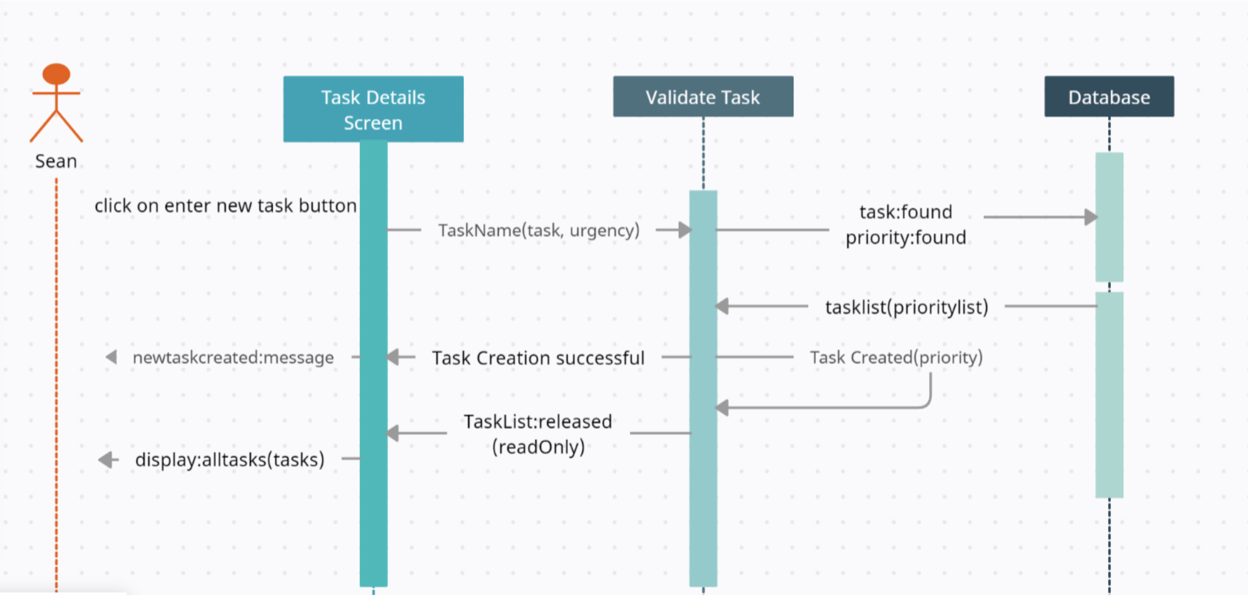
USE CASE 4: Habit Tracking and Personal Development.
Scenario: Abishek wants to practice new habits like reading, yoga, hitting the gym, and drinking more water every day.
Use Case: Abishek sets up recurring tasks in the TaskMaster to remind him of his new daily habits.
Stakeholders:
Abishek.
Preconditions: Abishek has the to-do app installed and an account set up.

Postconditions: Tasks for the new habit goals are set according to priority.
Step-by-step Scenario:
Abishek adds desired daily habits as tasks in the TaskMaster.
Sets up recurring reminders.
Checks off habits as they're completed each day.
If a habit is missed, he reschedules or adjusts the habit.
Requirements: Push notifications enabled for reminders.

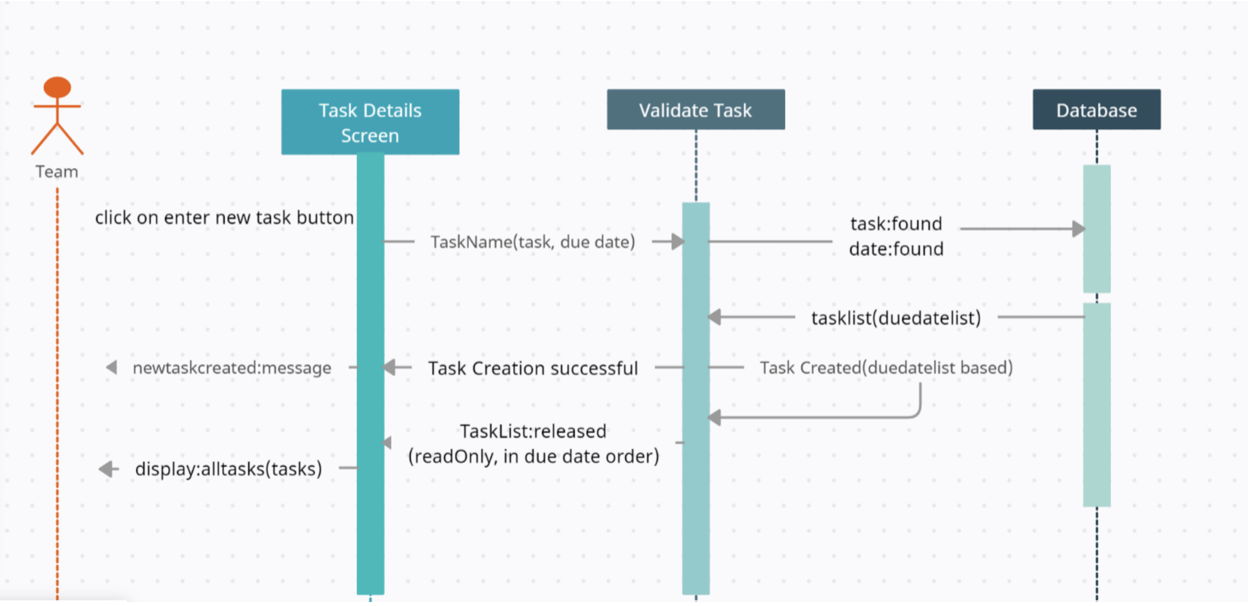
USE CASE 5: Travel Planning.
Scenario: Prahaara is planning a two-week vacation across Europe and needs to organize his itinerary, bookings, and activities for each destination.
Use Case: Prahaara makes a unique list for each city he intends to visit using the TaskMaster. He adds tasks for each place, such as checking in, buying train tickets, looking up nearby sights, and noting down recommended restaurants. To make sure he doesn't forget anything, he sets reminders for time-sensitive tasks like check-in.
Stakeholders:
Prahaara: Wants a well-organized and stress-free vacation.
Preconditions: Prahaara has the to-do app installed and an account set up.
Postconditions: All travel-related tasks are listed, and tracked, and relevant bookings are made.
Step-by-step Scenario:
Prahaara creates a list for each European city he plans to visit.
Adds tasks related to bookings, activities, and cuisine.
Sets reminders for time-sensitive tasks.
Requirements: Push notifications enabled for reminders.

Model (use case or sequence diagrams) for representing each use case:

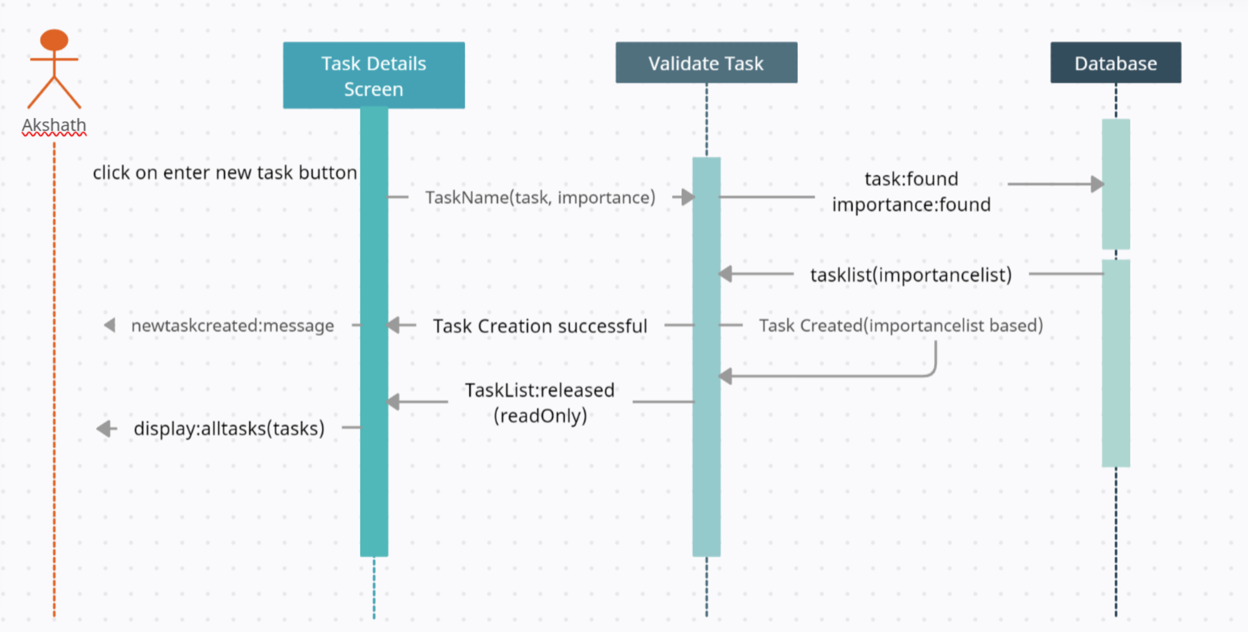
USE CASE 1 - Sequence diagrams:



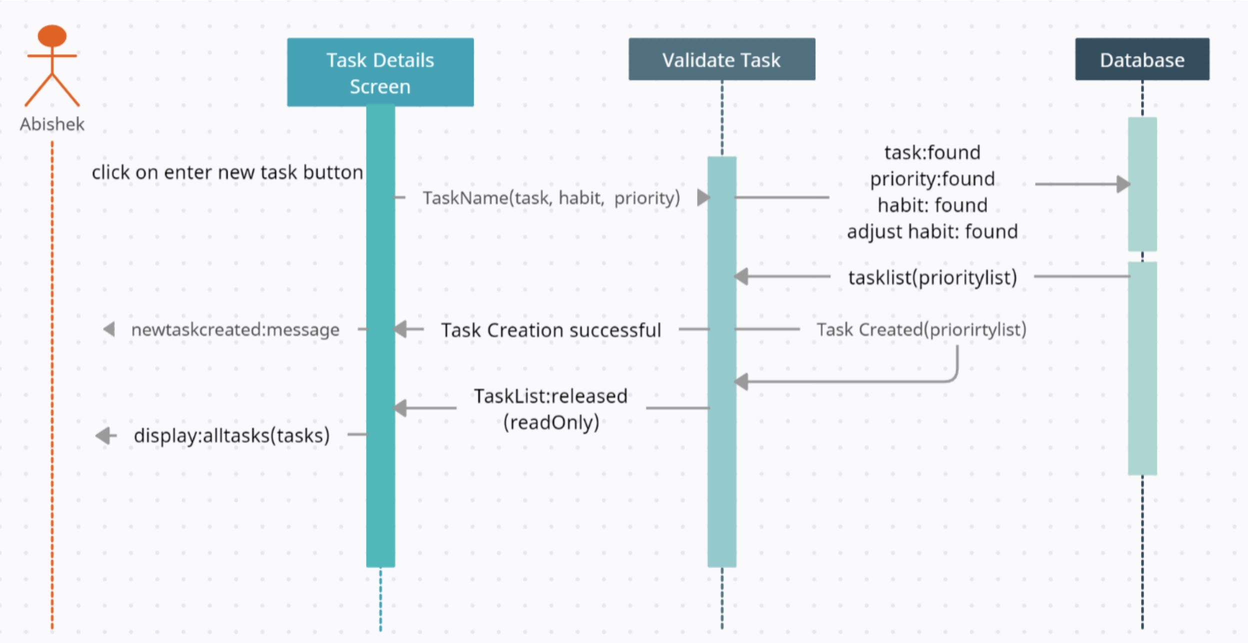
USE CASE 2 - Sequence diagram:



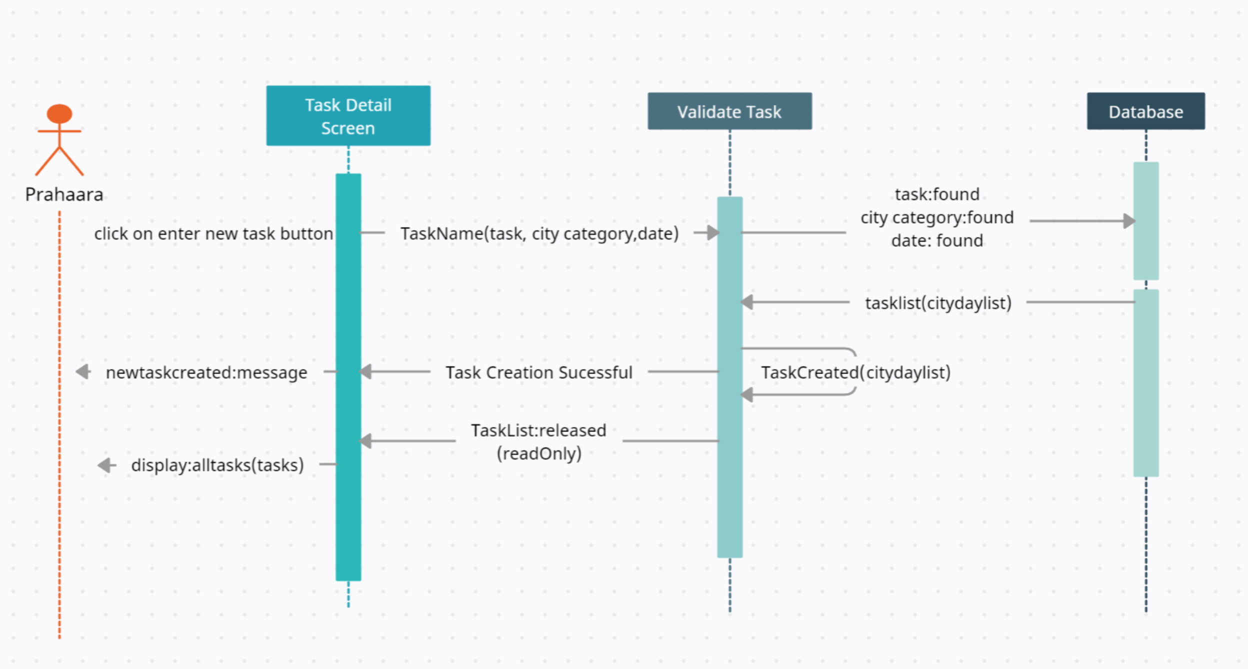
USE CASE 3 - Sequence diagram:



USE CASE 4 - Sequence diagrams:



USE CASE 5 - Sequence diagrams:



Process Deliverable (2%)

The submission for this deliverable will depend on the specific SE process model your team plans to use to complete the group project (as described in your project proposal). Example submissions for different processes include:

- ***Prototyping: submit a prototype of your system (it can be as formal/informal as needed)***
- ***Scrum: submit the notes (including each teammate) from your most recent scrum meeting***
- ***Kanban: submit a list of prioritized tasks from your task management system (and why they are prioritized)***
- ***Waterfall: submit supplementary planning documentation***
- ***Extreme programming: submit acceptance test criteria***
- ***Spiral: submit risk analysis***
- ***Code-and-fix: essay on why you used code-and-fix or up-to-date source code in your GitHub repository For other processes not listed above, the instructor will contact you with the exact submission requirements for this task.***

Prototype Figma Link:
https://www.figma.com/file/4P3nnGH0XAwUFPnR6c8U7K/TaskMaster?type=design&node-id=1%3A8499&mode=design&t=NH36JJq6Cs2roOD3-1

Screenshots:

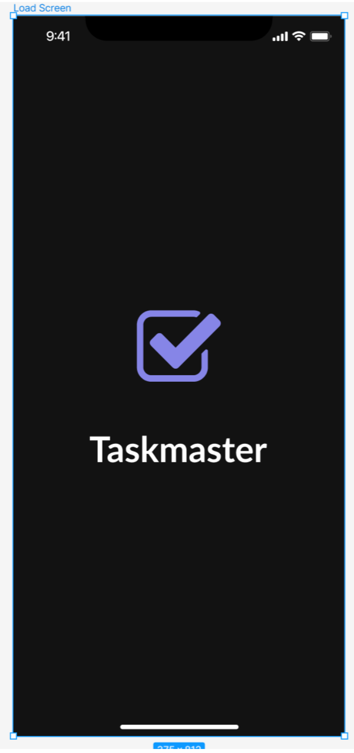
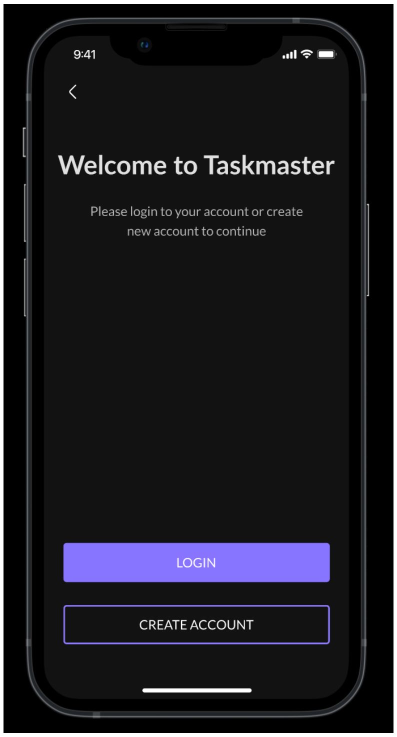


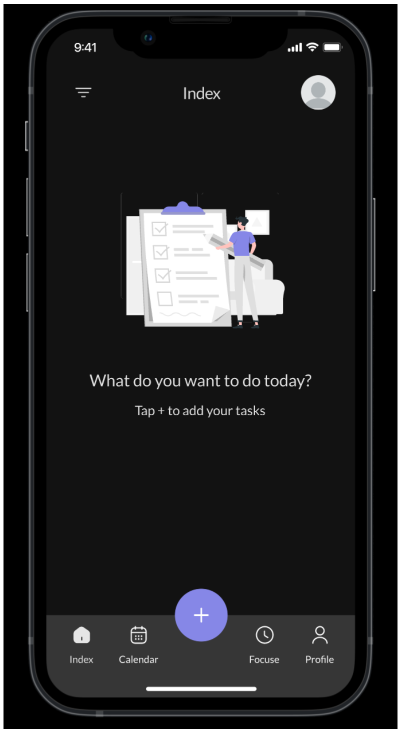
Figure 1: Load Screen



Welcome to Taskmaster

Please login to your account or create new account to continue

Figure 2: Start Screen



What do you want to do today?

Tap + to add your tasks

Figure 3: Home Screen

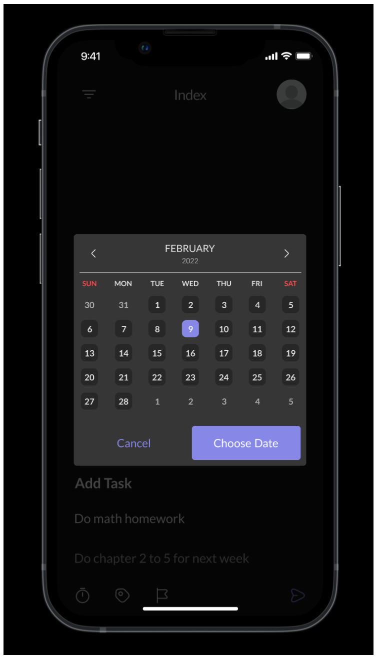


Figure 4: Task Date

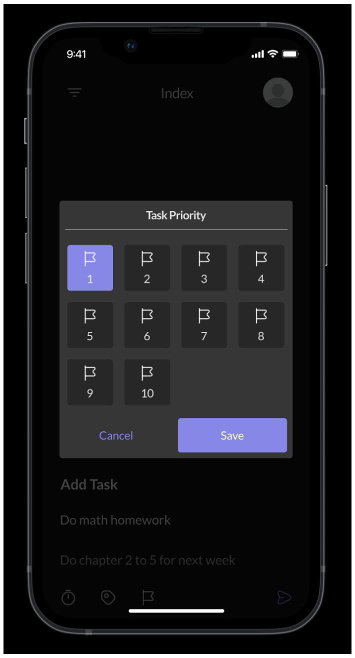


Figure 5: Task Priority

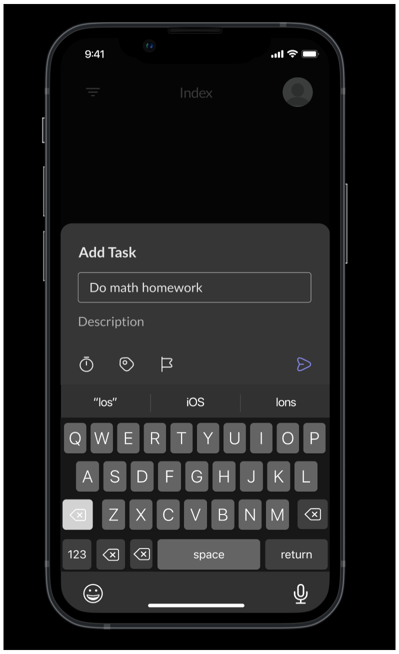


Figure 6: Home Screen with Tasks

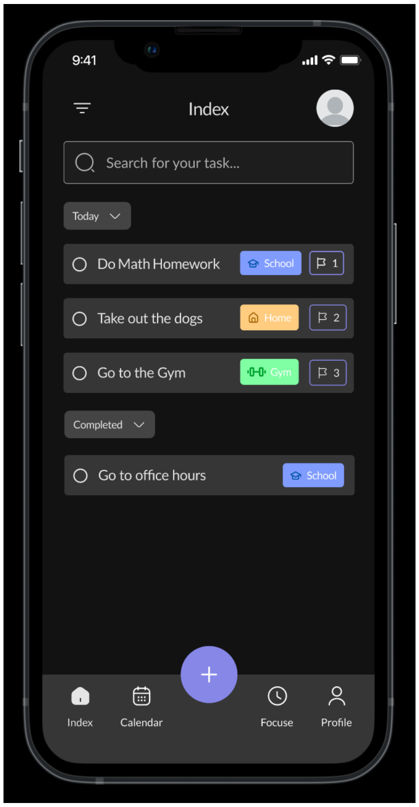


Figure 7: Home Menu with Tasks

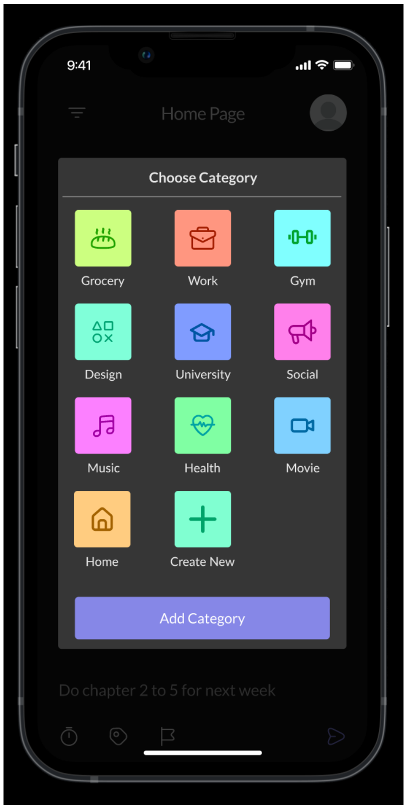


Figure 8: Choose Categories

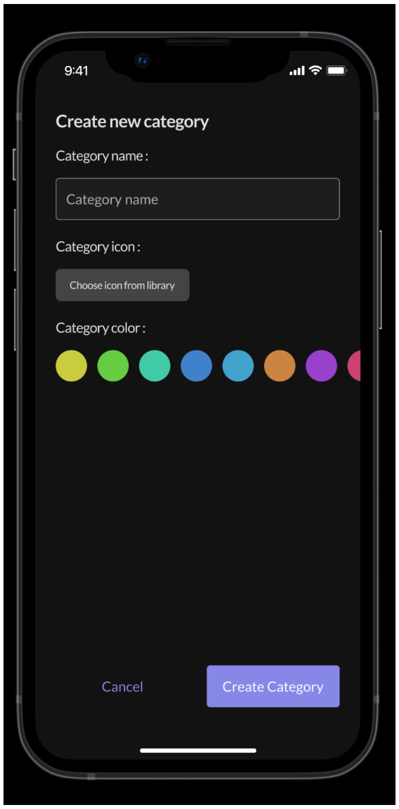


Figure 9: Create category