# Overview

There are plenty of task management apps available, but none that seem to do quite what we need in the way that we want. The purpose of this project is to develop an app that will allow for better task management.

# Tasks

All tasks have some attributes in common.

**Name**: the brief text that describes the work that the completing the task will accomplish
**Notes**: any additional information that can be added for context

Tasks may optionally have a **due date**.

There are two different task type types: one-time, and repeating.

I should also be able to mark any task as Complete, Skip (in the case that I want to ignore a repeating task for the current time interval – more on that below), or Cancel (in the case that I want to complete delete a task).

## One-time tasks

One-time tasks are tasks that are created for a one-time task. If they do not have a due date, they persist until completed or canceled.

## Repeating

Repeating tasks repeat based on a time span. Time spans should include:
- **Monthly**
- **Weekly**
- **Daily**
- **Every N days** (where N is provided by the user)

Repeating tasks that are not done by the end of their time frame should roll into the next iteration of their time frame without creating another task.

Monthly, Weekly, and Daily tasks that are completed should not have their next recurrence set until the beginning of the next iteration. For example, if you complete a Weekly task on

Wednesday, the next iteration of that Weekly task wouldn't show up until the first day of the next week (either Sunday or Monday, depending on the development frameworks used).

Tasks that repeat every N days should have their next due date set relative to the date of their actual completion, not based on a set calendar.

# Use Cases

## One-time task, no due date

Someone has recommended that I watch the movie John Wick 4. I would create a one-time task, entering "Watch John Wick 4" as the name, adding a note about who recommended I watch it. I would not assign a due date.

I can mark the task as complete at any point, but until I do, it remains active.

## One-time task, due date

Someone has asked me to send them a movie review of John Wick 4 by June 10th. I would create a one-time task, entering "Review John Wick 4 for Someone", assigning the date on which they wanted the review sent. I probably wouldn't enter any notes for this task.

I can complete this task at any point prior to or on the due date. If, for some reason, I miss the due date, the task should remain active.

## Repeating task, monthly

Once a month, I need to review my finances. It doesn't matter on which particular day I do the review. I would create a repeating task, entering "Review finances" as the name, and setting the repeat interval to Monthly. I probably wouldn't enter any notes for this task.

When I mark the task complete on the last day of the month, it disappears until the next day (which happens to be the first day of the following month).

- Note that if I'd managed to get the finances reviewed mid-month, the task would have been gone for at least a couple of weeks.

- Note also that if I didn't get the finances reviewed before the beginning of the next month, the task would effectively roll forward into the new month. There should not be an additional task created for the new month iteration.

## Repeating task, every 10 days

I need to water my plants every 10 days. I would create a repeating task, entering "Water plants" as the name, and setting the repeat interval to Every N Days, with 10 specified. I might enter some notes about how much water to give each plant.

A week later, I remember to water the plants. I mark the task as complete, and another task is immediately generated with a due date 10 days from now.

Two weeks later, I remember to water the plants. I mark the task as complete, and another task is immediately generated with a due date 10 days from now.

Every time I mark this task as complete, a new task should be generated with a due date of N days after today. This is true regardless of whether or not I actually completed this task by the original due date.