

## TODO Tracker

---

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any edge or corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Describe how you will implement Google Play Services or other external services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Link dispatcher](#)

[Task 4: Release task](#)

[Task 4: Test](#)

**GitHub Username:** ananthakrishnac

# ToDo Tracker

## Description

This app lists tasks and time intended to be completed by End of Day - TODAY.

Most of us would have a Dairy / Sheet of paper / Sticky Notes / other means of tracking what needs to be done today - Atleast I do. Other ToDo tracker apps, clutter the UI with left over tasks from yester times and todo from tomorrow. This app, just displays Tasks to be done today. Adds reminder and Widget featuring next task todo.

## Intended User

## TODO Tracker

Everyone who makes a list morning for tasks they intend to complete that day.

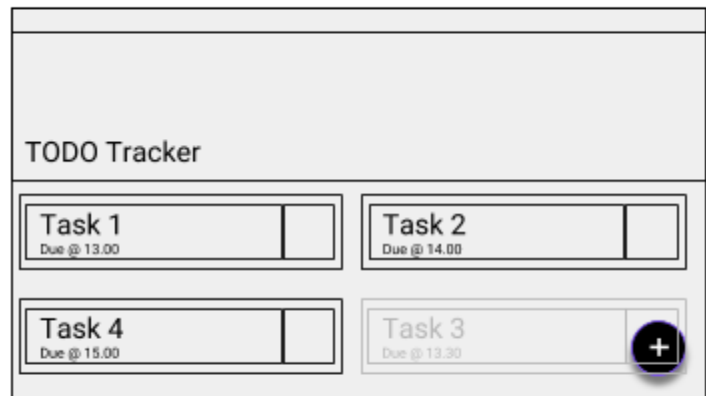
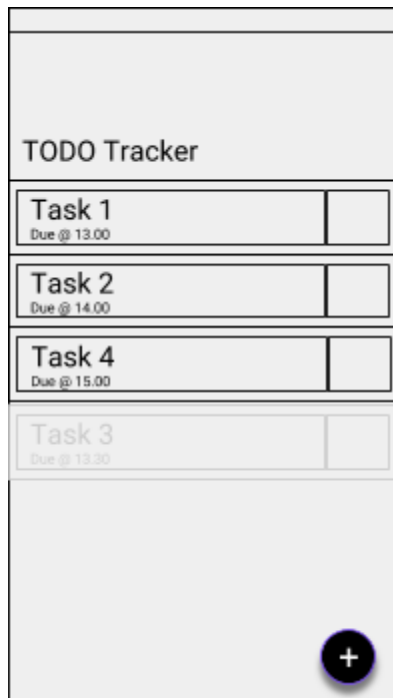
## Features

Main features:

- Tracks only daily tasks
- Shows reminders
- Shows completed tasks at the very end along with pending tasks.

## User Interface Mocks

### Screen 1

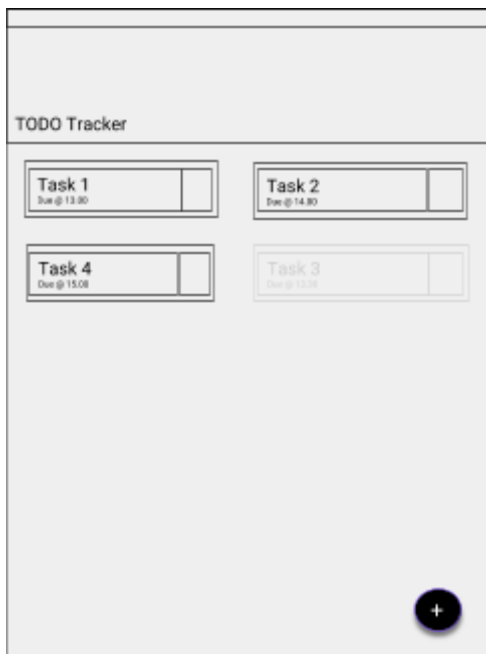
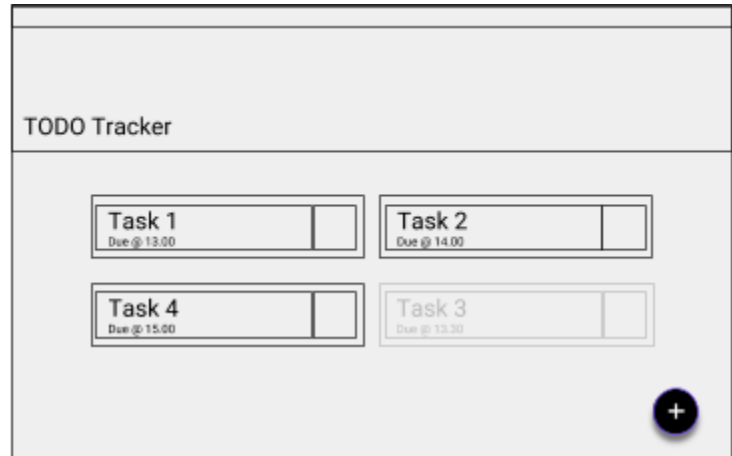


Portrait and Landscape versions of Initial screen for Mobile Phones.

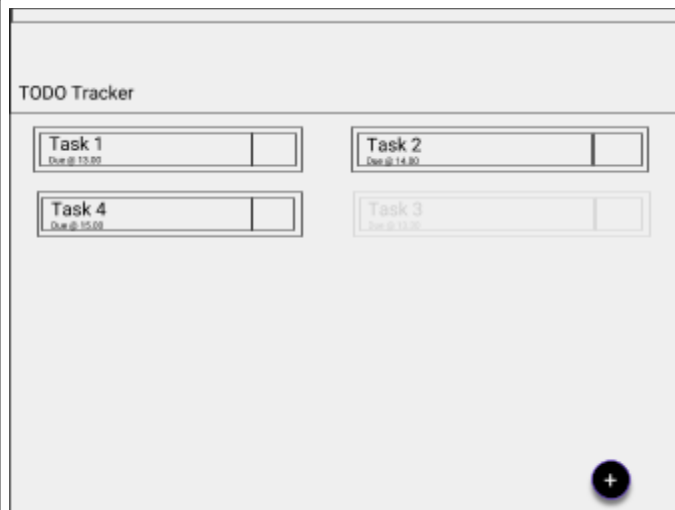
## TODO Tracker



Mid size Tablet version Initial screen

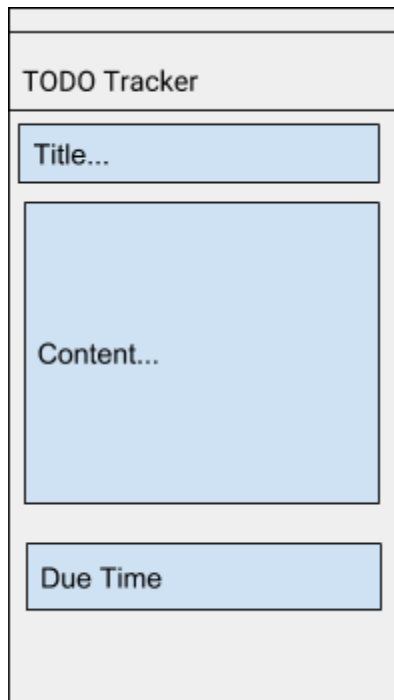


Full size Tablet version Initial Screen

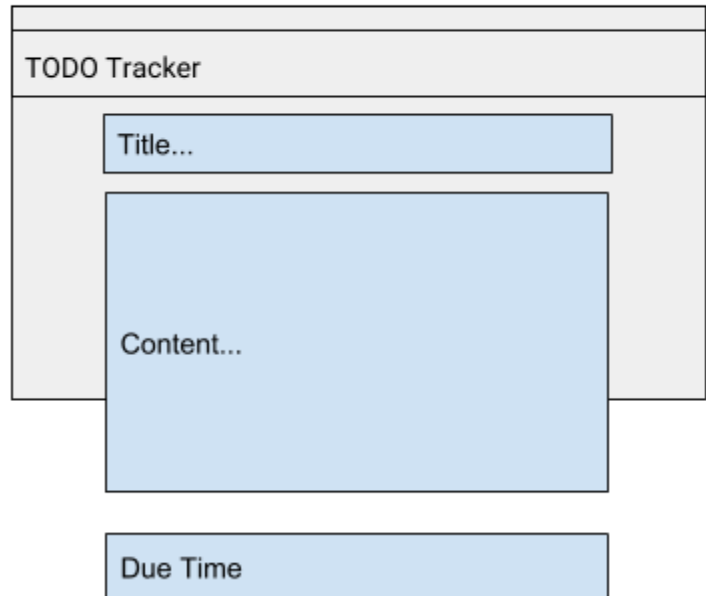


## TODO Tracker

## Screen 2

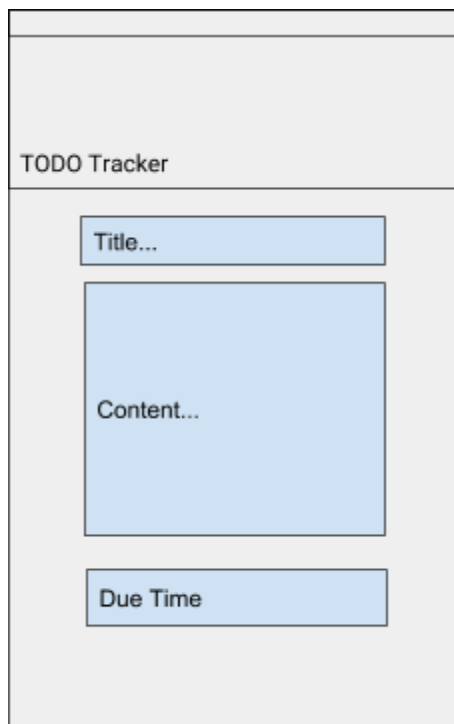


Mobile size layout for Screen 2. The screen has a light gray header bar with the text "TODO Tracker". Below the header, there is a blue input field labeled "Title...". Underneath the title field is a large blue rectangular area labeled "Content...". At the bottom of the screen is another blue input field labeled "Due Time".

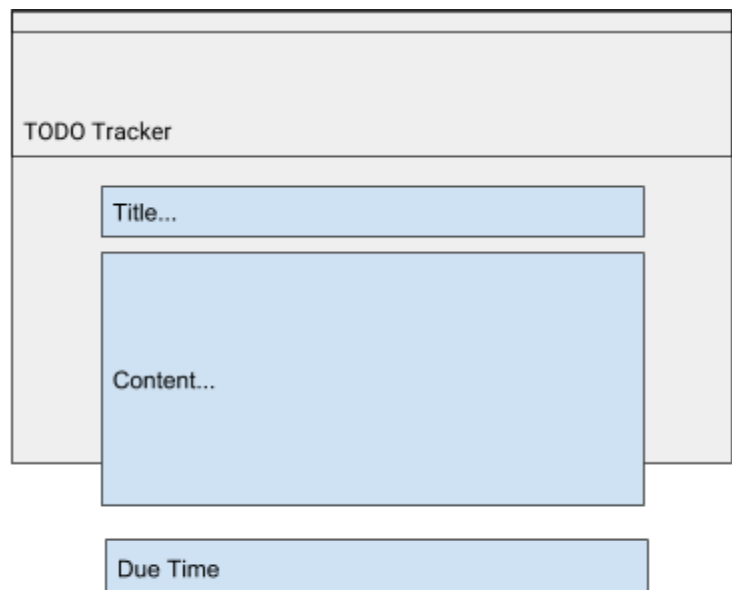


Tablet size layout for Screen 2. The screen has a light gray header bar with the text "TODO Tracker". Below the header, there is a blue input field labeled "Title...". Underneath the title field is a large blue rectangular area labeled "Content...". At the bottom of the screen is another blue input field labeled "Due Time".

Mobile size: Detail screen where contents of a Task are displayed. Blank screen opens when we use FAB to enter new Task.



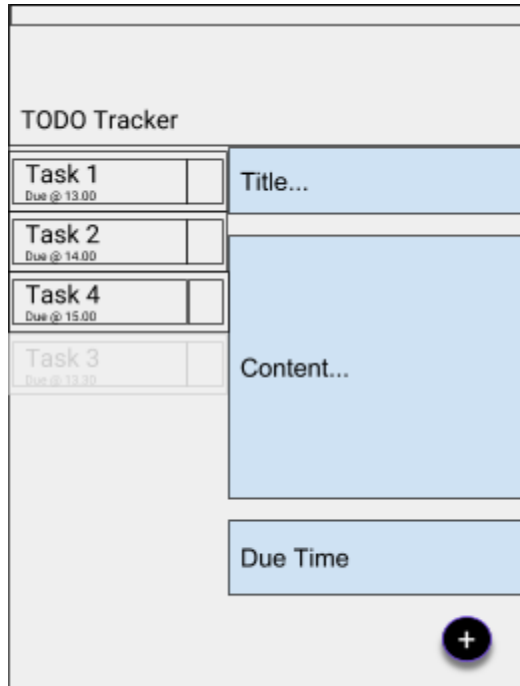
Midsize Tablet layout for Screen 2. The screen has a light gray header bar with the text "TODO Tracker". Below the header, there is a blue input field labeled "Title...". Underneath the title field is a large blue rectangular area labeled "Content...". At the bottom of the screen is another blue input field labeled "Due Time".



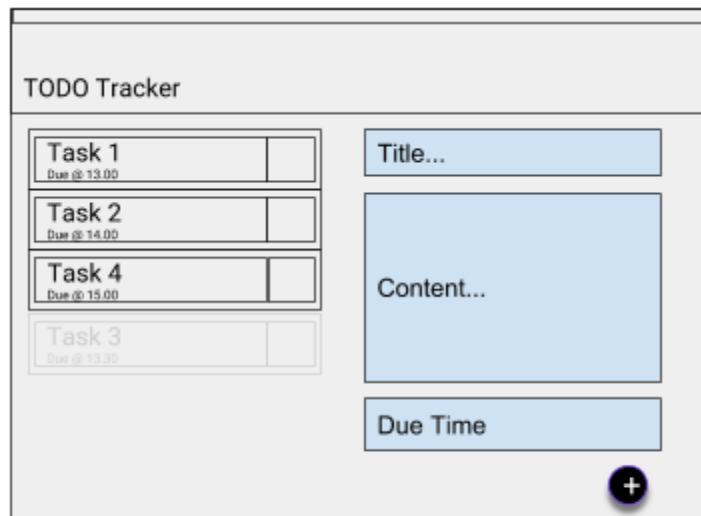
Large Tablet layout for Screen 2. The screen has a light gray header bar with the text "TODO Tracker". Below the header, there is a blue input field labeled "Title...". Underneath the title field is a large blue rectangular area labeled "Content...". At the bottom of the screen is another blue input field labeled "Due Time".

Midsize Tablet: Detailed screen

## TODO Tracker



Full size Tablet: Master Detailed Screen.



This will be accompanied by Reminder and Widget.

In the Detailed screen, there will be an option to discard changes and delete Item.

## Key Considerations

### How will your app handle data persistence?

Content Provider will be used for Data persistency.

### Describe any edge or corner cases in the UX.

None.

### Describe any libraries you'll be using and share your reasoning for including them.

Firebase-jobdispatcher: Firebase job scheduling.

## TODO Tracker

**Describe how you will implement Google Play Services or other external services.**

Firebase Job Dispatcher will be used for reminder services.

## Next Steps: Required Tasks

### Task 1: Project Setup

- Create a new Project
- Add Firebase library into build.gradle

### Task 2: Implement UI for Each Activity and Fragment

- Build Generic UI for MainActivity
- Extend the task and implement Landscape Layout.
- Build UI for detailed Activity.
- Extend these to various sizes. - Set width, margin and Layout Columns , weight to adjust and fit different screens.
- Implement Master/Detail flow for Large Tablet sizes.
- Implement Widget
- Implement Notification

### Task 3: Link dispatcher

- Implement Firebase Job dispatcher
- Link Notification and Firebase Job dispatcher.

### Task 4: Release task

- Build release variant

### Task 4: Test

- Test
- Test
- Submit