

## To-do app: React Native code challenge

### Intro

The goal of this challenge is to build a simple To-Do mobile application in React Native. You can use [Expo](#) to develop this application or React Native Client.

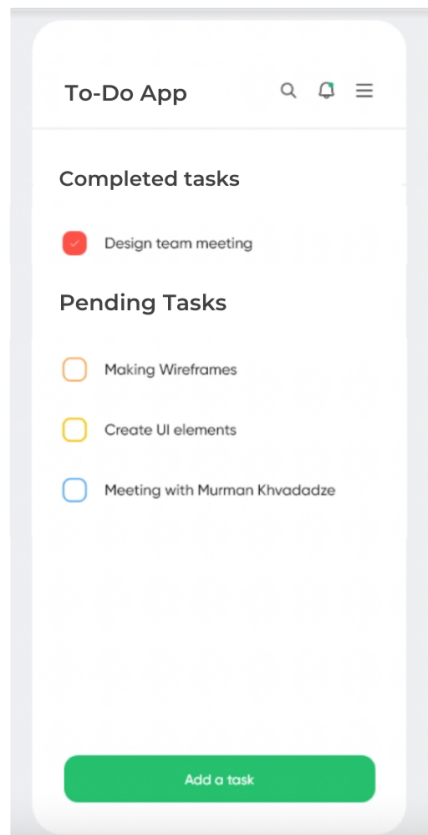
### Home Screen

In this screen the user can see and manage all the tasks that were created and the user can select as a **“complete” with the checkbox input**. In the section Pending Tasks: The tasks appear in the order of "creation" ordered from last to previous.

**Also if the user marks an input check as a complete task, this particular task will disappear from the Pending Task Section and appear in the completed tasks section.**

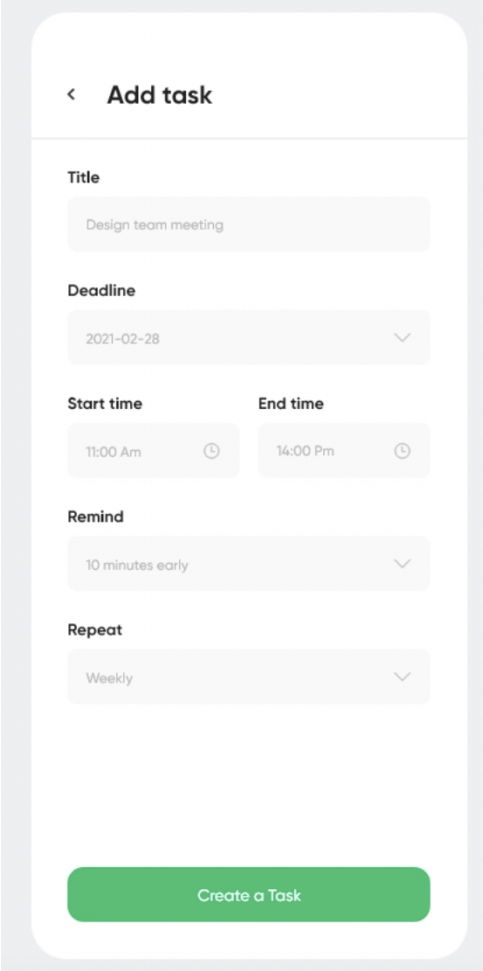
Favorite tab, Search, Alert, Hamburger Menu is not required to the scope of this test we only need the ui for those features. When the user taps “Add a task” the app navigates to **“AddTaskScreen”**.

**Important Feature: We need that all data will be persisted. So when the user closes the app and opens again, it will show all the tasks that were created. You Don't need to create or use any API or backend, just the local bd on the phone.**



## Add Tasks Screen

In this screen the user can create new tasks to the to-do app. Please follow the exact design as possible. The user **can create a new task** that will be visible on the other screen (Home). The new task will appear in the pending tasks section.

A mobile app screen titled "Add task" with a back arrow. It contains several form fields: "Title" with the text "Design team meeting", "Deadline" with the date "2021-02-28", "Start time" with "11:00 Am", "End time" with "14:00 Pm", "Remind" with "10 minutes early", and "Repeat" with "Weekly". Each field has a dropdown arrow. At the bottom is a green button labeled "Create a Task".

< Add task

Title

Design team meeting

Deadline

2021-02-28

Start time End time

11:00 Am 14:00 Pm

Remind

10 minutes early

Repeat

Weekly

Create a Task

## Rules

- Use React Native last stable version or Expo last stable release.
- Use some store management (Redux, Context for example)
- Implement typed data in create task ( PropTypes / Flow.js / Typescript )
- Implement navigation: Use React Navigation (<https://reactnavigation.org/>) V4 or V5.
- **Try to replicate the design as possible.**
- **If you have experience with testing, please add it. Is not mandatory.**
- **Please, don't create more features than we are asking in this document.**
- Create a Readme with clear instructions on how to run the application and specify the libraries that you used.
- Please don't use any backend, firebase, or api.
- The use of [StyledComponents](#) is recommended.



## **Delivery Steps**

1. Create a repo on GitHub and send to us the public link
2. The app must work in Android and iOS, so please send us a video demo showing the app working. If a good approach place the videos inside the repo.
3. Please give us 1-2 days for challenge review.
4. When the code challenge passes, we need another interview to show us the demo and explain the solution implemented.