

Name: Ahmad Hamdy

Sec: 3

React Native To-Do List App Report

To-Do List App Report

Overview

This React Native To-Do List App allows users to add, edit, and remove tasks. It features a clean UI with a custom font and a modern design, using icons for editing and deleting tasks.

Features

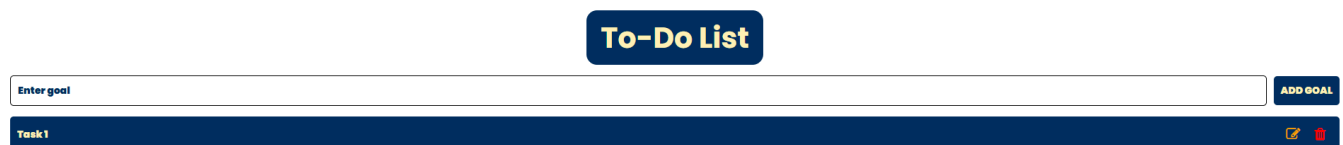
- Add new tasks
- Edit existing tasks in place
- Remove tasks from the list
- Uses custom fonts for a better visual experience
- Icons for editing and deleting tasks

Screenshots

A simple To-Do list



Add a new task or goal by typing your task then hit 'ADD GOAL' button



As you can see, you can edit the task or remove it.

Edit 'Task1' by hitting the edit icon,

The screenshot shows a 'To-Do List' application. At the top, there is a dark blue header with the title 'To-Do List' in yellow. Below the header is a white input field containing 'Task1' and a dark blue 'SAVE' button. Underneath the input field is a list of three tasks, each in a dark blue bar with white text and two small icons (edit and delete) on the right. The tasks are 'Task1', 'Task2', and 'Task3'. 'Task1' is currently selected.

Then edit its text and hit 'SAVE'

This screenshot shows the application after editing. The input field now contains the placeholder text 'Enter goal' and a dark blue 'ADD GOAL' button. The task list below shows 'Task1 (EDITED)', 'Task2', and 'Task3'. 'Task1 (EDITED)' is now the selected task.

And delete a task by hitting the 'delete' icon.

This screenshot shows the application after deleting a task. The input field still has 'Enter goal' and the 'ADD GOAL' button. The task list now only contains 'Task1 (EDITED)' and 'Task3'. 'Task1 (EDITED)' remains the selected task.

Colors and Fonts Used

Colors:

- Background color of title & buttons: #003161
- Text color: #FFF4B7
- Edit button color: #FFA500
- Delete button color: #FF0000
- Input border color: #003161

Fonts:

- Poppins Black (Used for list items, input text, and buttons)
- Poppins Bold (Used for the title)

Code Breakdown

Main Components:

App.js

This is the main file where the logic and UI of the To-Do list reside.

Adding & Editing a Goal

```
const addOrEditGoalHandler = () => {
  if (goalText.trim() === "") return;
  if (editGoalId) {
    setGoalList((prevGoals) =>
      prevGoals.map((goal) =>
        goal.id === editGoalId ? { ...goal, text: goalText } : goal
      )
    );
    setEditGoalId(null);
  } else {
    setGoalList([ ...goalList, { id: Math.random().toString(), text: goalText
    }]);
  }
  setGoalText("");
};
```

Removing a Goal

```
const removeGoalHandler = (goalId) => {
  setGoalList(goalList.filter((goal) => goal.id !== goalId));
};
```

Editing a Goal

```
const editGoalHandler = (goalId) => {
  const goalToEdit = goalList.find((goal) => goal.id === goalId);
  if (goalToEdit) {
    setGoalText(goalToEdit.text);
    setEditGoalId(goalId);
  }
};
```

Styling

The styling of the app is handled using `StyleSheet.create` in React Native. Below is a breakdown of the main styles:

Title Styling

```
title: {
  fontSize: 41,
  backgroundColor: "#003161",
  color: "#FFF4B7",
  textAlign: "center",
  paddingVertical: 8,
  paddingHorizontal: 20,
  borderRadius: 15,
  alignSelf: "center",
  marginBottom: 15,
  fontFamily: 'PoppinsBold',
},
```

Buttons and Inputs

```
addButton: {
  backgroundColor: "#003161",
  padding: 10,
  borderRadius: 5,
  alignItems: "center",
  justifyContent: "center",
},
iconButton: {
  padding: 5,
  marginHorizontal: 5,
},
```

Installation and Running the App

1. Clone the repository:

```
git clone https://github.com/Ahmad9h/ToDoList_ReactNative.git
```

2. Install dependencies:

```
npm install
```

3. Run the app using Expo:

```
expo start
```

Conclusion

This app provides a simple and effective way to manage tasks with an easy-to-use interface and a visually appealing design.