

## To-Do List Application - React Native Report

### Overview:

This project is a simple **To-Do List** mobile application built using **React Native** and **Expo**. The application allows users to add tasks, view them in a list, and delete tasks when completed by simply tapping on them.

### Key Features:

#### 1. Task Addition:

- Users can input a task through a text input field and add it to the task list by pressing the "Add Task" button.
- Input validation ensures that an empty task cannot be added.

#### 2. Task Display:

- Tasks are displayed in a scrollable list using **FlatList**. Each task is styled for clarity and ease of interaction.

#### 3. Task Deletion:

- Users can delete a task by tapping on it, which triggers the removal of the selected task from the list.

### React Hooks:

#### • **useState Hook:**

- The useState hook is used to manage two states:
  1. task: stores the current task input.
  2. tasks: stores the list of tasks with each task represented as an object ({ id, value }).

### Styling:

- The app uses **StyleSheet** for defining custom styles. Key elements like the container, input, task items, and buttons are styled for a clean and modern UI.
  - **Color Theme:** The primary color #6200ee is used for buttons and headers.
  - **Task Items:** Each task item has rounded corners, shadows, and padding to give a smooth, card-like appearance.

## **Components:**

### **1. TextInput:**

- Allows the user to enter new tasks. The onChangeText handler updates the task state.

### **2. Button:**

- Adds the task to the list when pressed, with the addTask function ensuring tasks are only added if non-empty.

### **3. FlatList:**

- Displays tasks dynamically. Each task is rendered with a TouchableOpacity that enables users to delete a task when tapped.

### **4. TouchableOpacity:**

- Wraps each task item to handle deletion through a touch event.

## **Improvements:**

- The app can be enhanced by adding features like:
  - Task editing.
  - Marking tasks as completed.
  - Task reordering or priority settings.

## **Conclusion:**

This basic To-Do List app demonstrates the use of essential React Native components, hooks, and styling techniques, providing a foundation for building more complex task management apps.