**Name:** Youssef Mohammed AbdelFattah Ahmed

**Section:** 4

**Date:** 12 / 3 / 2025

**React Native Task 1:** List-To-Do

**Screenshots of Design**

A screenshot of a phone

AI-generated content may be incorrect.A screenshot of a task list

AI-generated content may be incorrect.A screenshot of a task list

AI-generated content may be incorrect.A white surface with a black and white background

AI-generated content may be incorrect.A screenshot of a phone

AI-generated content may be incorrect.

**Colors and Fonts Used**

**Colors:**

* **Background Color:** #E8EAED (Light Gray)
* **Task Background:** #FFF (White)
* **Border Color:** #C0C0C0 (Gray)
* **Button Background:** #FFF (White)
* **Task Indicator:** #55BCF6 (Blue)

**Fonts:**

* **Section Title:** Font Size 24, Bold
* **Task Text:** Default Font, Max Width 80%
* **Add Button Text:** Font Size 24

**Code Used**

**App.js**

import React, {useState} from 'react';

import { Platform , KeyboardAvoidingView, StyleSheet, Text, View, TextInput, TouchableOpacity, Keyboard, ScrollView } from 'react-native';

import Task from './components/Task';

export default function App() {

  const [task, setTask] = useState('');

  const [taskItems, setTaskItems] = useState([]);

  const handleAddTask = () => {

  if (!task || task.trim().length === 0) return; // Ensure task is not empty or undefined

  Keyboard.dismiss();

  setTaskItems([...taskItems, task.trim()]); // Trim spaces before adding

  setTask('');

};

  const completeTask = (index) => {

    let itemsCopy = [...taskItems];

    itemsCopy.splice(index, 1);

    setTaskItems(itemsCopy)

  }

  return (

    <View style={styles.container}>

      {/\* Added scroll view to enable scrolling when list gets longer than the page \*/}

      <ScrollView

        contentContainerStyle={{

          flexGrow: 1

        }}

        keyboardShouldPersistTaps='handled'

      >

      {/\* Today's Tasks \*/}

      <View style={styles.tasksWrapper}>

        <Text style={styles.sectionTitle}>Today's tasks</Text>

        <View style={styles.items}>

          {

            taskItems.map((item, index) => {

              return (

                <TouchableOpacity key={index}  onPress={() => completeTask(index)}>

                  <Task text={item} />

                </TouchableOpacity>

              )

            })

          }

        </View>

      </View>

      </ScrollView>

      {/\* A keyboard avoiding view which ensures the keyboard does not cover the items on screen \*/}

      <KeyboardAvoidingView

        behavior={Platform.OS === "ios" ? "padding" : "height"}

        style={styles.writeTaskWrapper}

      >

        <TextInput style={styles.input} placeholder={'Write a task'} value={task} onChangeText={text => setTask(text)} />

        <TouchableOpacity onPress={() => handleAddTask()}>

          <View style={styles.addWrapper}>

            <Text style={styles.addText}>+</Text>

          </View>

        </TouchableOpacity>

      </KeyboardAvoidingView>

    </View>

  );

}

const styles = StyleSheet.create({

  container: {

    flex: 1,

    backgroundColor: '#E8EAED',

  },

  tasksWrapper: {

    paddingTop: 80,

    paddingHorizontal: 20,

  },

  sectionTitle: {

    fontSize: 24,

    fontWeight: 'bold'

  },

  items: {

    marginTop: 30,

  },

  writeTaskWrapper: {

    position: 'absolute',

    bottom: 60,

    width: '100%',

    flexDirection: 'row',

    justifyContent: 'space-around',

    alignItems: 'center',

  },

  input: {

    paddingVertical: 15,

    paddingHorizontal: 15,

    backgroundColor: '#FFF',

    borderRadius: 60,

    borderColor: '#C0C0C0',

    borderWidth: 1,

    width: 250,

  },

  addWrapper: {

    width: 60,

    height: 60,

    backgroundColor: '#FFF',

    borderRadius: 60,

    justifyContent: 'center',

    alignItems: 'center',

    borderColor: '#C0C0C0',

    borderWidth: 1,

  },

  addText: {

    fontSize: 24

  }

});

**Task.js**

import React from 'react';

import { View, Text, StyleSheet } from 'react-native';

const Task = (props) => {

    return (

    <View style={styles.item}>

        <View style={styles.itemLeft}>

            <View style={styles.square}></View>

                <Text style={styles.itemText}>{props.text}</Text>

            </View>

        <View style={styles.circular}></View>

    </View>

    )

}

const styles = StyleSheet.create({

    item: {

        backgroundColor: '#FFF',

        padding: 15,

        borderRadius: 10,

        flexDirection: 'row',

        alignItems: 'center',

        justifyContent: 'space-between',

        marginBottom: 20,

    },

        itemLeft: {

        flexDirection: 'row',

        alignItems: 'center',

        flexWrap: 'wrap',

    },

    square: {

        width: 24,

        height: 24,

        backgroundColor: '#55BCF6',

        opacity: 0.4,

        borderRadius: 5,

        marginRight: 15,

    },

    itemText: {

        maxWidth: '80%',

    },

    circular: {

        width: 12,

        height: 12,

        borderColor: '#55BCF6',

        borderWidth: 2,

        borderRadius: 5,

    },

});

export default Task;