Name: Abdelrahman Helal
Sec: 2

ToDo App

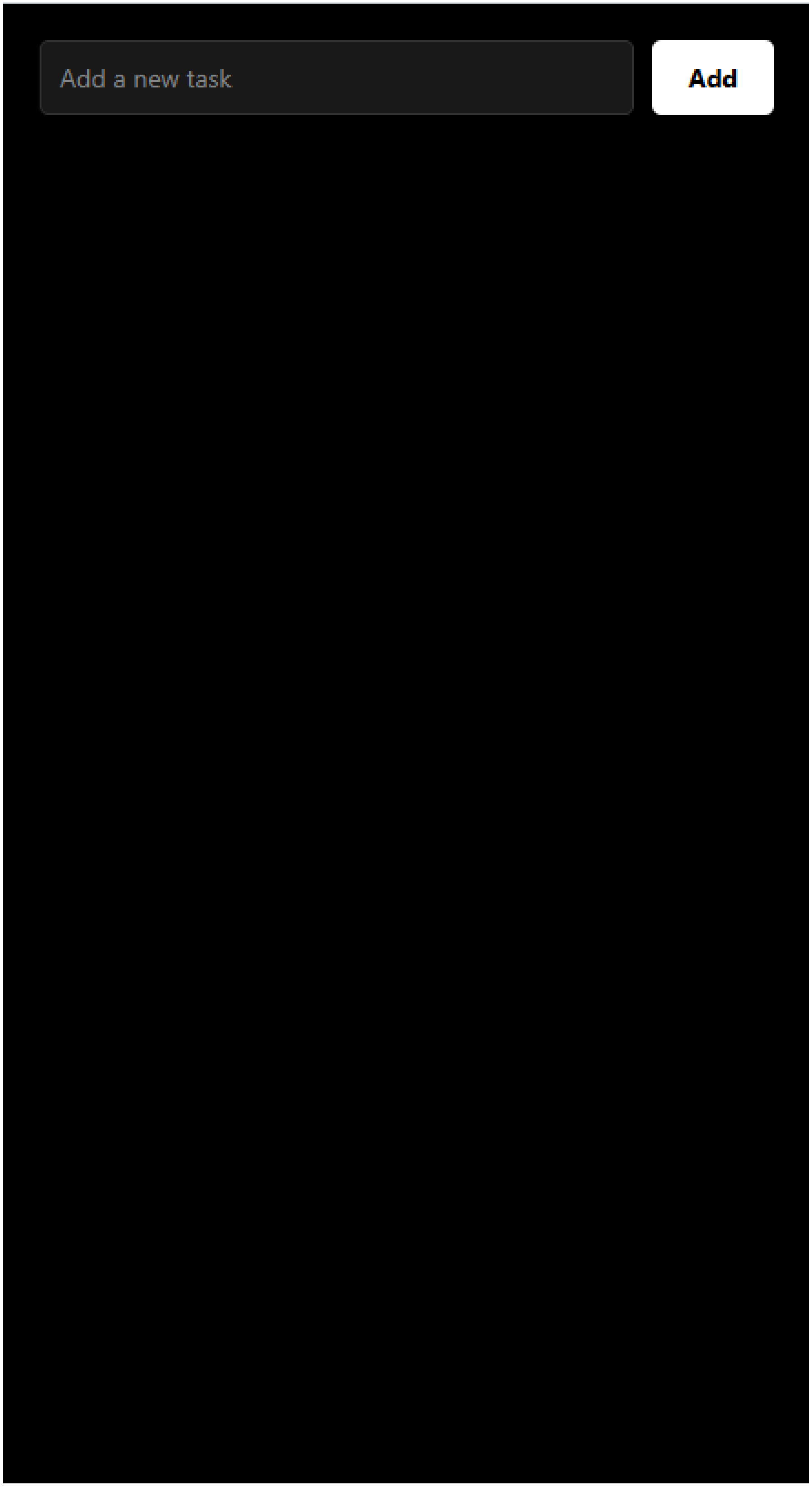
1. Introduction

This report provides an overview of the React Native Todo App, including screenshots demonstrating different actions and results, design choices (colors and fonts), and a breakdown of the code used for creating components, styling, and logic.

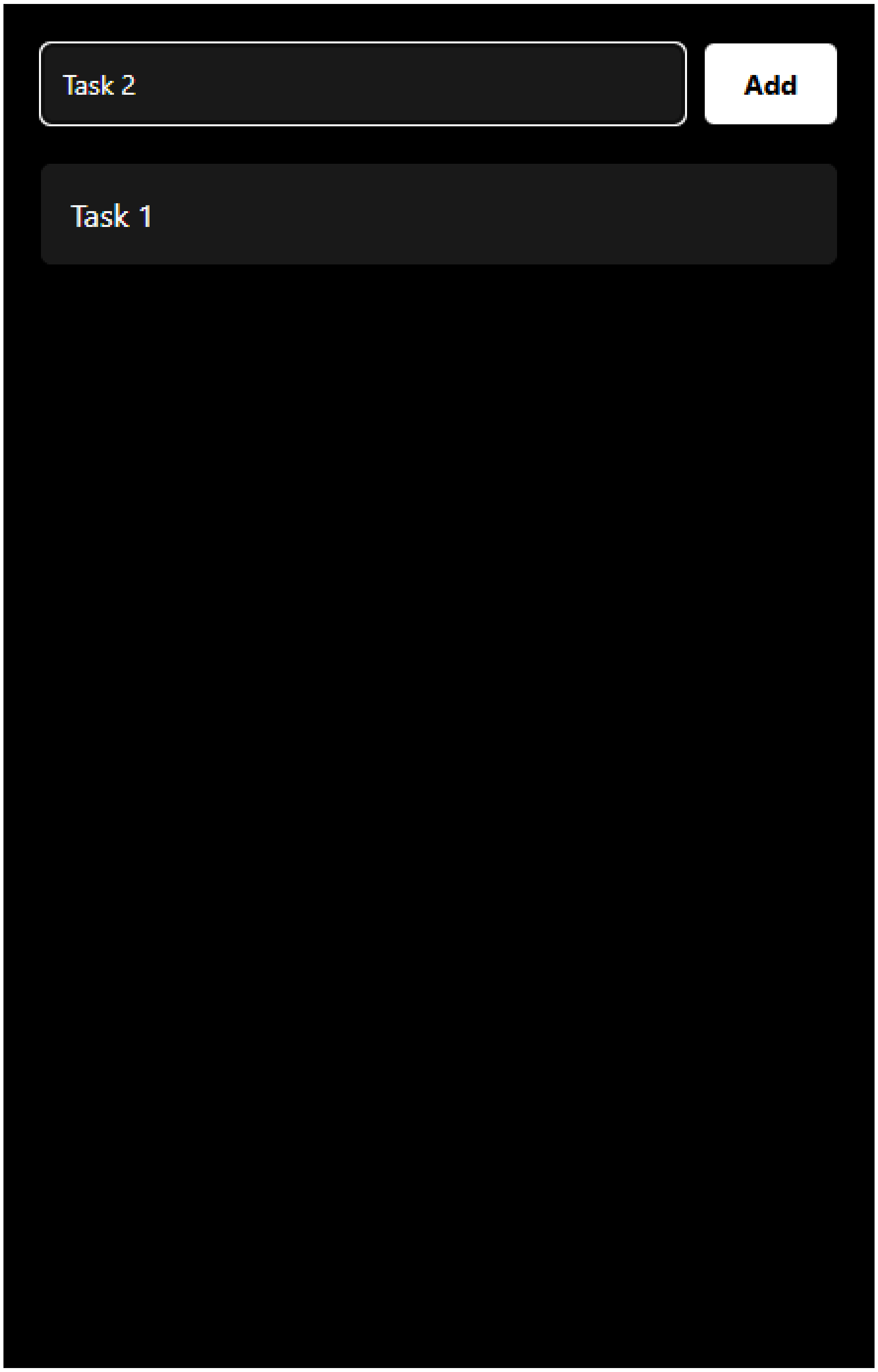
---

2. Screenshots

1. Home Screen (Initial State)



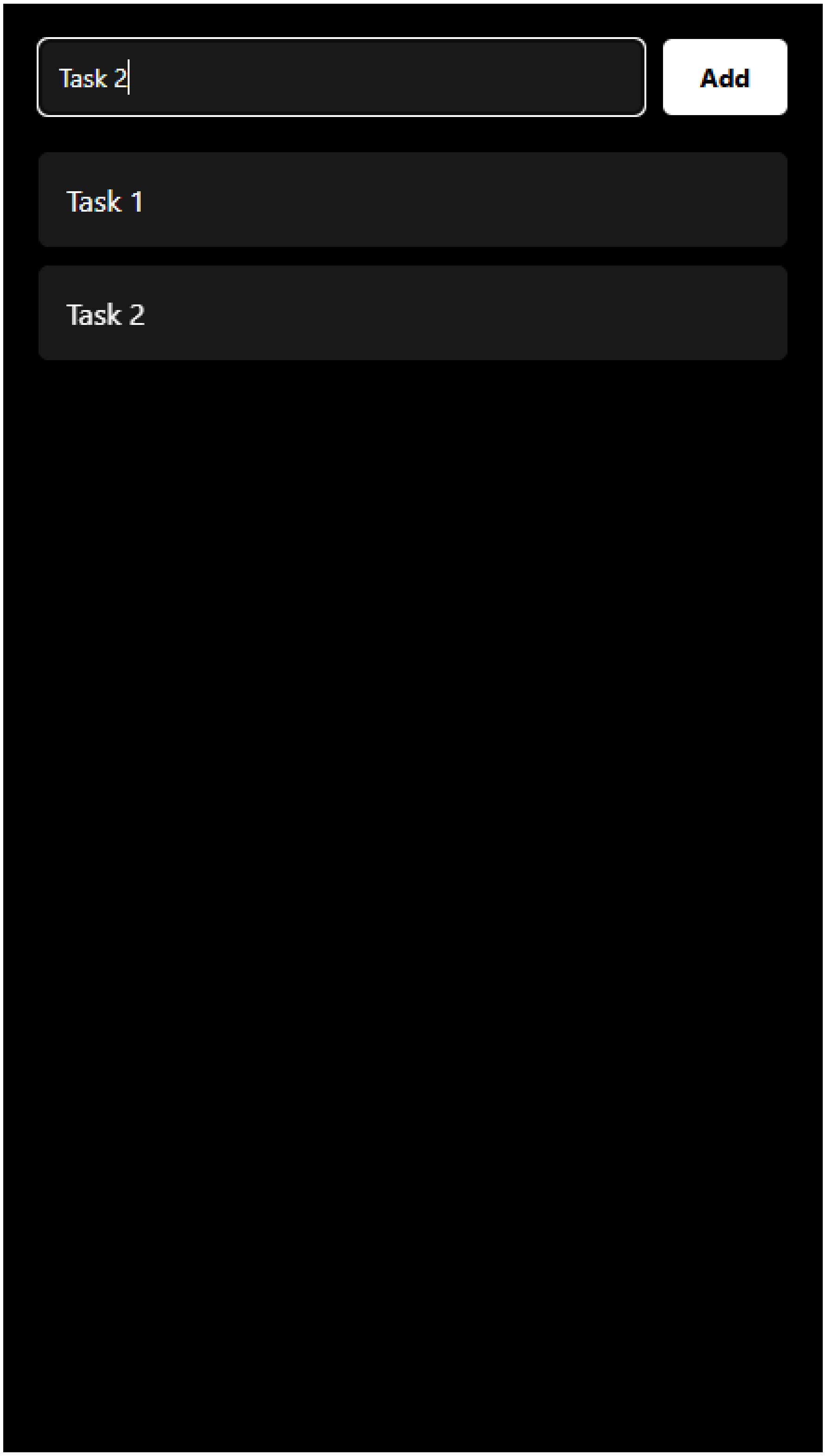
2. Adding a Task



Task 2

Task 1

3. Task Added to the List



Task 1

Task 2

3. Colors and Fonts

Color Palette:

- **Background:** `#000000` (Black)
- **Input Field:** `#1a1a1a` (Dark Gray)
- **Button Background:** `#ffffff` (White)

- **Button Text:** `#000000` (Black)
- **Task Text:** `#ffffff` (White)

Fonts Used:

- Default system font provided by React Native.

---

4. Code Breakdown

Component: App.js

```
import React, { useState } from 'react';
import {
  StyleSheet,
  View,
  TextInput,
  FlatList,
  Text,
  TouchableOpacity
} from 'react-native';

export default function App() {
  const [todos, setTodos] = useState([]);
  const [input, setInput] = useState('');

  const addTodo = () => {
    if (input.trim()) {
      setTodos([...todos, { text: input.trim() }]);
      setInput('');
    }
  };

  return (
    <View style={styles.container}>
      <View style={styles.inputContainer}>
        <TextInput
          style={styles.input}
          onChangeText={setInput}
          value={input}
          placeholder="Add a new task"
          placeholderTextColor="#888"
        />
        <TouchableOpacity
          style={styles.addButton}
          onPress={addTodo}
        >
          <Text style={styles.addButtonText}>Add</Text>
        </TouchableOpacity>
      </View>

      <FlatList
        data={todos}
        contentContainerStyle={styles.listContent}
        renderItem={({ item }) => (
          <View style={styles.todoItem}>
            <Text style={styles.todoText}>{item.text}</Text>
          </View>
        )}
      />
    </View>
  );
}
```

Styling: styles.js

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    padding: 20,
    backgroundColor: '#000000',
  },
  inputContainer: {
    flexDirection: 'row',
    marginBottom: 20,
    gap: 10,
  },
  input: {
    flex: 1,
    borderColor: '#333',
    borderWidth: 1,
    borderRadius: 5,
    padding: 10,
    color: '#fff',
    backgroundColor: '#1a1a1a',
  },
  addButton: {
    backgroundColor: '#fff',
    borderRadius: 5,
    paddingVertical: 10,
    paddingHorizontal: 20,
    justifyContent: 'center',
  },
  addButtonText: {
    color: '#000',
    fontWeight: 'bold',
  },
  todoItem: {
    padding: 15,
    marginBottom: 10,
    backgroundColor: '#1a1a1a',
    borderRadius: 5,
  },
  todoText: {
    fontSize: 16,
    color: '#fff',
  },
  listContent: {
    flexGrow: 1,
  },
});
```

5. Conclusion

This React Native Todo App provides a minimalistic and user-friendly interface for managing tasks. The report covered UI screenshots, design elements, and the source code used in the project. Future enhancements could include features such as task deletion, persistence using AsyncStorage, and task completion indicators.