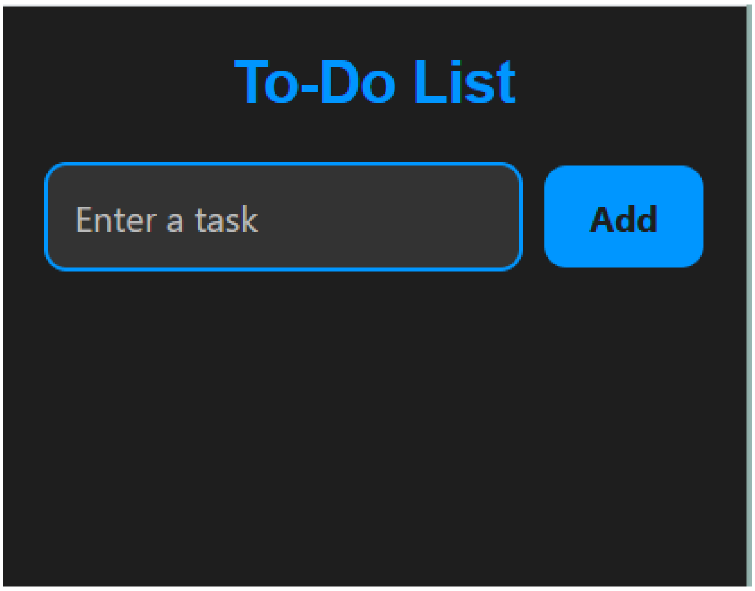
To-Do List App Report

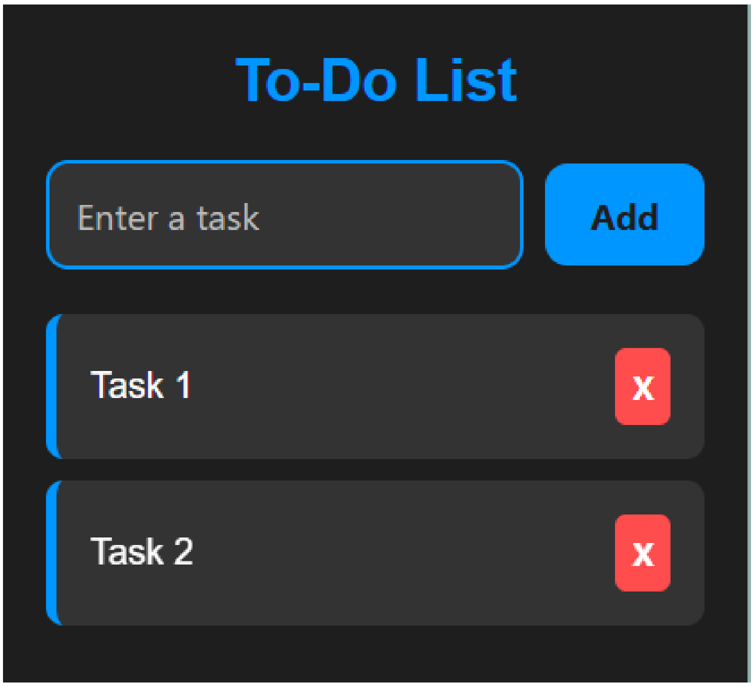
1. Screenshots

Below are screenshots for different stages of the app:

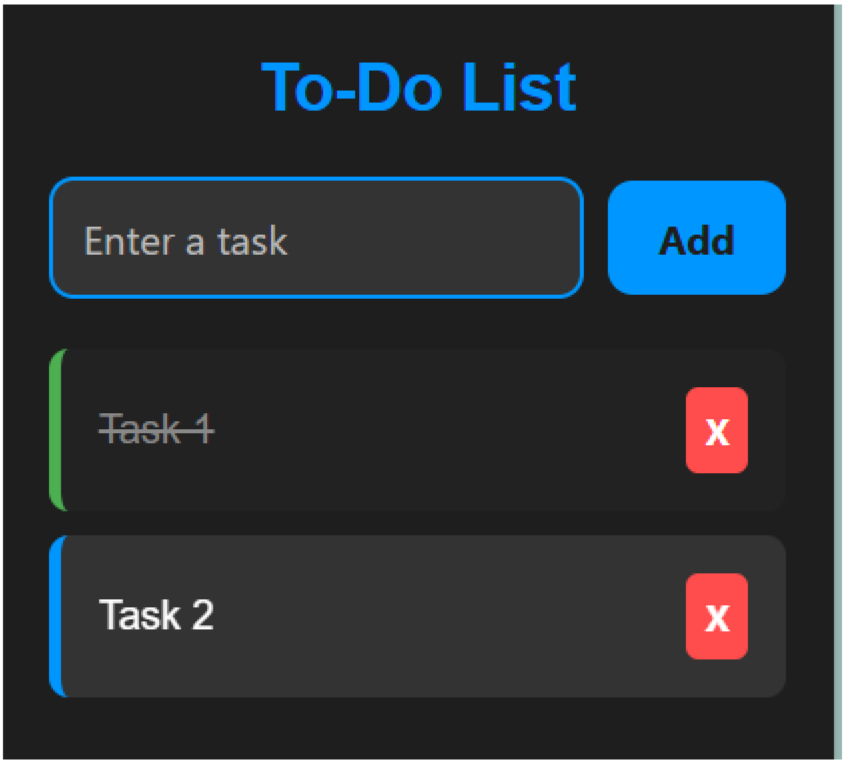
Starting Page



After Adding a Task



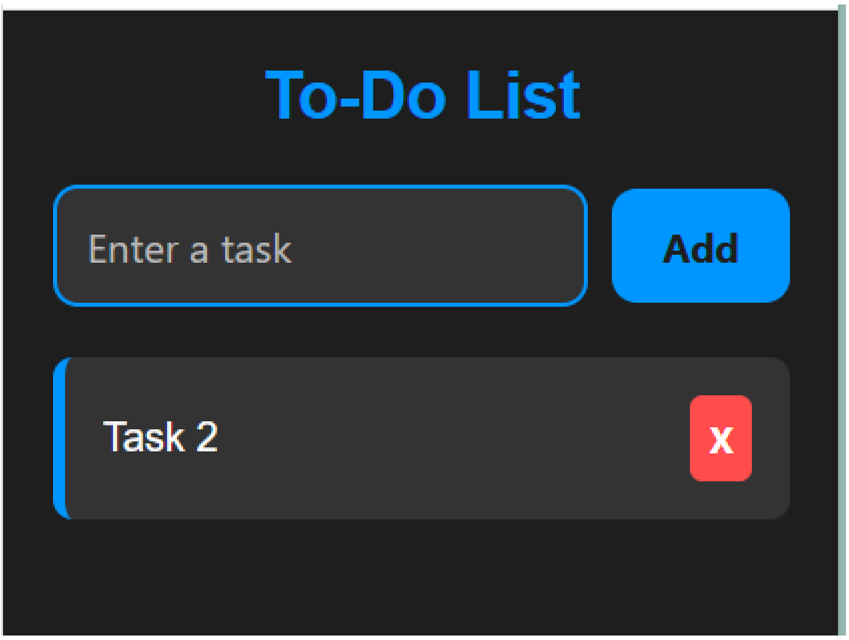
After Completing a Task


To-Do List

~~Task 1~~

Task 2

After Deleting a Task


To-Do List

Task 2

2. Colors and Fonts Used

Colors

Element	Color Code
Background	`#1E1E1E`
Input Field	`#333333`
Button	`#0096FF`
Completed Task	`#222222`
Delete Button	`#FF4C4C`

Fonts

- **Font Used:** **ARIAL**
- **Weights:**
  - **BOLD** for Titles and Buttons
  - **REGULAR** for Task Items
- **Font Sizes:**
  - Title: **26PX**
  - Task Text: **16PX**
  - Button Text: **16PX**

3. Code

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

export default function App() {
  const [goal, setGoal] = useState('');
  const [goals, setGoals] = useState([]);

  const addGoal = () => {
    if (goal.trim()) {
      setGoals([...goals, { id: Date.now().toString(), text: goal, completed: false }]);
      setGoal('');
    }
  };

  const toggleCompletion = (id) => {
    setGoals(goals.map(goal => goal.id === id ? { ...goal, completed: !goal.completed } : goal));
  };

  const deleteGoal = (id) => {
    setGoals(goals.filter(goal => goal.id !== id));
  };

  return (
    <View style={styles.container}>
      <Text style={styles.title}>To-Do List</Text>
      <View style={styles.inputContainer}>
        <TextInput
          style={styles.input}
          placeholder="Enter a task"
          placeholderTextColor="#bbb"
          value={goal}
          onChangeText={setGoal}
        />
        <TouchableOpacity style={styles.button} onPress={addGoal}>
          <Text style={styles.buttonText}>Add</Text>
        </TouchableOpacity>
      </View>
      <FlatList
        data={goals}
        keyExtractor={(item) => item.id}
        renderItem={({ item }) => (
          <View style={[styles.goalItem, item.completed && styles.completedGoal]}>
            <TouchableOpacity onPress={() => toggleCompletion(item.id)} style={styles.goalTextContainer}>
              <Text style={[styles.goalText, item.completed && styles.goalTextCompleted]}>
                {item.text}
              </Text>
            </TouchableOpacity>
            <TouchableOpacity onPress={() => deleteGoal(item.id)}
style={styles.deleteButton}>
              <Text style={styles.deleteButtonText}>X</Text>
            </TouchableOpacity>
          </View>
        )}
      />
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#1E1E1E',
    padding: 20,
    justifyContent: 'center',
  },
  title: {
    fontSize: 26,
    fontWeight: 'bold',
    color: '#0096FF',
    textAlign: 'center',
    marginBottom: 20,
    fontFamily: 'Arial',
  },
  inputContainer: {
    flexDirection: 'row',
    alignItems: 'center',
    marginBottom: 20,
  },
  input: {
    flex: 1,
    borderWidth: 2,
    borderColor: '#0096FF',
    padding: 12,
    borderRadius: 10,
    backgroundColor: '#333',
    fontSize: 16,
    color: '#FFF',
  },
  button: {
    backgroundColor: '#0096FF',
    paddingVertical: 12,
    paddingHorizontal: 20,
    borderRadius: 10,
    marginLeft: 10,
  },
  buttonText: {
    fontSize: 16,
    fontWeight: 'bold',
    color: '#1E1E1E',
  },
  goalItem: {
    flexDirection: 'row',
    alignItems: 'center',
    backgroundColor: '#333',
    padding: 15,
    borderRadius: 8,
    marginBottom: 10,
    borderLeftWidth: 5,
    borderLeftColor: '#0096FF',
  },
  completedGoal: {
    borderLeftColor: '#4CAF50',
    backgroundColor: '#222',
  },
  goalTextContainer: {
    flex: 1,
  },
  goalText: {
    fontSize: 16,
    color: '#FFF',
    fontFamily: 'Arial',
  },
  goalTextCompleted: {
    textDecorationLine: 'line-through',
    color: '#888',
  },
  deleteButton: {
    backgroundColor: '#FF4C4C',
    padding: 8,
    borderRadius: 5,
    marginLeft: 10,
  },
  deleteButtonText: {
    color: '#FFF',
    fontWeight: 'bold',
  },
});
```