| **Software Engineering Department - ITU** |
| --- |
| **MD442: Mobile Application and Development** |

| **Course Instructor: Usama Bin Shakeel** | **Dated: 4/02/2025** |
| --- | --- |
| **Teaching Assistant: Hammad Kamran** | **Semester: Spring 2025** |
| **Lab Engineer: Hateem Hassan** | **Batch: BSCE2021& BSEE2021** |

# **Lab 2. Task Management App**

| **Name** | **Roll number** | **Report**  **(out of 35)** |
| --- | --- | --- |
|  |  |  |

Checked on: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## **Objective**

## The objective of this lab assignment is to develop a simple task management app using React Native. This project will help students:

## Learn to manage user input and state in a mobile app.

## Implement CRUD operations (Create, Read, Update, Delete) for tasks.

## Design a clean, responsive user interface.

## Use navigation to transition between different screens.

## **Equipment and Component**

| **Component Description** | **Value** | **Quantity** |
| --- | --- | --- |
| Computer | Available in lab | 1 |

## **Conduct of Lab**

1. Students are required to perform this experiment individually.
2. In case the lab experiment is not understood, the students are advised to seek help from the course instructor, lab engineers, assigned teaching assistants (TA) and lab attendants.

## **Components Required**

* **FlatList** - Renders the list of tasks dynamically.  
    
   <FlatList

data={tasks}

keyExtractor={(item) => item.id}

renderItem={({ item }) => <Text>{item.title}</Text>}

/>

* **Stack Navigation** - Enables navigation between task list and task details screens.  
    
   const Stack = createStackNavigator();

function AppNavigator() {

return (

<Stack.Navigator>

<Stack.Screen name="Home" component={TaskListScreen} />

<Stack.Screen name="Details" component={TaskDetailsScreen} />

</Stack.Navigator>

);

}

### Lab Tasks

#### **Task 1: User Interface**

**Layout:**

* Include a task list screen to display tasks.
* Add forms for adding and editing tasks.

**Task List Screen:**

* Display tasks in a **FlatList** with their title and status.
* Use different colors or badges for task statuses:
  + **Todo:** Red
  + **In Progress:** Yellow
  + **Complete:** Green
* Allow tapping on a task to view its details.

**Task Form Screen:**

* Include input fields for:
  + Task Title (Required)
  + Description (Optional)
  + Status (Dropdown or segmented control: **Todo, In Progress, Complete**)
* Provide **Save** and **Cancel** buttons.

#### **Task 2: Basic Functionalities**

**Add Task:**

* Allow users to create a new task with a title, description, and status.
* Save the task and update the task list.

**Edit Task:**

* Allow users to modify the title, description, and status of an existing task.
* Update the task list after saving changes.

**Delete Task:**

* Provide a **delete** button on the task details screen.
* Show a confirmation alert before deleting a task.

#### **Task 3: Navigation & Task Details**

**Navigation:**

* Implement **React Navigation** to switch between:
  + **Task List Screen** (Home)
  + **Task Details Screen**
  + **Task Form Screen** (Add/Edit)

**Task Details View:**

* Display full task details:
  + Title
  + Description
  + Current Status
  + Creation Date/Time (Optional)
* Provide an **Edit** button to modify the task.

Assessment Rubric for Lab

| **Performance metric** | **CLO** | **Able to complete the task over 80% (4-5)** | **Able to complete the task 50-80% (2-3)** | **Able to complete the task below 50% (0-1)** | **Marks** |
| --- | --- | --- | --- | --- | --- |
| 1. Realization of experiment | 1 | Executes without errors excellent user prompts, good use of symbols, spacing in output. The testing has been completed. | Executes without errors, user prompts are understandable,minimum use of symbols or spacing in output. Some testing has been completed. | Does not execute due to syntax errors, runtime errors, user prompts are misleading or non- existent. No testing has been completed. |  |
| 2. Conducting experiment | 1 | Able to make changes and answer all questions. | Partially able to make changes and few incorrect answers. | Unable to make changes and answer all questions. |  |
| 3. Computer use | 2 | Document submission timely. | Document submission late. | Document submission not done. |  |
| 4. Teamwork | 3 | Actively engages and cooperates with other group member(s) in an effective manner. | Cooperates with other group member(s) in a reasonable manner but conduct can be improved. | Distracts or discourages other group members from conducting the experiment |  |
| 5. Laboratory safety and disciplinary rules | 3 | Code comments are added and do help the reader to understand the code. | Code comments are added and do not help the reader to understand the code. | Code comments are not added. |  |
| 6. Data collection | 3 | Excellent use of white space, creatively organized work, excellent use of variables and constants, correct identifiers for constants, No line-wrap. | Includes name, and assignment, white space makes the program fairly easy to read. Title, organized work, good use of variables. | Poor use of white space (indentation, blank lines) making code hard to read, disorganized and messy. |  |
| 7. Data analysis | 4 | Solution is efficient, easy to understand, and maintain. | A logical solution that is easy to follow but it is not the most efficient. | A difficult and inefficient solution. |  |
| **Total (out of 35):** | | | | |  |