CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY
FACULTY OF TECHNOLOGY AND ENGINEERING
CHANDUBHAI S. PATEL INSTITUTE OF TECHNOLOGY
DEPARTMENT OF ARTIFICIAL INTELLIGENCE & MACHINE LEARNING

Subject : Mobile Application Development **Semester:** 5

Subject Code: AIML308 **Academic Year :**2025-26(ODD)

NAME : CHOKSI OM CHIRAGBHAI **ID: 23AIML010**

Practical 3

Problem Definition

You are building a mobile application where users navigate through multiple screens like login, dashboard, and profile. Create a dynamic TODO app using State Management (setState) and ListView.builder.

Supplementary Problems -	Add categories, filter completed tasks

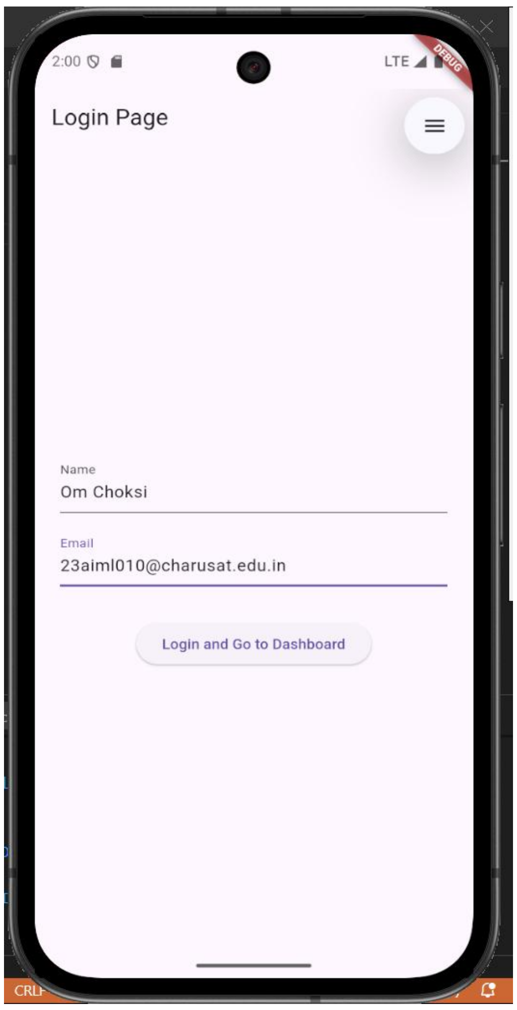
Technical Approach:

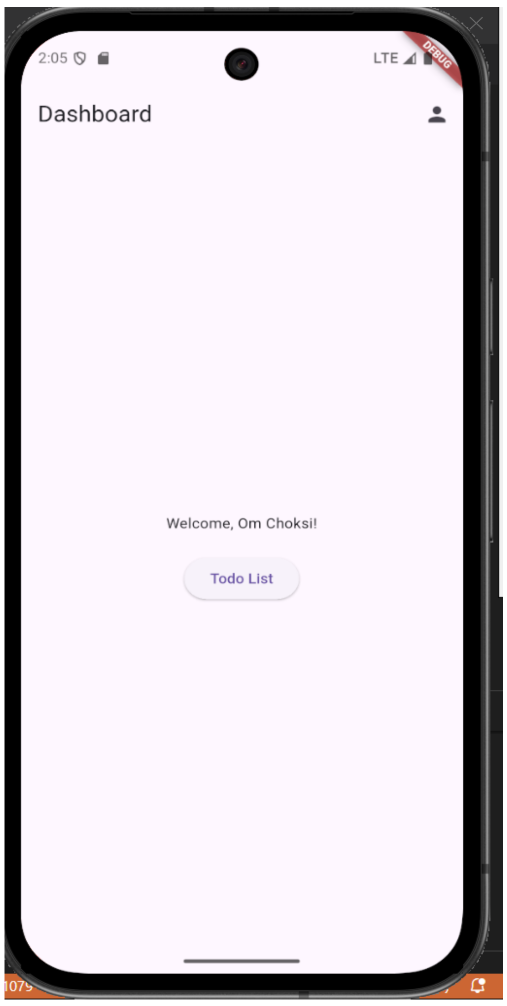
This Flutter app implements a dynamic TODO list using setState for state management within StatefulWidget to handle UI updates efficiently. Core logic relies on Dart functions for task operations like adding, deleting, and filtering, with ListView.builder for rendering dynamic lists. Key widgets include TextField for input, DropdownButton for categories, Checkbox for completion, and Switch for filtering, ensuring responsive and interactive UI without external packages beyond Flutter's material library.

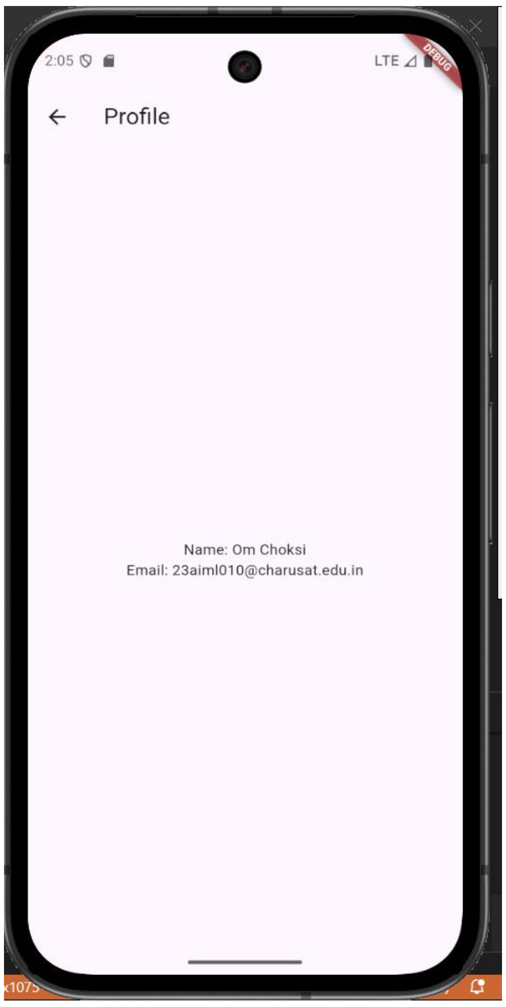
File Structure:

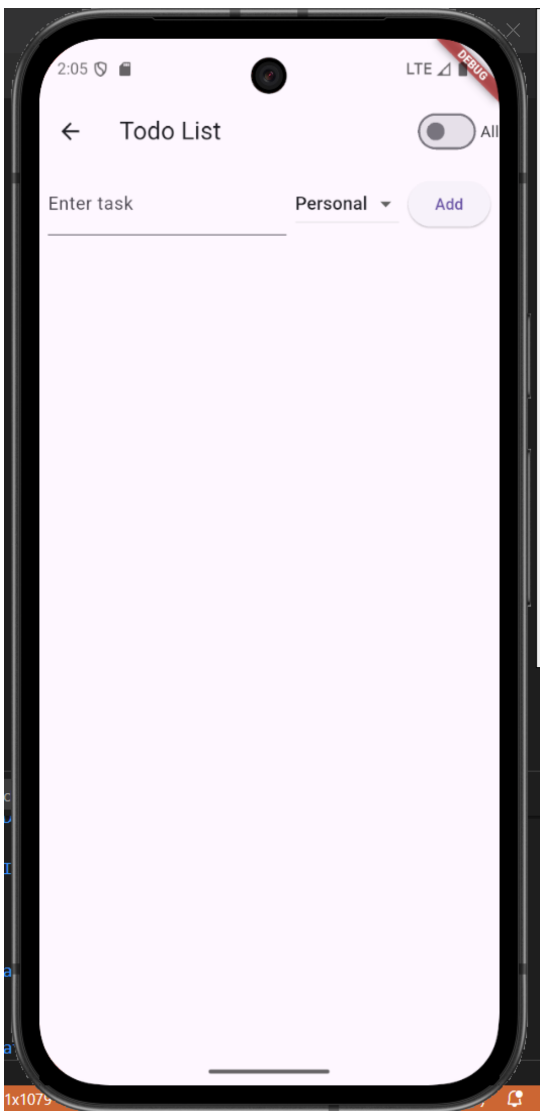
```
lib/
├──── main.dart        # App entry point with route configuration
├──── login.dart       # Login screen with name and email input
├──── dashboard.dart    # Dashboard with user greeting and navigation
├──── profile.dart      # Profile screen displaying user details
└──── todolist.dart     # TODO list screen with dynamic task management
```

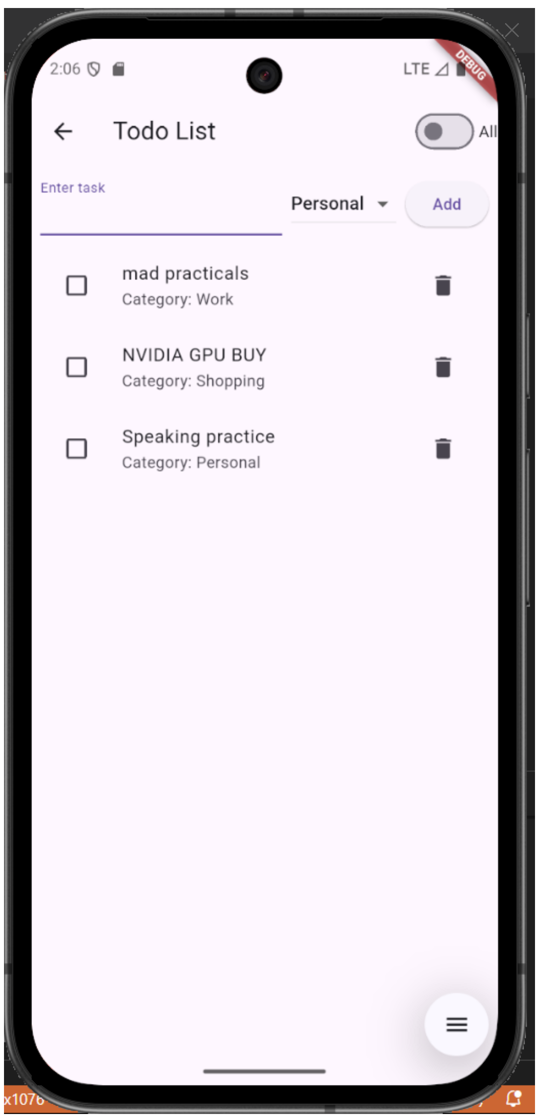
Screenshots:

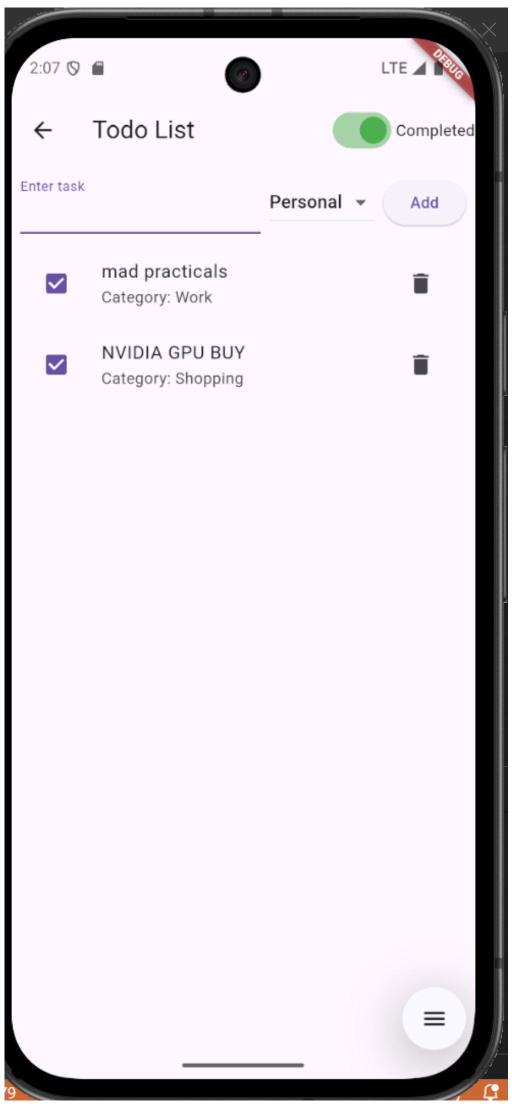


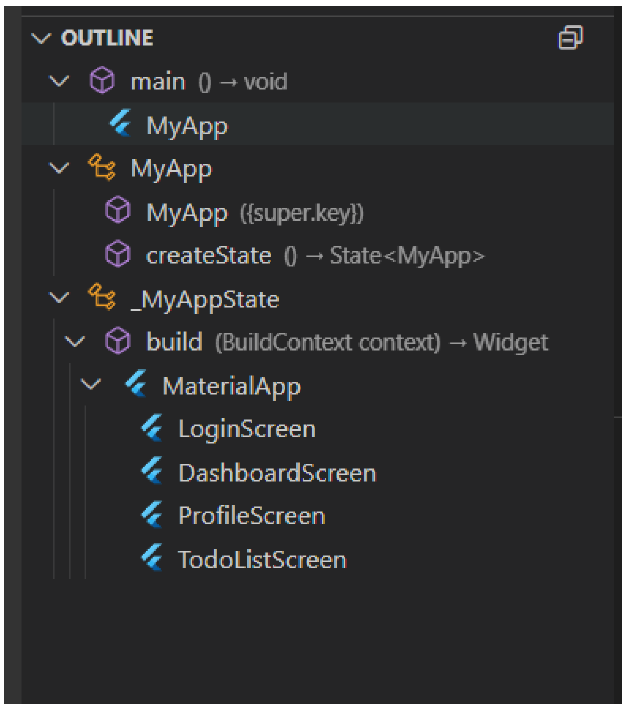




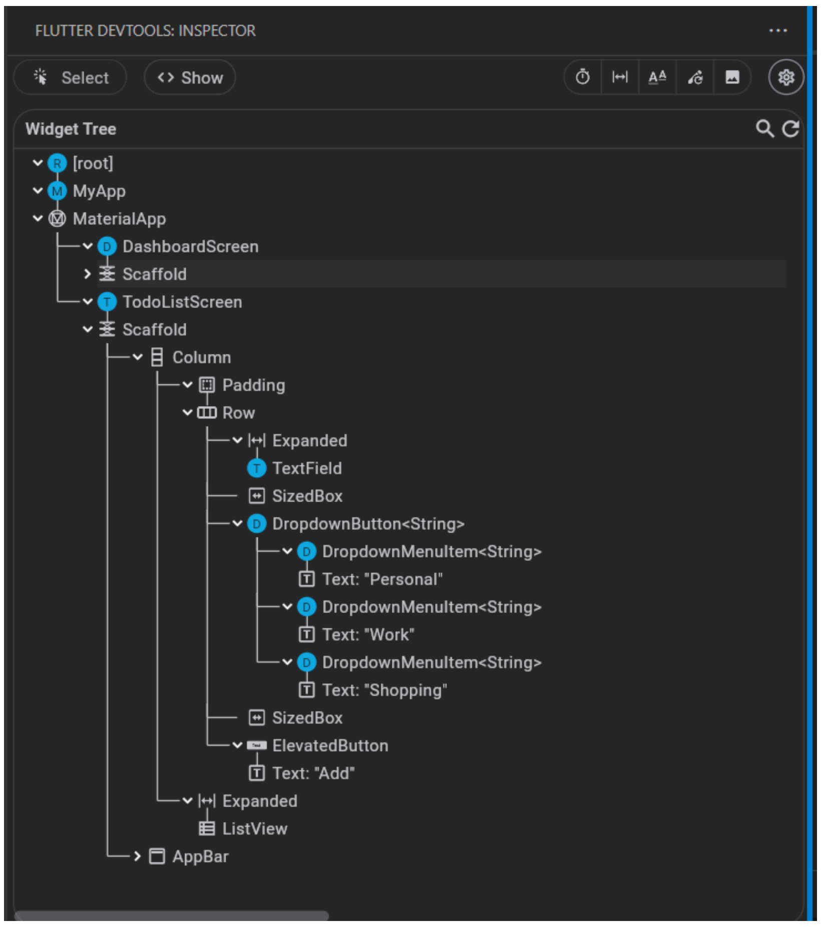








```
OUTLINE
  main () → void
    MyApp
  MyApp
    MyApp ({super.key})
    createState () → State<MyApp>
  _MyAppState
    build (BuildContext context) → Widget
      MaterialApp
        LoginScreen
        DashboardScreen
        ProfileScreen
        TodoListScreen
```



```
[root]
MyApp
MaterialApp
├─ DashboardScreen
│  Scaffold
└─ TodoListScreen
   Scaffold
   ├─ Column
   │  ├─ Padding
   │  │  Row
   │  │  ├─ Expanded
   │  │  │  TextField
   │  │  ├─ SizedBox
   │  │  ├─ DropdownButton<String>
   │  │  │  ├─ DropdownMenuItem<String>
   │  │  │  │  Text: "Personal"
   │  │  │  ├─ DropdownMenuItem<String>
   │  │  │  │  Text: "Work"
   │  │  │  └─ DropdownMenuItem<String>
   │  │  │     Text: "Shopping"
   │  │  ├─ SizedBox
   │  │  └─ ElevatedButton
   │  │     Text: "Add"
   │  └─ Expanded
   │     ListView
   └─ AppBar
```

Key Questions:

1. How to take and validate user input?
Ans: Items are added using tasks.add() in _addTask function with setState to update UI; deleted using tasks.removeAt(index) in _deleteTask, triggering rebuild.

2. What is the role of Dart functions?
Ans: setState marks the widget as dirty, notifying Flutter to rebuild the widget tree with updated state, ensuring UI reflects changes like task additions or completions.

3. How to update UI on user action?
Ans: Use ListView.builder for efficient rendering of variable-length lists; update the underlying List<Task> with setState to reflect additions, deletions, or filters dynamically.

Key Skills to be Understand:

setState, ListView, Dynamic UI