Name:Bhushan Mukund Kor **Academic Year:2024-2025**

Division: D15C **Roll No: 28**

Experiment 2

Aim: To design Flutter UI by including common widgets.

Theory:
Flutter is an open-source UI toolkit developed by Google for building natively compiled applications across mobile, web, and desktop platforms using a single codebase. It is centered around the use of widgets, which are the foundational elements for constructing user interfaces. Each widget defines a part of the UI and how it should appear based on its configuration and internal state.

Types of Widgets in Flutter
Flutter widgets fall into two primary categories:
1. Stateless Widgets
   - These widgets do not retain any state information.
   - They are immutable, meaning their properties cannot change once built.
   - Typically used for UI elements that remain constant throughout the app.
   - Examples: Text, Icon, Container
2. Stateful Widgets
   - These widgets can hold and manage state dynamically.
   - They are mutable, allowing updates based on user interactions or other runtime events.
   - Ideal for components that need to reflect changes in real-time.
   - Examples: Checkbox, TextField, Slider

Common Flutter Widgets for UI Design

Widget	Description
Scaffold	Provides the basic structure for an app, including app bars, drawers, etc.
AppBar	Displays a toolbar at the top with titles and optional actions.
Text	Used to display simple or styled text.
Container	A versatile box widget used for layout, styling, and decoration.
Row & Column	Layout widgets used for arranging children horizontally or vertically.
ListView	A scrollable list of widgets arranged linearly.
TextField	An input field that accepts user text and interacts with the keyboard.
Buttons	Interactive widgets like ElevatedButton, TextButton, and IconButton.

Image	Displays images from local assets or the internet.
Card	A material design card with elevation and rounded corners for containing content.

Codes:

1. **Scaffold**

```
return Scaffold(
    appBar: AppBar(
     title: Align(
      alignment: Alignment.centerLeft,
      child: Text(
       AppLocalizations.of(context).app_name,
       style: const TextStyle(color: Colors.white),
      ),
     ),
     backgroundColor: Colors.blue,
     actions: [],
    ),
    body: _screens[_selectedIndex],
    bottomNavigationBar: Theme(
     data: Theme.of(context).copyWith(
      splashColor: Colors.transparent,
      highlightColor: Colors.transparent,
     ),
     child: BottomNavigationBar(
      type: BottomNavigationBarType.fixed,
      currentIndex: _selectedIndex,
      onTap: _onItemTapped,
      items: [
       BottomNavigationBarItem(
        icon: const Icon(Icons.home_outlined),
        activeIcon: const Icon(Icons.home, color: Colors.blue),
        label: AppLocalizations.of(context).home,
       ),
       BottomNavigationBarItem(
        icon: const Icon(Icons.group_outlined),
        activeIcon: const Icon(Icons.group, color: Colors.blue),
        label: AppLocalizations.of(context).groups,
       ),
       BottomNavigationBarItem(
        icon: const Icon(Icons.settings_outlined),
```

```
            activeIcon: const Icon(Icons.settings, color: Colors.blue),
            label: AppLocalizations.of(context).settings,
          ),
          BottomNavigationBarItem(
            icon: const Icon(Icons.person_outline),
            activeIcon: const Icon(Icons.person, color: Colors.blue),
            label: AppLocalizations.of(context).profile,
          ),
        ],
      ),
    ),
  );
```

2. **Row & Column**

```
child: Column(
        mainAxisSize: MainAxisSize.min,
        children: [
          const Text(
            "Select Targeted Time",
            style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
          ),
          const SizedBox(height: 10),
          Row(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              SizedBox(
                width: 80,
                child: DropdownButton<int>(
                  value: selectedHours,
                  items: List.generate(13, (index) => index)
                      .map((e) => DropdownMenuItem(value: e, child: Text("$e h")))
                      .toList(),
                  onChanged: (value) {
                    setState(() {
                      selectedHours = value!;
                    });
                  },
                ),
              ),
              const SizedBox(width: 20),
              SizedBox(
                width: 80,
                child: DropdownButton<int>(
```

```
                value: selectedMinutes,
                items: [0, 15, 30, 45]
                    .map((e) => DropdownMenuItem(value: e, child: Text("$e min")))
                    .toList(),
                onChanged: (value) {
                  setState(() {
                    selectedMinutes = value!;
                  });
                },
              ),
            ),
          ],
        ),
        const SizedBox(height: 20),
        ElevatedButton(
          onPressed: () {
            setState(() {
                              _targetedTime = Duration(hours: selectedHours, minutes:
selectedMinutes);
            });
            Navigator.pop(context);
          },
          child: const Text("Set Time"),
        ),
      ],
    ),
```

3. **Container**

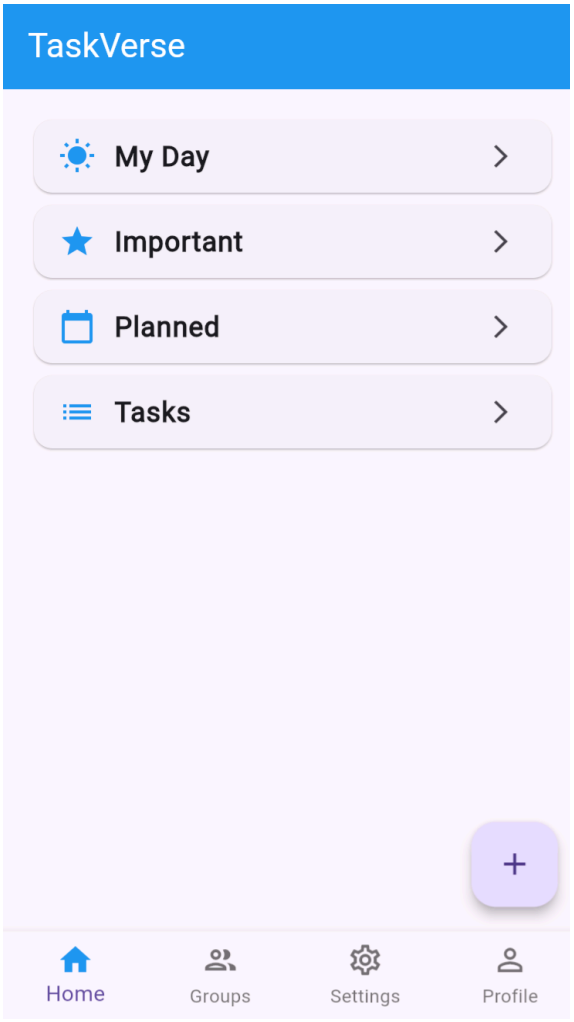
```
Container(
          width: 30,
          height: 30,
          decoration: BoxDecoration(
            shape: BoxShape.circle,
            border: Border.all(color: Colors.grey, width: 2),
            color: widget.task.isCompleted ? Colors.green : Colors.transparent,
          ),
          child: widget.task.isCompleted
              ? const Icon(Icons.check, color: Colors.white, size: 20)
              : null,
        ),
```

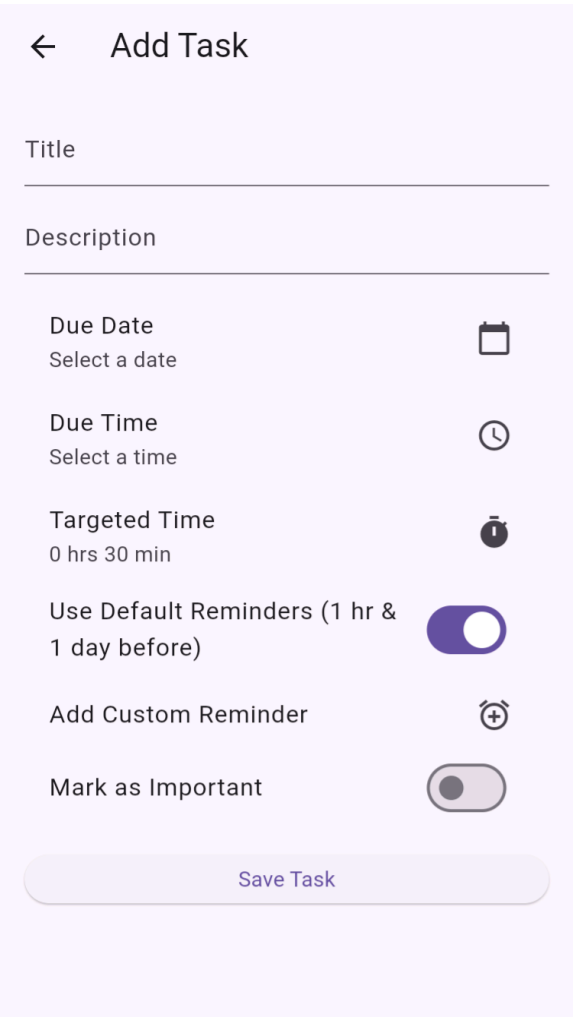
4. **ListView**

Name:Bhushan Mukund Kor **Academic Year:2024-2025**

Division: D15C **Roll No: 28**

```
ListView.builder(
            itemCount: importantTasks.length,
            itemBuilder: (context, index) {
              return TaskTile(
                task: importantTasks[index],
                  _firestoreService.deleteTask(importantTasks[index]); // Delete from Firestore
                },
                  importantTasks[index].isCompleted = true;
                  _firestoreService.updateTask(importantTasks[index]); // Update in Firestore
                },
                  importantTasks[index].isCompleted = false;
                  _firestoreService.updateTask(importantTasks[index]); // Update in Firestore
                },
              );
            },
          );
```





Name:Bhushan Mukund Kor **Academic Year:2024-2025**

Division: D15C **Roll No: 28**

GitHub Link: https://github.com/BKCODE2003/ToDo

Conclusion:

This experiment demonstrated the effective use of common Flutter widgets such as Scaffold, Row, Column, Container, and ListView to design a structured and interactive UI. By implementing these widgets, we learned how to create responsive layouts, manage navigation, and display dynamic content, reinforcing the importance of widgets as fundamental building blocks in Flutter app development. The practical application of these concepts helps in developing visually appealing and functional user interfaces efficiently.