

## MAD & PWA Lab Journal

Experiment No.	02
Experiment Title.	To design Flutter UI by including common widgets.
Roll No.	60
Name	Pranav Titambe
Class	D15A
Subject	MAD & PWA Lab
Lab Outcome	LO2: Design and Develop interactive Flutter App by using widgets, layouts, gestures and animation
Grade:	

**Aim:** To design Flutter UI by including common widgets.

## **Theory:**

Flutter follows a widget-based approach where everything in the UI is a widget. Widgets can be classified into two main types:

- Stateless Widgets: Do not change their state once built (e.g., Text, Container).
- Stateful Widgets: Can update dynamically based on user interaction (e.g., TextField, Checkbox).

### Commonly Used Widgets in Flutter-

#### (a) Scaffold Widget

The Scaffold widget provides the basic structure for a Flutter app, including an AppBar, Drawer, FloatingActionButton, and BottomNavigationBar. It is a fundamental widget used to create a standard screen layout in Flutter.

#### (b) Container Widget

A Container is a box model widget that can hold other widgets. It is commonly used for adding padding, margins, borders, and background decorations.

#### (c) Row and Column Widgets

- Row: Arranges widgets horizontally.
- Column: Arranges widgets vertically.

These two widgets are fundamental for designing layouts in Flutter.

#### (d) ListView Widget

The ListView widget is used for displaying a scrollable list of items. It is useful for showing large amounts of data dynamically.

#### (e) Stack Widget

The Stack widget is used to place widgets on top of each other. This is useful for creating overlapping UI elements such as banners, profile images, or layered designs.

#### (f) ElevatedButton Widget

The ElevatedButton widget is used for clickable buttons with a raised effect. It is a commonly used button in Flutter applications.

#### (g) TextField Widget

The TextField widget is used to take user input, such as entering a name, email, or password. It is commonly used in forms and authentication screens.

## Code:

### *main.dart file*

```
import 'package:flutter/material.dart';
import 'package:dailyhunt/login.dart';
import
'package:permission_handler/permission_handl
er.dart';
```

```
void main() {
  runApp(const MyApp());
}
```

```
class MyApp extends StatelessWidget {
  const MyApp({super.key});
```

```
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: "The Daily Globe",
      color: Color(0xFFE1FFBB),
      theme: ThemeData(
        primarySwatch: Colors.blue,
        scaffoldBackgroundColor: const
Color(0xFF001A6E),
        // textTheme: const TextTheme(
        //   bodyLarge: TextStyle(color:
Color(0xFFEFE9D5)),
        //   bodyMedium: TextStyle(color:
Color(0xFFEFE9D5)),
        //   titleLarge: TextStyle(color:
Color(0xFFEFE9D5)),
        // ),
        // textTheme: const TextTheme(
        //   titleLarge: TextStyle(
        //     color: Color(0xFFEFE9D5)
        //   )
        // ),
        home: const OnboardingFlow(),
        debugShowCheckedModeBanner: false,
      );
  }
}
```

```
class OnboardingFlow extends StatefulWidget {
  const OnboardingFlow({super.key});
```

```
  @override
  State<OnboardingFlow> createState() =>
_OnboardingFlowState();
}
```

```
class _OnboardingFlowState extends
State<OnboardingFlow> {
  final PageController _pageController =
PageController();
  int _currentPage = 0;
```

```
  @override
```

```
Widget build(BuildContext context) {
  return Scaffold(
    body: Column(
      children: [
        Expanded(
          child: PageView(
            controller: _pageController,
            onPageChanged: (index) {
              setState(() {
                _currentPage = index;
              });
            },
            children: const [
              SplashScreen(),
              TrackingScreen(),
              NotificationScreen(),
            ],
          ),
          // Page Navigation Buttons
          Padding(
            padding: const
EdgeInsets.symmetric(horizontal: 16, vertical:
20),
            child: Row(
              mainAxisAlignment:
MainAxisAlignment.spaceBetween,
              children: [
                if (_currentPage > 0)
                  TextButton(
                    onPressed: () {
                      _pageController.previousPage(
                        duration: const
Duration(milliseconds: 300),
                        curve: Curves.easeInOut,
                      );
                    },
                    child: const Text(
                      "Back",
                      style: TextStyle(color:
Color(0xFFE1FFBB)),
                    ),
                  ),
                ElevatedButton(
                  onPressed: () {
                    if (_currentPage < 2) {
                      _pageController.nextPage(
                        duration: const
Duration(milliseconds: 300),
                        curve: Curves.easeInOut,
                      );
                    } else {
                      // Navigate to login screen
                      Navigator.pushReplacement(
                        context,
                        MaterialPageRoute(builder:
(context) => const Login()),

```

```

    );
  }
},
style: ElevatedButton.styleFrom(
  foregroundColor:
Color(0xFF001A6E),
  backgroundColor:
Color(0xFF009990),
),
child: Text(_currentPage == 2 ? "Get
Started" : "Next"),
),
],
),
),
],
),
);
}
}

```

```

class SplashScreen extends StatelessWidget {
  const SplashScreen({super.key});

```

```

  @override
  Widget build(BuildContext context) {
    return const Center(
      child: Text(
        'THE DAILY GLOBE',
        style: TextStyle(
          color: Color(0xFFE1FFBB),
          fontSize: 32,
          fontWeight: FontWeight.bold,
        ),
      ),
    );
  }
}

```

```

class TrackingScreen extends StatelessWidget {
  const TrackingScreen({super.key});

```

```

  @override
  Widget build(BuildContext context) {
    return SafeArea(
      child: Padding(
        padding: const EdgeInsets.all(24.0),
        child: Column(
          mainAxisAlignment:
MainAxisAlignment.center,
          children: [
            const Text(
              'Welcome to\nThe Daily Globe!',
              textAlign: TextAlign.center,
              style: TextStyle(
                color: Color(0xFFE1FFBB),
                fontSize: 24,
                fontWeight: FontWeight.bold,
              ),
            ),
            const SizedBox(height: 8),
            const Text(
              'Please set your preferences to\nget the

```

```

app up and running.',
      textAlign: TextAlign.center,
      style: TextStyle(
        color: Colors.white70,
        fontSize: 16,
      ),
    ),
    const SizedBox(height: 24),
    Card(
      child: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment:
CrossAxisAlignment.stretch,
          mainAxisAlignment: MainAxisAlignment.min,
          children: [
            const Text(
              'This publication is\nad-
supported.',
              textAlign: TextAlign.center,
              style: TextStyle(
                fontSize: 16,
                fontWeight: FontWeight.w500,
              ),
            ),
            const SizedBox(height: 8),
            const Text(
              'Enable ad tracking to see
personalized advertising that is relevant to your
interests.',
              textAlign: TextAlign.center,
              style: TextStyle(
                fontSize: 14,
                color: Colors.black54,
              ),
            ),
            const SizedBox(height: 16),
            ElevatedButton(
              onPressed: () {},
              style: ElevatedButton.styleFrom(
                backgroundColor: const
Color(0xFF074799),
                padding: const
EdgeInsets.symmetric(vertical: 12),
              ),
              child: const Text(
                'Personalize My Ads',
                style: TextStyle(
                  color: Color(0xFFE1FFBB),
                ),
              ),
            ),
            TextButton(
              onPressed: () {
                print('No Thanks');
              },
              child: const Text(
                'No Thanks',
                style: TextStyle(color:
Colors.black54),
              ),
            ),
          ],
        ),
      ),
    ),
  ],
),

```

```

    ),
    ),
    ),
    ],
    ),
    ),
    );
  }
}

class NotificationScreen extends
StatelessWidget {
  const NotificationScreen({super.key});

  Future<void>
requestNotificationPermission(BuildContext
context) async {
  var status = await
Permission.notification.request();

  if (status.isGranted) {
    // Show success message
    // ignore: use_build_context_synchronously
    ScaffoldMessenger.of(context).showSnackBar
ar(
  const SnackBar(content: Text('Notification
permission granted!')),
  );
  } else if (status.isDenied) {
    // Show warning message
    // ignore: use_build_context_synchronously
    ScaffoldMessenger.of(context).showSnackBar
ar(
  const SnackBar(content: Text('Notification
permission denied!')),
  );
  } else if (status.isPermanentlyDenied) {
    // Open app settings if permanently denied
    // ignore: use_build_context_synchronously
    ScaffoldMessenger.of(context).showSnackBar
ar(
  const SnackBar(
    content: Text(
      'Notification permission permanently
denied. Enable from settings.'),
    ),
  );
  await openAppSettings();
  }
}

@override
Widget build(BuildContext context) {
  return SafeArea(
    child: Padding(
      padding: const EdgeInsets.all(24.0),
      child: Column(
        mainAxisAlignment:
MainAxisAlignment.center,
        children: [
          const Text(
            'Welcome to\nThe Daily Globe!',
            textAlign: TextAlign.center,

```

```

style: TextStyle(
  color: Color(0xFFE1FFBB),
  fontSize: 24,
  fontWeight: FontWeight.bold,
),
),
const SizedBox(height: 8),
const Text(
  'Please set your preferences to\nget the
app up and running.',
  textAlign: TextAlign.center,
  style: TextStyle(
    color: Colors.white70,
    fontSize: 16,
  ),
),
const SizedBox(height: 24),
Card(
  child: Padding(
    padding: const EdgeInsets.all(16.0),
    child: Column(
      crossAxisAlignment:
CrossAxisAlignment.stretch,
      mainAxisAlignment: MainAxisAlignment.min,
      children: [
        const Text(
          "Don't miss our\ntop stories!",
          textAlign: TextAlign.center,
          style: TextStyle(
            fontSize: 16,
            fontWeight: FontWeight.w500,
          ),
        ),
        const SizedBox(height: 8),
        const Text(
          'Subscribe to our notifications to
stay up to date on the latest breaking news.',
          textAlign: TextAlign.center,
          style: TextStyle(
            fontSize: 14,
            color: Colors.black54,
          ),
        ),
        const SizedBox(height: 16),
        ElevatedButton(
          onPressed: ()=>
requestNotificationPermission(context),
          style: ElevatedButton.styleFrom(
            backgroundColor: const
Color(0xFF074799),
            padding: const
EdgeInsets.symmetric(vertical: 12),
          ),
          child: const Text(
            'Subscribe to Notifications Now',
            style: TextStyle(
              color: Color(0xFFE1FFBB),
            ),
          ),
        ),
        TextButton(
          onPressed: () {},
          child: const Text(

```

```
        'No Thanks',
        style: TextStyle(color:
Colors.black54),
      ),
    ),
    1,
  ),
),
),
1,
),
),
);
}
```







