

MAD & PWA Lab Journal

Experiment No.	02
Experiment Title.	To design Flutter UI by including common widgets.
Roll No.	27
Name	Rujuta Medhi
Class	D15A
Subject	MAD & PWA Lab
Lab Outcome	LO1: Understand cross platform mobile application development using Flutter framework
Grade:	

AIM : To design Flutter UI by including common widgets.

THEORY:

Designing a Flutter UI using **common widgets** involves leveraging the foundational widgets provided by Flutter to build interactive, responsive, and visually appealing user interfaces. These common widgets allow developers to construct the basic building blocks of the app interface, such as text, buttons, images, forms, lists, and more. Here's a breakdown of how Flutter UI design incorporates common widgets:

1. Container

- **Role:** A container is a fundamental widget in Flutter, used to create a rectangular box with various properties like padding, margin, decoration, color, and alignment. It can be used to wrap other widgets and customize their appearance.
- **Usage:** Containers are useful for aligning, decorating, and sizing widgets. For example, it can be used to create a card-like structure, set background colors, or add rounded corners.

2. Row and Column

- **Role:** These are layout widgets used for arranging children widgets either horizontally (Row) or vertically (Column).
- **Usage:** These are commonly used to create flexible layouts. Rows and columns are used for simple linear layouts, but they can also be combined with other layout widgets to create more complex structures (e.g., GridView, Flex).

3. Text

- **Role:** Displays a string of text.
- **Usage:** The Text widget is used to display static text on the screen. It can be customized with various styles (font size, color, weight, etc.).

4. TextField and Form

- **Role:** TextField allows users to enter text, while Form and TextFormField are used for form validation and handling text input.
- **Usage:** Common for building login forms, registration forms, or any interactive form where user input is required.

5. Buttons (ElevatedButton, TextButton, IconButton)

- **Role:** Buttons are used to perform an action when pressed.
- **Usage:** These are the primary way to handle user interaction in Flutter. You can customize their color, shape, padding, and interaction effects.

6. Font

- **Role:** A widget used to display and customize the text within the application.
- **Usage:** The font is applied to text widgets to define their appearance, including font family, size, weight, and style. It's essential for customizing the typographic design of your app, whether you're using default fonts, custom fonts, or fonts from Google Fonts. You can apply different fonts to various text elements to enhance readability and visual appeal.

PROGRAM:

Dishes.dart

```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'dish_details.dart';
```

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

```
class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);
```

@override

```
Widget build(BuildContext context) {
  SystemChrome.setSystemUIOverlayStyle(
    const SystemUiOverlayStyle(
      statusBarColor: Colors.green,
      statusBarIconBrightness: Brightness.light,
    ),
  );
```

```
  return MaterialApp(
    debugShowCheckedModeBanner: false,
    theme: ThemeData(
      primaryColor: Colors.green,
      scaffoldBackgroundColor: Colors.white,
    ),
    home: const HomePage(),
  );
}
```

```
class HomePage extends StatelessWidget {
  const HomePage({Key? key}) : super(key: key);
```

```
  final List<Map<String, dynamic>> dishes = const [
    {
      "name": "Pav Bhaji",
      "ingredients": [
        {"name": "Potatoes", "weight": "500g"},
        {"name": "Tomatoes", "weight": "250g"},
        {"name": "Onions", "weight": "150g"},
```

```

        {"name": "Pav Bhaji Masala", "weight": "2 tbsp"},
        {"name": "Butter", "weight": "50g"},
        {"name": "Capsicum", "weight": "100g"},
        {"name": "Garlic", "weight": "10g"},
        {"name": "Green Peas", "weight": "100g"}
    ],
    "image": "assets/pavbhaji.png"
},
{
    "name": "Chinese Noodles",
    "ingredients": [
        {"name": "Noodles", "weight": "200g" },
        {"name": "Capsicum", "weight": "100g" },
        {"name": "Carrots", "weight": "100g" },
        {"name": "Spring Onions", "weight": "50g" },
        {"name": "Soy Sauce", "weight": "2 tbsp" },
        {"name": "Vinegar", "weight": "1 tbsp" },
        {"name": "Garlic", "weight": "10g" },
        {"name": "Oil", "weight": "2 tbsp" }
    ],
    "image": "assets/noodles.png"
},
{
    "name": "Pulao",
    "ingredients": [
        {"name": "Basmati Rice", "weight": "250g"},
        {"name": "Carrots", "weight": "100g"},
        {"name": "Green Peas", "weight": "100g"},
        {"name": "Beans", "weight": "100g"},
        {"name": "Onions", "weight": "100g"},
        {"name": "Cumin Seeds", "weight": "1 tsp"},
        {"name": "Ghee", "weight": "2 tbsp"},
        {"name": "Bay Leaf", "weight": "1 leaf"}
    ],
    "image": "assets/poolao.png"
},
];

```

@override

```

Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text('Dishes List'),
      backgroundColor: Colors.green,
    ),
    body: ListView.builder(

```

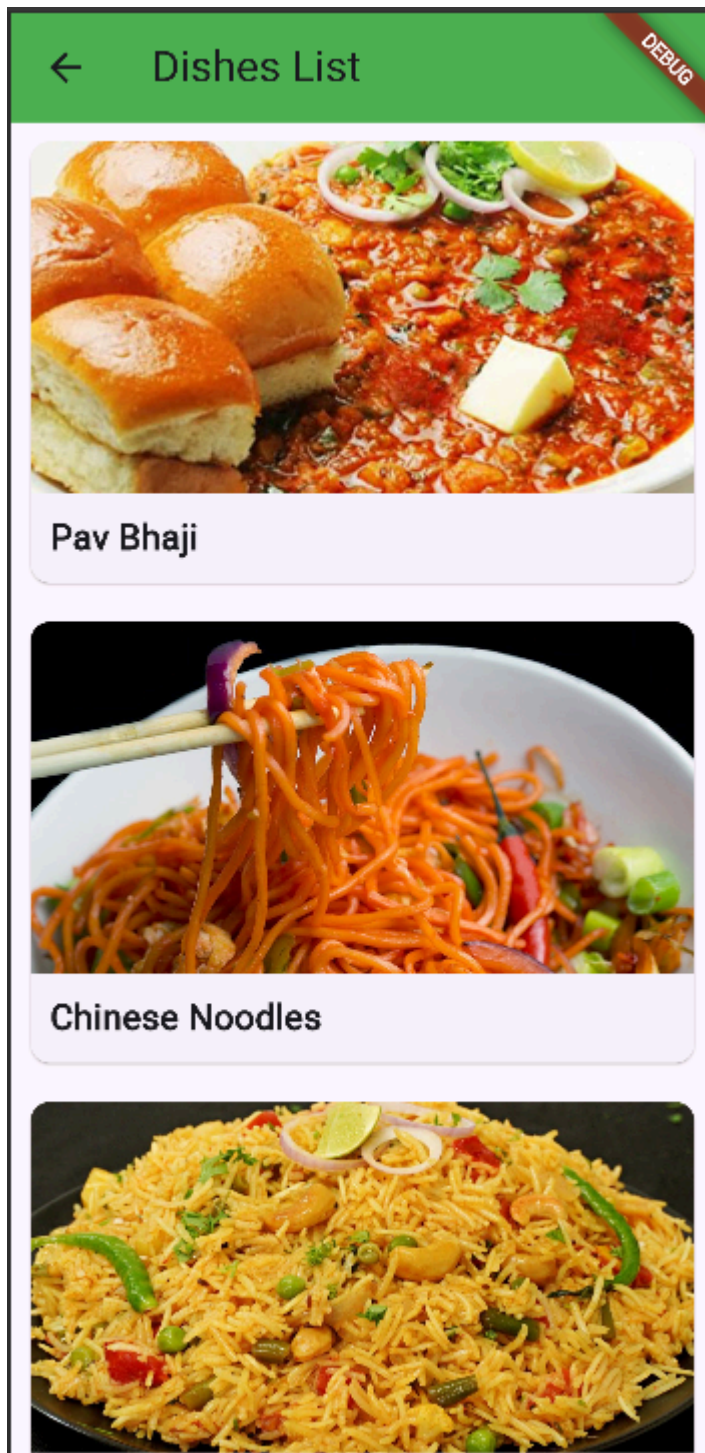
```

itemCount: dishes.length,
itemBuilder: (context, index) {
  final dish = dishes[index];

  return GestureDetector(
    onTap: () {
      Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => DishDetails(dish: dish),
        ),
      );
    },
    child: Card(
      margin: const EdgeInsets.all(10),
      shape: RoundedRectangleBorder(
        borderRadius: BorderRadius.circular(10),
      ),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          ClipRRect(
            borderRadius:
              const BorderRadius.vertical(top: Radius.circular(10)),
            child: Image.network(
              dish['image'] ?? "",
              width: double.infinity,
              height: 180,
              fit: BoxFit.cover,
              errorBuilder: (context, error, stackTrace) => Container(
                height: 180,
                color: Colors.grey[300],
                child: const Center(child: Icon(Icons.broken_image)),
              ),
            ),
          ),
          Padding(
            padding: const EdgeInsets.all(10),
            child: Text(
              dish['name'] ?? 'Unknown Dish',
              style: const TextStyle(
                fontSize: 18,
                fontWeight: FontWeight.bold,
              ),
            ),
          ),
        ],
      ),
    ),
  );
}

```

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



```

import 'package:flutter/material.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Product Detail',
      theme: ThemeData(
        primarySwatch: Colors.pink,
        elevatedButtonTheme: ElevatedButtonThemeData(
          style: ElevatedButton.styleFrom(
            backgroundColor: Colors.green,
            foregroundColor: Colors.white,
          ),
        ),
      ),
      home: const ProductDetailPage(),
    );
  }
}

```

```

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        leading: IconButton(
          icon: const Icon(Icons.arrow_back),
          onPressed: () {},
        ),
        title: const Text('Ice cream'),
      ),
      body: SingleChildScrollView(
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Image.network(

```

'https://hebbkx1anhila5yf.public.blob.vercel-storage.com/image-1hitW4RCVecC6jelz0Hlz1YCWy7YbG.png',

height: 300,

width: double.infinity,

fit: BoxFit.cover,

),

Padding(

padding: const EdgeInsets.all(16.0),

child: Column(

crossAxisAlignment: CrossAxisAlignment.start,

children: [

const Text(

"Kwality Wall's Choco-Mint Oreo Brownie Fudge", // Fixed this line

style: TextStyle(

fontSize: 20,

fontWeight: FontWeight.bold,

),

),

const SizedBox(height: 8),

Row(

children: [

const Icon(Icons.star, color: Colors.amber, size: 20),

const Text('4.5'),

const Spacer(),

Text(

'₹255.20',

style: TextStyle(

fontSize: 20,

fontWeight: FontWeight.bold,

color: Theme.of(context).primaryColor,

),

),

],

),

const SizedBox(height: 24),

const Text(

'About the Product',

style: TextStyle(

fontSize: 18,

fontWeight: FontWeight.bold,

),

),

const SizedBox(height: 8),

const Text(


'A delicious blend of chocolate and mint ice cream with Oreo cookie pieces and brownie fudge swirls. Perfect for dessert lovers who enjoy the combination of refreshing mint with rich chocolate.'

```
      style: TextStyle(
        fontSize: 16,
        color: Colors.grey,
      ),
    ),
    const SizedBox(height: 24),
    const Text(
      'How to Use',
      style: TextStyle(
        fontSize: 18,
        fontWeight: FontWeight.bold,
      ),
    ),
    const SizedBox(height: 8),
    const Text(
      'Store in freezer. Allow to soften slightly before serving for best taste and texture.',
      style: TextStyle(
        fontSize: 16,
        color: Colors.grey,
      ),
    ),
    const SizedBox(height: 24),
    Row(
      children: [
        Expanded(
          child: ElevatedButton(
            onPressed: () {},
            style: ElevatedButton.styleFrom(
              padding: const EdgeInsets.symmetric(vertical: 16),
            ),
            child: const Text(
              'Add',
              style: TextStyle(fontSize: 16),
            ),
          ),
        ),
      ],
    ),
  ],
),
],
),
```

```
),  
);  
}  
}
```

Pav Bhaji

DEBUG



Ingredients

Potatoes

-

1

+

500 g

Tomatoes

-

1

+

250 g

Onions

-

1

+

150 g

Pav Bhaji Masala

-

1

+

2 tbsp

Butter

-

1

+

50 g

Capsicum

-

1

+

100 g

Garlic

-

1

+

10 g

Green Peas

-

1

+

100 g

Place Order

Order Form

```
import 'package:flutter/material.dart';
import 'package:google_fonts/google_fonts.dart';

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

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        textTheme: GoogleFonts.poppinsTextTheme(Theme.of(context).textTheme),
      ),
      home: const RegistrationForm(),
    );
  }
}

class RegistrationForm extends StatefulWidget {
  const RegistrationForm({Key? key}) : super(key: key);

  @override
  _RegistrationFormState createState() => _RegistrationFormState();
}

class _RegistrationFormState extends State<RegistrationForm> {
  final _formKey = GlobalKey<FormState>();
  final _controllers = List.generate(5, (_) => TextEditingController());
  final _focusNodes = List.generate(5, (_) => FocusNode());

  @override
  void dispose() {
    for (var controller in _controllers) {
      controller.dispose();
    }
    for (var node in _focusNodes) {

```

```

    node.dispose();
  }
  super.dispose();
}

```

@override

```

Widget build(BuildContext context) {
  return Scaffold(
    body: Container(
      decoration: const BoxDecoration(
        gradient: LinearGradient(
          begin: Alignment.topLeft,
          end: Alignment.bottomRight,
          colors: [Color(0xFFE8F5E9), Colors.white],
        ),
      ),
    ),
    child: SafeArea(
      child: CustomScrollView(
        slivers: [
          SliverAppBar(
            floating: true,
            backgroundColor: Colors.transparent,
            elevation: 0,
            flexibleSpace: FlexibleSpaceBar(
              title: Text(
                'Order Form',
                style: GoogleFonts.poppins(
                  color: Colors.green[800],
                  fontWeight: FontWeight.w600,
                ),
              ),
            ),
          ),
          SliverToBoxAdapter(
            child: Padding(
              padding: const EdgeInsets.all(24.0),
              child: Form(
                key: _formKey,
                child: Column(
                  crossAxisAlignment: CrossAxisAlignment.start,
                  children: [
                    Text(
                      'Please fill out the form to order',
                      style: GoogleFonts.poppins(
                        fontSize: 16,
                        color: Colors.green[800],

```

```

        fontWeight: FontWeight.w500,
      ),
    ),
    const SizedBox(height: 24),
    ..._buildFormFields(),
    const SizedBox(height: 32),
    _buildSubmitButton(),
  ],
),
),
),
),
),
],
),
),
),
);
}

```

```

List<Widget> _buildFormFields() {
  final labels = ['First Name', 'Last Name', 'Address', 'Landmark', 'Mobile Number'];
  final icons = [Icons.person, Icons.person, Icons.home, Icons.location_on, Icons.phone];

  return List.generate(
    5,
    (index) => Padding(
      padding: const EdgeInsets.only(bottom: 16),
      child: TextFormField(
        controller: _controllers[index],
        focusNode: _focusNodes[index],
        decoration: _getInputDecoration(labels[index], icons[index]),
        validator: (value) {
          if (value == null || value.isEmpty) {
            return 'Please enter ${labels[index].toLowerCase()}'; // Ensures the field is required
          }
          if (index == 4 && value.length != 10) {
            return 'Mobile number must be 10 digits'; // Mobile number validation
          }
          return null;
        },
        onFieldSubmitted: (_) {
          if (index < 4) {
            FocusScope.of(context).requestFocus(_focusNodes[index + 1]);
          }
        },
        keyboardType: index == 4 ? TextInputType.phone : TextInputType.text,

```

```

    ),
  ),
);
}

```

```

InputDecoration _getInputDecoration(String label, IconData icon) {
  return InputDecoration(
    labelText: label,
    prefixIcon: Icon(icon, color: Colors.green[800]),
    labelStyle: TextStyle(color: Colors.green[800]),
    enabledBorder: OutlineInputBorder(
      borderRadius: BorderRadius.circular(12),
      borderSide: BorderSide(color: Colors.green[200]!),
    ),
    focusedBorder: OutlineInputBorder(
      borderRadius: BorderRadius.circular(12),
      borderSide: BorderSide(color: Colors.green[800]!, width: 2),
    ),
    errorBorder: OutlineInputBorder(
      borderRadius: BorderRadius.circular(12),
      borderSide: BorderSide(color: Colors.red[400]!),
    ),
    focusedErrorBorder: OutlineInputBorder(
      borderRadius: BorderRadius.circular(12),
      borderSide: BorderSide(color: Colors.red[400]!, width: 2),
    ),
    filled: true,
    fillColor: Colors.white,
  );
}

```

```

Widget _buildSubmitButton() {
  return SizedBox(
    width: double.infinity,
    child: ElevatedButton(
      onPressed: _submitForm,
      style: ElevatedButton.styleFrom(
        backgroundColor: Colors.green[800],
        padding: const EdgeInsets.symmetric(vertical: 16),
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(12),
        ),
      ),
      child: Text(
        'SUBMIT',
        style: GoogleFonts.poppins(

```

```
        color: Colors.white,  
        fontSize: 18,  
        fontWeight: FontWeight.w600,  
      ),  
    ),  
  ),  
);  
}  
  
void _submitForm() {  
  if (_formKey.currentState!.validate()) {  
    ScaffoldMessenger.of(context).showSnackBar(  
      SnackBar(  
        content: Text(  
          'Order Successful!',  
          style: GoogleFonts.poppins(),  
        ),  
        backgroundColor: Colors.green[800],  
      ),  
    );  
  }  
}  
}
```