

MAD & PWA Lab Journal

Experiment No.	03
Experiment Title.	To include image, fonts and icons in flutter app.
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 include image, fonts and icons in flutter app.

## **THEORY:**

### **Including Images, Fonts, and Icons in a Flutter App (Theory)**

Flutter allows developers to **enhance UI/UX** by including **images, custom fonts, and icons** efficiently. Below is an overview of how each of these assets can be integrated into a Flutter application.

#### **1. Adding Images in Flutter**

Flutter supports **both local and network images** for UI components.

##### **Local Images**

- Images can be stored in the **assets** folder of the project.
- The `pubspec.yaml` file needs to be updated to include the images.
- Supported formats: **PNG, JPG, GIF, SVG** (with additional packages like `flutter_svg`).

##### **Network Images**

- Images can be loaded from the internet using URLs.
- Cached network images can be used for better performance.

#### **2. Adding Custom Fonts in Flutter**

Custom fonts improve branding and UI aesthetics.

- **Download and store** font files in the `assets/fonts/` directory.
- Declare them in `pubspec.yaml` under the `fonts` section.
- Assign the font to specific text styles using the `TextStyle` widget.

#### **3. Using Icons in Flutter**

Icons enhance UI clarity and usability. Flutter supports **built-in icons and custom icons**.

##### **Built-in Icons (Material Icons)**

- Flutter provides a rich set of **Material Icons** through the `Icons` class.
- These icons do not require additional setup.

##### **Custom Icons (SVG, PNG, Font Icons)**

- Icons can be added as **SVG** (using `flutter_svg` package) or **PNG** images.
- Font-based icon sets like **FontAwesome** or **Custom Icon Fonts** can be integrated.

## **PROGRAM:**

```
Prod_details.dart
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(
    'assets/sunday.jpg',
    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),
            ),
          ),
        ),
      ],
    ),
  ],
),
],
),
```

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

 Ice cream DEBUG



### Kwality Wall's Choco-Mint Oreo Brownie Fudge

 4.5 ₹255.20

#### About the Product

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.

#### How to Use

Store in freezer. Allow to soften slightly before serving for best taste and texture.

Add

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

## Order Form

Please fill out the form to order

 First Name

 Last Name

 Address

 Landmark


 Mobile Number

SUBMIT

## Order Form

Please fill out the form to order

First Name

 Rujuta

Please enter first name

 Last Name

Please enter last name

 Address

Please enter address

 Landmark

Please enter landmark

 Mobile Number

Please enter mobile number

SUBMIT