**Name: Niyati V. Gaonkar**
**Class: D15B Roll no.: 17**

# MAD-PWD Experiment - 3

**AIM:** To add advanced Flutter UI by including widgets like Image, Fonts, Icons.

**THEORY:**
Flutter is a popular open-source framework for building cross-platform mobile applications. One of its key strengths is the large collection of customisable widgets that make it easy to create beautiful and functional user interfaces.Whenever you are going to code for building anything in Flutter, it will be inside a widget. Widgets are nested with each other to build the app. It means the root of your app is itself a widget, and all the way down is a widget also. For example, a widget can display something, can define design, can handle interaction, etc. The Flutter UI widgets are designed for speed. These widget parts are grouped in the shape of a widget tree.

Types of advance Widget:

- AnimatedContainer
  The AnimatedContainer widget is a convenient way to animate the properties of a container, such as its size, padding, and color. To use this widget, simply wrap it around another widget and specify the properties you want to animate. Here's an example that animates the width of a container over 500 milliseconds:
  Syntax:

```
AnimatedContainer(
  duration: Duration(milliseconds: 500),
  width: _width,
  child: Container(
    color: Colors.red,
  ),
);
```

- Expanded
  The Expanded widget is used to control the distribution of space within a Flex layout. You can use it to make one widget take up more space than the others within the same row or column. Here's an example that creates a row with two Expanded widgets and a Text widget, where the first Expanded widget takes up 2 times more space than the other:
  Syntax:

```
Row(
  children: [
    Expanded(
      flex: 2,
      child: Container(
        color: Colors.red,
      ),
    ),
    Expanded(
      child: Container(
        color: Colors.blue,
      ),
    ),
    Text("Hello World"),
  ],
);
```

- Image :
  An asset is a file, which is bundled and deployed with the app and is accessible at runtime. The asset can include static data, configuration files, icons, and images. The Flutter supports many image formats, such as JPEG, WebP, PNG, GIF, animated WebP/GIF, BMP, and WBMP.
  Displaying images is the fundamental concept of most of the mobile apps. Flutter has an Image widget that allows displaying different types of images in the mobile application.
  Syntax:

```
children: <Widget>[
          Image.asset('assets/tablet.png'),
]
```

- Icon:
  The row widget is similar to the column widget, but it constructs a widget horizontally rather than vertically.
  Syntax:
  children: <Widget>[
  Icon(Icons.camera_enhance),
  Icon(Icons.camera_front),
  Icon(Icons.camera_rear),
  ]),

**CODE:**

main.dart :

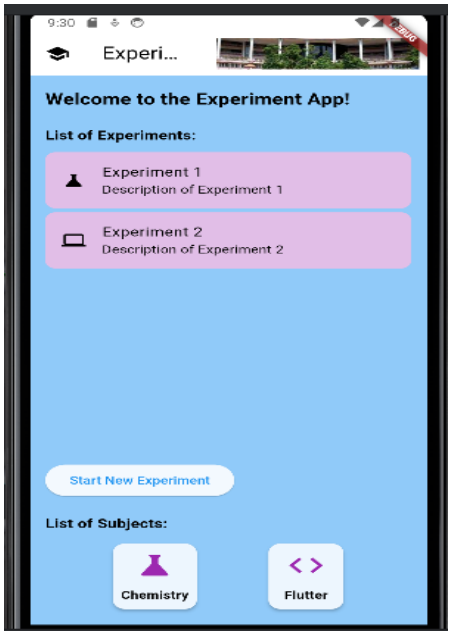
```
import 'package:flutter/material.dart';

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

class MyApp extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
   return MaterialApp(
     title: 'Experiment App',
     theme: ThemeData(
       primarySwatch: Colors.lightBlue,
       colorScheme: ColorScheme.fromSwatch().copyWith(
         secondary: Colors.purple,
       ),
       hintColor: Colors.black,
       fontFamily: 'Montserrat',
     ),
     home: MyHomePage(),
   );
 }
}

class MyHomePage extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
   return Scaffold(
     appBar: AppBar(
       title: const Text('Experiment App'),
```

```
      leading: const Icon(Icons.school), // Adding an icon to the app bar
      actions: [
        Padding(
          padding: const EdgeInsets.all(8.0),
          child: Image.asset(
            'assets/vesit.jpg',
            width: 200,
            height: 50,
            fit: BoxFit.cover,
          ),
        ),
      ],
    ),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          const Text(
            'Welcome to the Experiment App!',
            style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold),
          ),
          const SizedBox(height: 20),
          const Text(
            'List of Experiments:',
            style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
          ),
          const SizedBox(height: 10),
          Expanded(
            child: ListView(
              children: [
                ListTile(
                  leading: const Icon(Icons.science), // Experiment 1 icon
                  title: const Text('Experiment 1'),
                  subtitle: const Text('Description of Experiment 1'),
                  tileColor: Colors.purple.shade100,
                  shape: RoundedRectangleBorder(
                    borderRadius: BorderRadius.circular(10),
                  ),
                  onTap: () {},
                ),
                const SizedBox(height: 5),
                ListTile(
                  leading: const Icon(Icons.computer), // Experiment 2 icon
                  title: const Text('Experiment 2'),
                  subtitle: const Text('Description of Experiment 2'),
                  tileColor: Colors.purple.shade100,
                  shape: RoundedRectangleBorder(
                    borderRadius: BorderRadius.circular(10),
```

```
                  ),
                  onTap: () {},
                ),
              ],
            ),
          ),
          const SizedBox(height: 20),
          ElevatedButton(
            onPressed: () {},
            child: const Text('Start New Experiment'),
          ),
          const SizedBox(height: 20),
          const Text(
            'List of Subjects:',
            style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
          ),
          const SizedBox(height: 10),
          const Row(
            mainAxisAlignment: MainAxisAlignment.spaceEvenly,
            children: [
              _SubjectCard(
                icon: Icons.science,
                title: 'Chemistry',
              ),
              _SubjectCard(
                icon: Icons.code,
                title: '  Flutter  ',
              ),
            ],
          ),
        ],
      ),
    ),
  );
 }
}

class _SubjectCard extends StatelessWidget {
 final IconData icon;
 final String title;

 const _SubjectCard({
   required this.icon,
   required this.title,
 });

 @override
 Widget build(BuildContext context) {
   return Card(
```

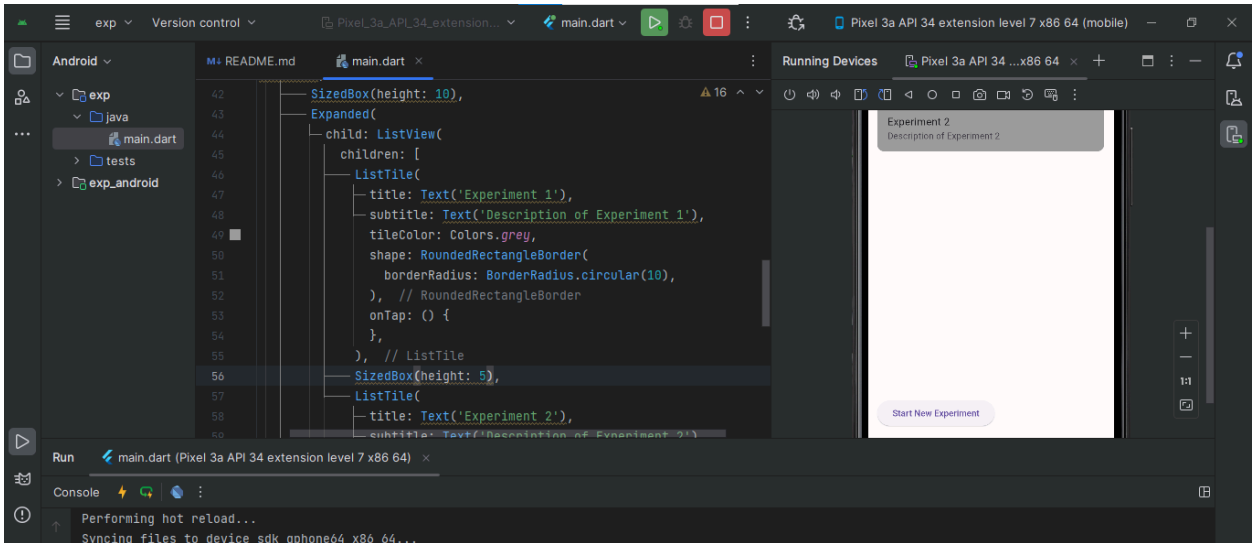
```
      elevation: 3,
      shape: RoundedRectangleBorder(
        borderRadius: BorderRadius.circular(12),
      ),
      child: Padding(
        padding: const EdgeInsets.all(8.0),
        child: Column(
          mainAxisSize: MainAxisSize.min,
          children: [
            Icon(
              icon,
              size: 40,
              color: Colors.purple,
            ),
            const SizedBox(height: 8),
            Text(
              title,
              style: const TextStyle(
                fontWeight: FontWeight.bold,
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

**Output**:

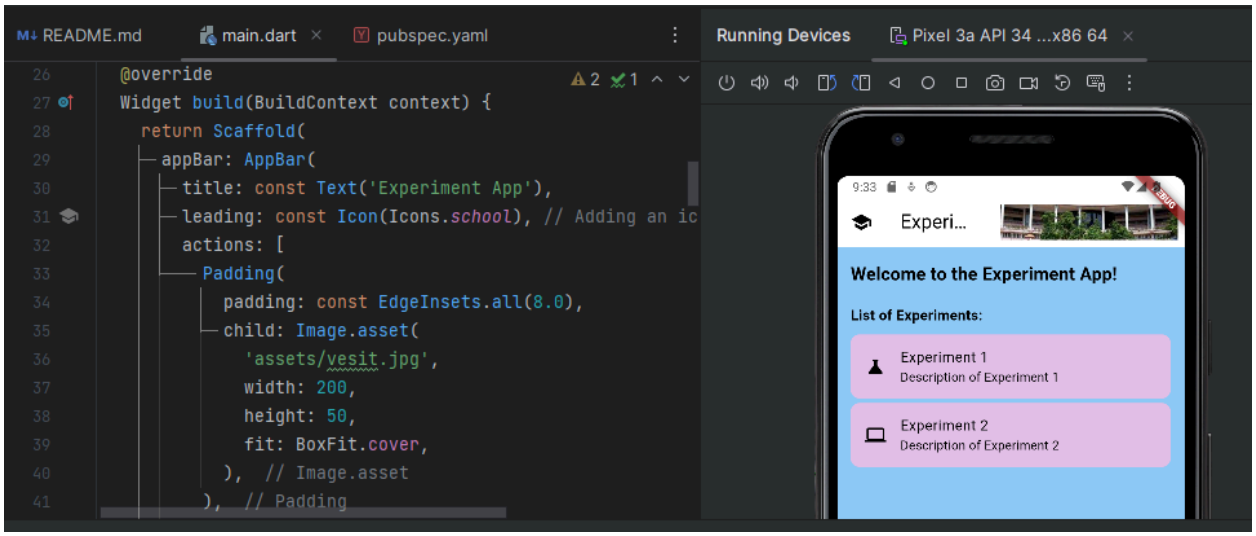


**EXPLANATION** :

Step 1: Create a new flutter project and setup with initial code. Initial template is set from experiment 2.

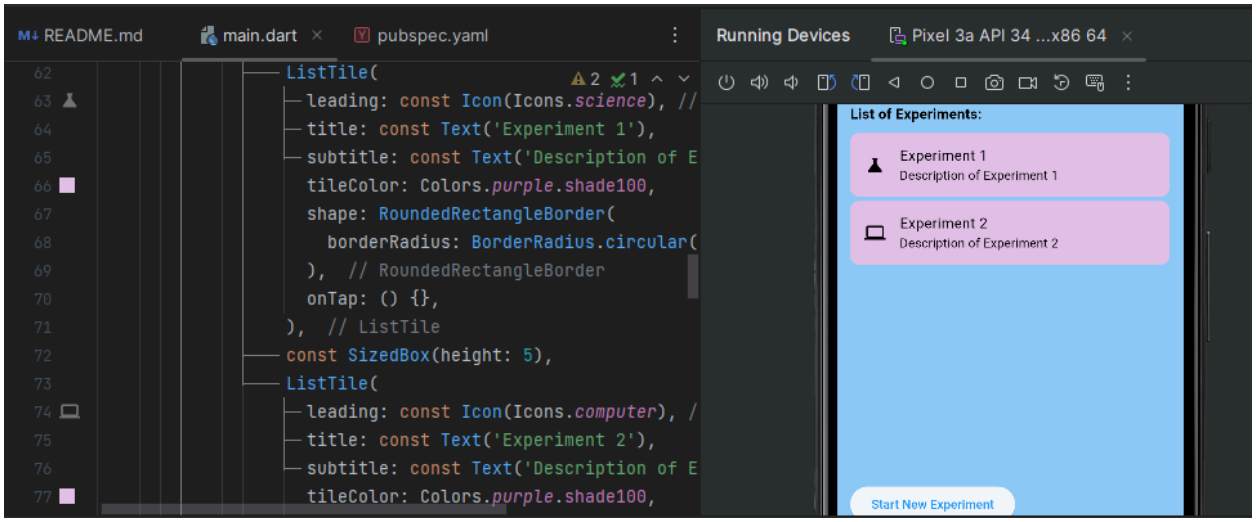



Step 2: Add image at the top. For that first initialize the image in .yaml file . Then put it wherever necessary .

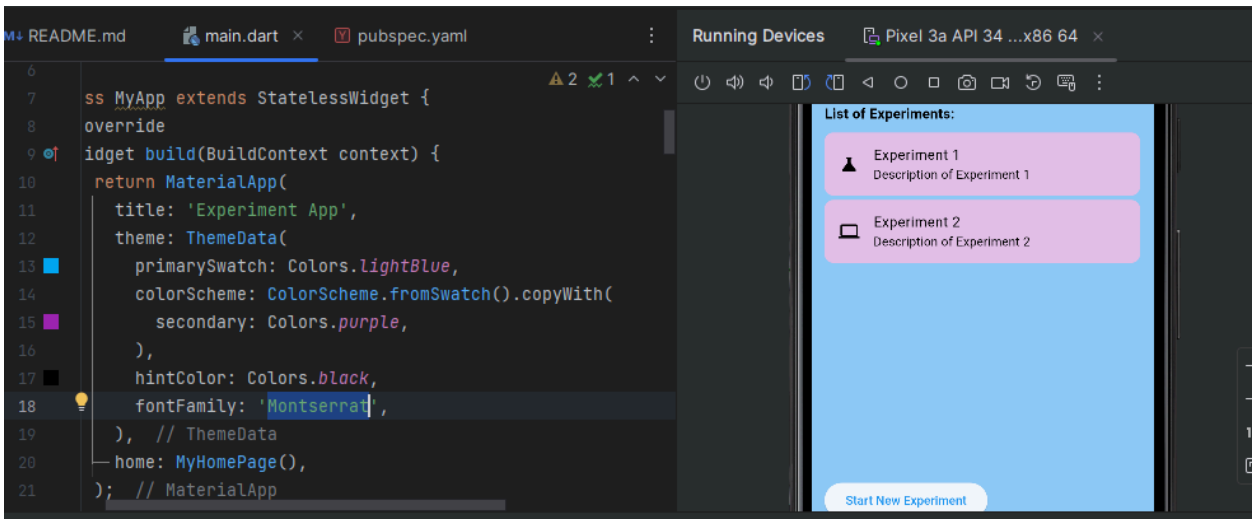



Step 3: Then add icons for each experiment..

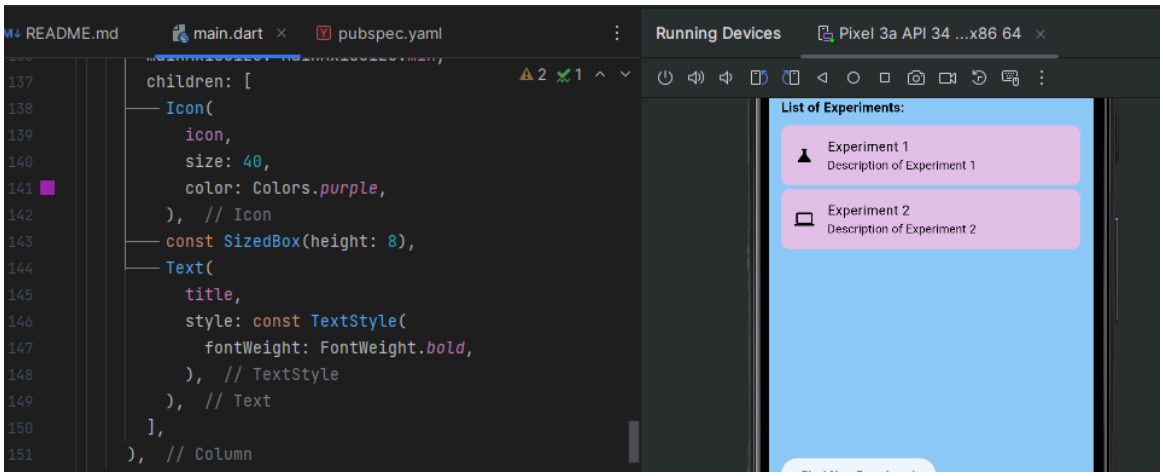
- Text: It displays a short piece of text. In your code, you have used it for displaying welcome messages and headings.



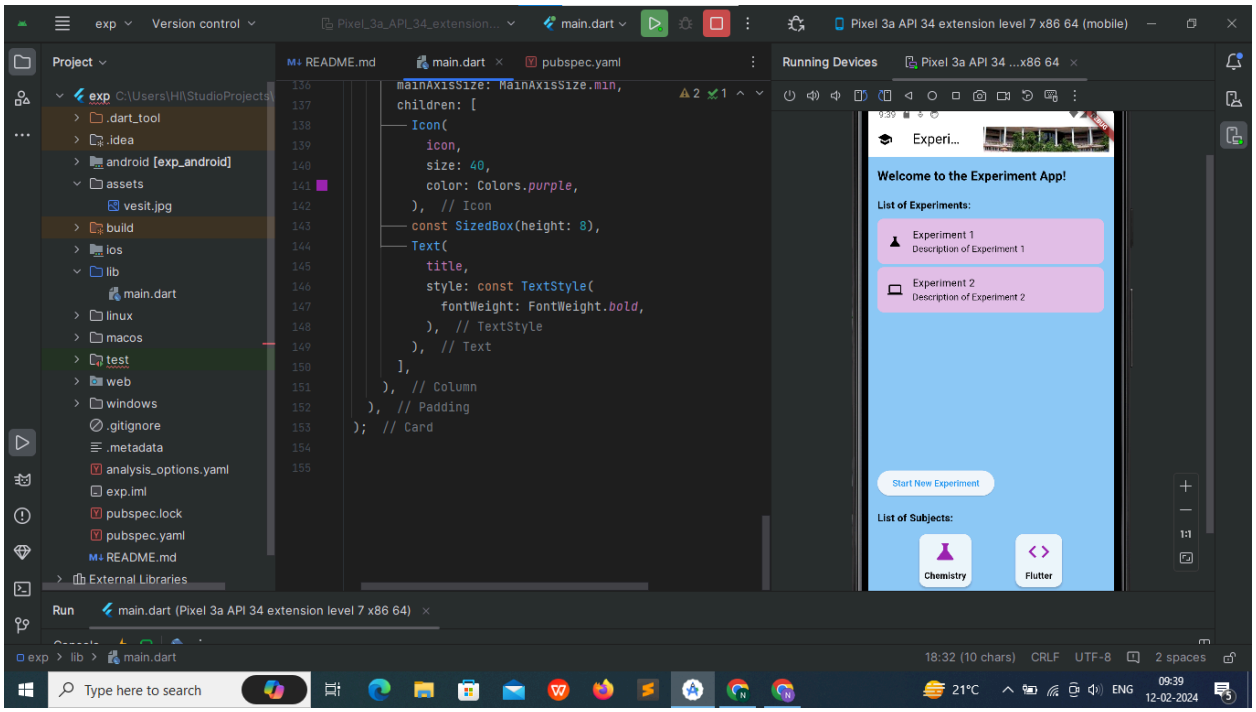

Step 4: Add different styles of font.




You can also make it bold using fontWeight.



Step 5: Add some more icons at the bottom and the app is ready.




**CONCLUSION:** Thus we have created an advanced flutter UI . We have used different widgets of flutter like: Icons,Images,Fonts, etc and we have understood their usage. We have created an experiment app to add experiments of different subjects.