

## Exp 2

Rachana Rane  
Batch:C

Roll No:48  
D15A

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

**Theory:** Flutter is a powerful open-source UI framework developed by Google for creating natively compiled applications for mobile, web, and desktop from a single codebase. It uses the Dart programming language.

Common Widgets in Flutter:

Container:

- A box model for styling and positioning other widgets.

Text:

- Displays text with customizable styles.

Image:

- Displays images from various sources.

Column and Row:

- Arranges children widgets in a vertical (Column) or horizontal (Row) sequence.

ListView:

- Displays a scrolling list of widgets.

AppBar:

- A material design app bar with a title and optional actions.

Scaffold:

- Implements the basic material design visual layout structure.

**Code:**

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

```
void main() {  
  runApp(login());  
}
```

```
class login extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Login Page',  
      theme: ThemeData(  
        primarySwatch: Colors.blue,  
      ),  
      home: LoginPage(),  
    );  
  }  
}
```

```
}  
}
```

```
class LoginPage extends StatefulWidget {  
  @override  
  _LoginPageState createState() => _LoginPageState();  
}
```

```
class _LoginPageState extends State<LoginPage> {  
  TextEditingController _usernameController = TextEditingController();  
  TextEditingController _passwordController = TextEditingController();
```

```
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      body: Container(  
        decoration: BoxDecoration(  
          image: DecorationImage(  
            image: NetworkImage(  

```

```
'https://e0.pxfuel.com/wallpapers/281/550/desktop-wallpaper-advertising-background-stirfried-synthetic-creative-catering-stirfrystir-fried-creative-synthe-food-poster-food-poster-design-food-background-thumbnail.jpg',
```

```
      ),  
      fit: BoxFit.cover,  
    ),  
  ),  
  child: Padding(  
    padding: const EdgeInsets.all(16.0),  
    child: Column(  
      mainAxisAlignment: MainAxisAlignment.center,  
      children: [  
        TextField(  
          controller: _usernameController,  
          style: TextStyle(  
            color: Colors.white), // Change the color of entered text  
          decoration: InputDecoration(  
            labelText: 'Enter Username',  
            labelStyle: TextStyle(color: Colors.white),  
            enabledBorder: UnderlineInputBorder(  
              borderSide: BorderSide(color: Colors.white),  
            ),  
          ),  
        ),  
        SizedBox(height: 16.0),  
        TextField(  
          controller: _passwordController,  
          obscureText: true,
```

```

        style: TextStyle(
          color: Colors.white), // Change the color of entered text
        decoration: InputDecoration(
          labelText: 'Enter Password',
          labelStyle: TextStyle(color: Colors.white),
          enabledBorder: UnderlineInputBorder(
            borderSide: BorderSide(color: Colors.white),
          ),
        ),
      ),
    ),
    SizedBox(height: 16.0),
    ElevatedButton(
      onPressed: () {
        // Implement your login logic here
        String username = _usernameController.text;
        String password = _passwordController.text;

        // Dummy validation for illustration purposes
        if (username == 'user' && password == 'password') {
          // Navigate to the home page on successful login
          Navigator.push(
            context,
            MaterialPageRoute(builder: (context) => HomePage()),
          );
        } else {
          // Display an error message or handle unsuccessful login
          print('Login Failed');
        }
      },
      child: Text('Login'),
    ),
  ],
),
),
),
);
}
}

```

Output:

