Experiment - 4

Aim: Create an interactive form using form widgets in flutter.

Theory:

Creating an interactive form using form widgets in Flutter involves utilizing various widgets provided by the Flutter framework to capture user input, validate it, and handle form submission. Here's the theoretical explanation of the key components involved:

StatefulWidget:

- A StatefulWidget is used to create a widget that can dynamically change its state.
- It is required because forms in Flutter often need to update their UI based on user interactions.

GlobalKey<FormState>:

- A GlobalKey is a unique identifier used to access the state of a widget across different parts of the widget tree.
- GlobalKey<FormState> specifically is used to uniquely identify a Form widget and perform operations such as validation and saving form data.

Form Widget:

- The Form widget is used to create a form and manages the form's state.
- It provides methods for validating and saving form data.
- It is typically placed as the root widget of the form's widget tree.

TextFormField Widget:

- TextFormField is a form field that allows the user to enter text.
- It includes options for decoration, validation, and saving the entered value.

Decoration:

- InputDecoration is used to customize the appearance of form fields.
- It allows you to add labels, hints, icons, and borders to form fields.

Validation:

- Validation ensures that the entered data meets specific criteria.
- In Flutter, validation is typically performed using the validator property of form fields.
- Validators are functions that return an error message if the entered data is invalid, or null if it is valid.

Saving Form Data:

- Saving form data involves extracting the entered values from form fields and storing them for further processing.
- In Flutter, the onSaved property of form fields is used to specify a callback function that is called when the form is saved.
- Inside this callback function, you can save the entered values to variables or perform any necessary operations.

Form Submission:

- Form submission involves handling the data entered by the user, typically by sending it to a server or performing some other action.
- In the example provided, form submission is simulated by printing the entered data to the console.
- In a real-world scenario, you would replace the print statements with code to handle the form data as required by your application.

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

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Health App Dashboard',
      theme: ThemeData(
        primarySwatch: Colors.green,
      ),
      home: const Dashboard(),
    );
  }
}

class Dashboard extends StatelessWidget {
```

```dart
  const Dashboard({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Health App Dashboard'),
        backgroundColor: const Color.fromARGB(255, 99, 91, 209),
        leading: Image.asset(
          'health.png', // Replace 'health.png' with the actual
path to your image asset
          width: 40,
          height: 40,
        ),
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.stretch,
          children: <Widget>[
            buildCard(
              context,
              title: 'BMI Calculator',
              icon: Icons.calculate,
              onTap: () {
                // Navigate to BMI Calculator page
              },
            ),
            const SizedBox(height: 10),
            buildCard(
              context,
              title: 'Exercise Tracker',
              icon: Icons.directions_run,
              onTap: () {
```

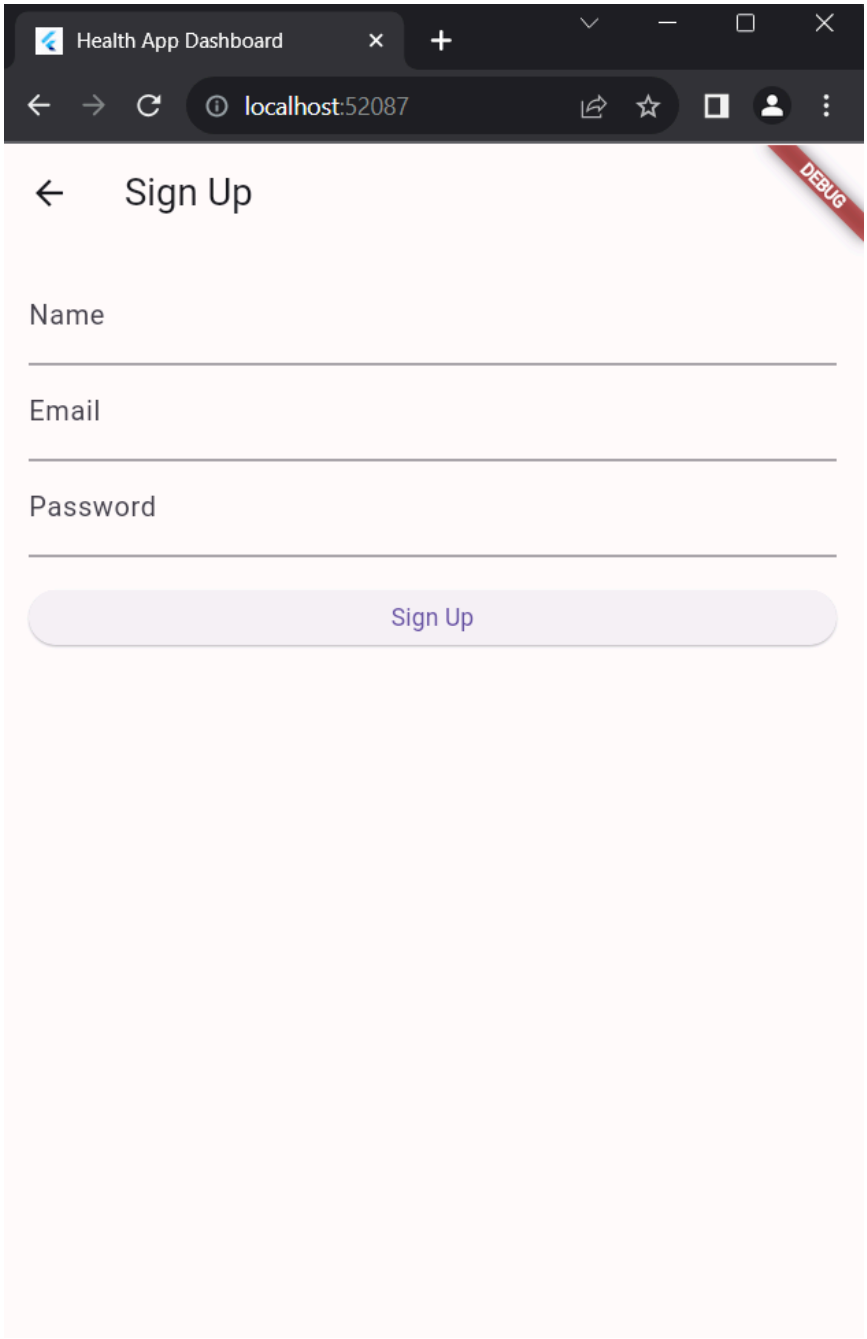
```
              // Navigate to Exercise Tracker page
            },
          ),
          const SizedBox(height: 10),
          buildCard(
            context,
            title: 'Nutrition Tracker',
            icon: Icons.restaurant_menu,
            onTap: () {
              // Navigate to Nutrition Tracker page
            },
          ),
          const SizedBox(height: 10),
          buildCard(
            context,
            title: 'Sleep Tracker',
            icon: Icons.hotel,
            onTap: () {
              // Navigate to Sleep Tracker page
            },
          ),
          const SizedBox(height: 10),
          buildCard(
            context,
            title: 'Meditation',
            icon: Icons.self_improvement,
            onTap: () {
              // Navigate to Meditation page
            },
          ),
          const SizedBox(height: 20),
          Container(
            margin: const EdgeInsets.symmetric(horizontal:
20),
```

```
              decoration: BoxDecoration(
                color: Colors.blue,
                borderRadius: BorderRadius.circular(10),
              ),
              child: Material(
                color: Colors.transparent,
                child: InkWell(
                  onTap: () {
                    Navigator.push(
                      context,
                      MaterialPageRoute(builder: (context) =>
SignUpForm()),
                    );
                  },
                  borderRadius: BorderRadius.circular(10),
                  child: Container(
                    padding: const
EdgeInsets.symmetric(vertical: 14),
                    alignment: Alignment.center,
                    child: Row(
                      mainAxisAlignment:
MainAxisAlignment.center,
                      children: [
                        Image.asset(
                          'android.png', // Replace
'android.png' with the actual path to your Android image asset
                          width: 20,
                          height: 20,
                        ),
                        const SizedBox(width: 10),
                        Image.asset(
                          'ios.png', // Replace 'ios.png' with
the actual path to your iOS image asset
                          width: 20,
```

```
                        height: 20,
                      ),
                      const SizedBox(width: 10),
                      const Text(
                        'Sign Up',
                        style: TextStyle(
                          color: Colors.white,
                          fontSize: 18,
                          fontWeight: FontWeight.bold,
                        ),
                      ),
                    ],
                  ),
                ),
              ),
            ),
          ),
        ],
      ),
    ),
  );
 }

 Widget buildCard(BuildContext context,
     {required String title,
     required IconData icon,
     required VoidCallback onTap}) {
   return InkWell(
     onTap: onTap,
     child: Card(
       elevation: 4,
       shape: RoundedRectangleBorder(borderRadius:
BorderRadius.circular(10)),
       child: Padding(
```

```
          padding: const EdgeInsets.all(16.0),
          child: Row(
            children: [
              Icon(icon, size: 32, color: Colors.blueAccent),
              const SizedBox(width: 20),
              Text(
                title,
                style: const TextStyle(fontSize: 20, fontWeight:
FontWeight.bold),
              ),
              const Spacer(),
              const Icon(Icons.arrow_forward),
            ],
          ),
        ),
      ),
    );
  }
}

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Sign Up'),
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.stretch,
          children: <Widget>[
```

```
            // Your sign-up form fields go here
            TextFormField(
              decoration: InputDecoration(labelText: 'Name'),
            ),
            TextFormField(
              decoration: InputDecoration(labelText: 'Email'),
            ),
            TextFormField(
              decoration: InputDecoration(labelText:
'Password'),
              obscureText: true,
            ),
            const SizedBox(height: 20),
            ElevatedButton(
              onPressed: () {
                // Implement sign-up logic here
              },
              child: const Text('Sign Up'),
            ),
          ],
        ),
      ),
    );
  }
}
```



Conclusion: Creating interactive forms in Flutter using form widgets is essential for building user-friendly applications that collect and process user input effectively.