## Experiment 4

**Aim: To create an interactive Form using form widget**

**Theory:**
Flutter provides a Form widget to handle user input in a structured and validated manner. Forms are used to collect data such as login credentials, contact information, or any user-provided input.

Key widgets and concepts used:

- Form: Groups multiple input fields together.
- GlobalKey<FormState>: Used to validate and manage the form's state.
- TextFormField: Input field with built-in validation capabilities.
- Validator: A function that checks the user input and displays error messages if invalid.
- ElevatedButton: Triggers form submission or validation.

**Code:**

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


void main() => runApp(FormApp());


class FormApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Form Demo',
      debugShowCheckedModeBanner: false,
      theme: ThemeData.dark(),
```

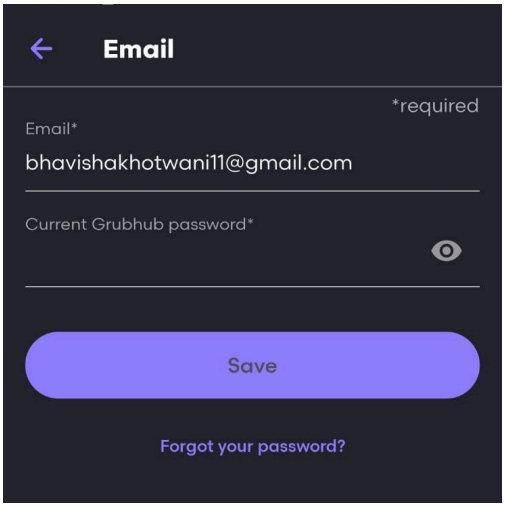
```
      home: EmailFormScreen(),
    );
  }
}

class EmailFormScreen extends StatefulWidget {
  @override
  _EmailFormScreenState createState() => _EmailFormScreenState();
}

class _EmailFormScreenState extends State<EmailFormScreen> {
  final _formKey = GlobalKey<FormState>();
  bool _obscurePassword = true;

  final TextEditingController _emailController = TextEditingController();
  final TextEditingController _passwordController = TextEditingController();

  void _submitForm() {
    if (_formKey.currentState!.validate()) {
      // Do something (e.g., API call or local processing)
      ScaffoldMessenger.of(context).showSnackBar(SnackBar(content:
Text("Form Submitted")));
    }
  }

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

```
    appBar: AppBar(
      leading: BackButton(),
      title: Text("Email"),
    ),
    body: Padding(
      padding: const EdgeInsets.all(20.0),
      child: Form(
        key: _formKey,
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Text("Email*", style: TextStyle(color: Colors.grey)),
            TextFormField(
              controller: _emailController,
              decoration: InputDecoration(
                hintText: 'yourname@email.com',
              ),
              validator: (value) =>
                  value != null && value.contains('@') ? null : 'Enter a valid email',
            ),
            SizedBox(height: 20),
            Text("Current Grubhub password*", style: TextStyle(color:
Colors.grey)),
            TextFormField(
              controller: _passwordController,
              obscureText: _obscurePassword,
              decoration: InputDecoration(
                hintText: 'Password',
```

```
    suffixIcon: IconButton(
      icon: Icon(
        _obscurePassword ? Icons.visibility : Icons.visibility_off,
      ),
      onPressed: () {
        setState(() {
          _obscurePassword = !_obscurePassword;
        });
      },
    ),
  ),
  validator: (value) => value != null && value.length >= 6
      ? null
      : 'Password must be at least 6 characters',
),
SizedBox(height: 30),
ElevatedButton(
  onPressed: _submitForm,
  style: ElevatedButton.styleFrom(
    backgroundColor: Color(0xFFAA8FFF), // Grubhub style purple
    minimumSize: Size(double.infinity, 50),
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(30),
    ),
  ),
  child: Text("Save", style: TextStyle(fontSize: 16)),
),
```

```
        SizedBox(height: 20),
        Center(
          child: Text(
            "Forgot your password?",
            style: TextStyle(color: Color(0xFFAA8FFF)),
          ),
        ),
      ],
```

**Output:**




**Conclusion:** The experiment successfully demonstrated how to create an interactive form in Flutter using the Form and TextFormField widgets. With validation logic and state management, user inputs were efficiently handled, making the form both functional and user-friendly. This forms the basis for implementing real-world features like login, registration, and feedback forms in Flutter apps.