

EXPERIMENT NO: - 04

Name : Aditi Taksale

Class : D15A

Roll:No : 58

AIM: - To create an interactive Form using form widget.

Theory: -

Forms are a crucial part of mobile applications, allowing users to input and submit data efficiently. Whether it's for user registration, login, feedback collection, or other purposes, forms need to be interactive and user-friendly. In Flutter, the Form widget provides a structured way to manage multiple input fields, handle validation, and ensure proper data submission. It acts as a container for input elements and maintains their state throughout user interactions. Unlike simple text fields, the Form widget ensures that all inputs are validated before proceeding with further actions, making it an essential component in app development.

Creation of a Form in Flutter

In Flutter, creating a form involves using the Form widget, which acts as a container for multiple input fields, allowing for seamless validation and data management. Forms are crucial for gathering user input in applications such as login pages, registration forms, or feedback sections. The Form widget enables efficient handling of input fields by maintaining their state, validating user entries, and submitting data once all requirements are met.

A GlobalKey<FormState> is necessary to uniquely identify a form and access its state. This key allows developers to perform essential operations like validation, saving input, or resetting fields. Within the form, the TextFormField widget is commonly used to capture user data such as names, phone numbers, and email addresses. It provides built-in validation support, making it ideal for structured data collection. To improve user experience, InputDecoration can be applied to input fields, allowing for customization of labels, icons, borders, and placeholder text to enhance usability and aesthetics..

Key Properties of the Form Widget

- **key** – A `GlobalKey` that identifies the form and enables interaction, such as validation, saving, or resetting.
- **child** – Defines the widget that contains the form fields, typically arranged using a `Column`, `ListView`, or other layout structures.
- **autovalidateMode** – Determines when the form should automatically validate its fields, such as on input change or on submit.

Methods of the Form Widget

- **validate()** – Triggers validation for all fields in the form, returning `true` if all inputs are valid, otherwise `false`. This is used before submitting data.
- **save()** – Saves the current values of all form fields by calling the `onSaved` callback for each field. Typically used after validation succeeds.
- **reset()** – Clears all user-entered data, restoring the form to its initial state.
- **currentState** – Retrieves the current `FormState`, allowing interaction with the form programmatically.

By leveraging the **Form widget**, along with validation techniques and proper state management, developers can create **interactive and user-friendly** forms in Flutter, ensuring smooth data collection and processing.

CODE :

Sign_page.darts:

```
import 'package:flutter/material.dart';
import 'package:google_fonts/google_fonts.dart';
import 'package:booking/screens/home_screen.dart';

// Correct import

class SignInScreen extends StatefulWidget {
  @override
  _SignInScreenState createState() => _SignInScreenState();
}

class _SignInScreenState extends State<SignInScreen> {
  final TextEditingController _emailController =
    TextEditingController();

  final TextEditingController _passwordController =
    TextEditingController();

  bool _isPasswordVisible = false;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.white,
      appBar: AppBar(
        backgroundColor: const Color(0xFF003580),
        elevation: 0,
        leading: IconButton(
          icon: const Icon(Icons.close, color: Colors.white),
          onPressed: () => Navigator.pop(context),
        ),
```

```
centerTitle: true,
    title: Text(
      "Booking.com",
      style: GoogleFonts.montserrat(
        fontSize: 22,
        fontWeight: FontWeight.w600,
        color: Colors.white,
      ),
    ),
  ),
  body: Padding(
    padding: const EdgeInsets.all(20.0),
    child: Column(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: [
        Text(
          "Sign in",
          style: GoogleFonts.montserrat(
            fontSize: 22,
            fontWeight: FontWeight.bold,
            color: Colors.black,
          ),
        ),
        const SizedBox(height: 20),

        // Email Field
        Text(
          "Email address",
          style: GoogleFonts.montserrat(
            fontSize: 14,
            fontWeight: FontWeight.w500,
            color: Colors.black,
```

```

controller: _emailController,
  keyboardType: TextInputType.emailAddress,
  decoration: InputDecoration(
    hintText: "Enter your email",
    hintStyle: GoogleFonts.montserrat(color:
Colors.grey[500]),
    border: OutlineInputBorder(
      borderRadius: BorderRadius.circular(8),
      borderSide: const BorderSide(color: Colors.blue),
    ),
    focusedBorder: OutlineInputBorder(
      borderRadius: BorderRadius.circular(8),
      borderSide: const BorderSide(color:
Colors.blue, width: 2),
    ),
    contentPadding: const EdgeInsets.symmetric(
      vertical: 12,
      horizontal: 15,
    ),
  ),
),
// Sign In Button
SizedBox(
  width: double.infinity,
  height: 50,
  child: ElevatedButton(
    onPressed: () {
      String email = _emailController.text.trim();
      String password = _passwordController.text.trim();

```

```

    if (email.isEmpty || !email.contains("@")) {
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(
          content: Text("Please enter a valid email"),
        ),
      );
    } else if (password.isEmpty || password.length < 6) {
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(
          content: Text("Password must be at least 6 characters"),
        ),
      );
    } else {
      // Navigate to Home Screen after successful login
      Navigator.pushReplacement(
        context,
        MaterialPageRoute(
          builder: (context) => HomeScreen(),
        ),
      );
    }
  },
  style: ElevatedButton.styleFrom(
    backgroundColor: const Color(0xFF003580), // Booking.com Blue
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(8),
    ),
  ),
);
}
}

```

Login_page.dart :

```
import 'package:flutter/material.dart';
import 'package:google_fonts/google_fonts.dart';
import 'home_screen.dart'; // Import your home
screen

class SignUpScreen extends StatefulWidget {
  @override
  _SignUpScreenState createState() =>
  _SignUpScreenState();
}

class _SignUpScreenState extends
State<SignUpScreen> {
  final TextEditingController _emailController =
TextEditingController();
  final TextEditingController _passwordController =
TextEditingController();
  final TextEditingController
_confirmPasswordController =
  TextEditingController();
  bool _isPasswordVisible = false;
  bool _isConfirmPasswordVisible = false;

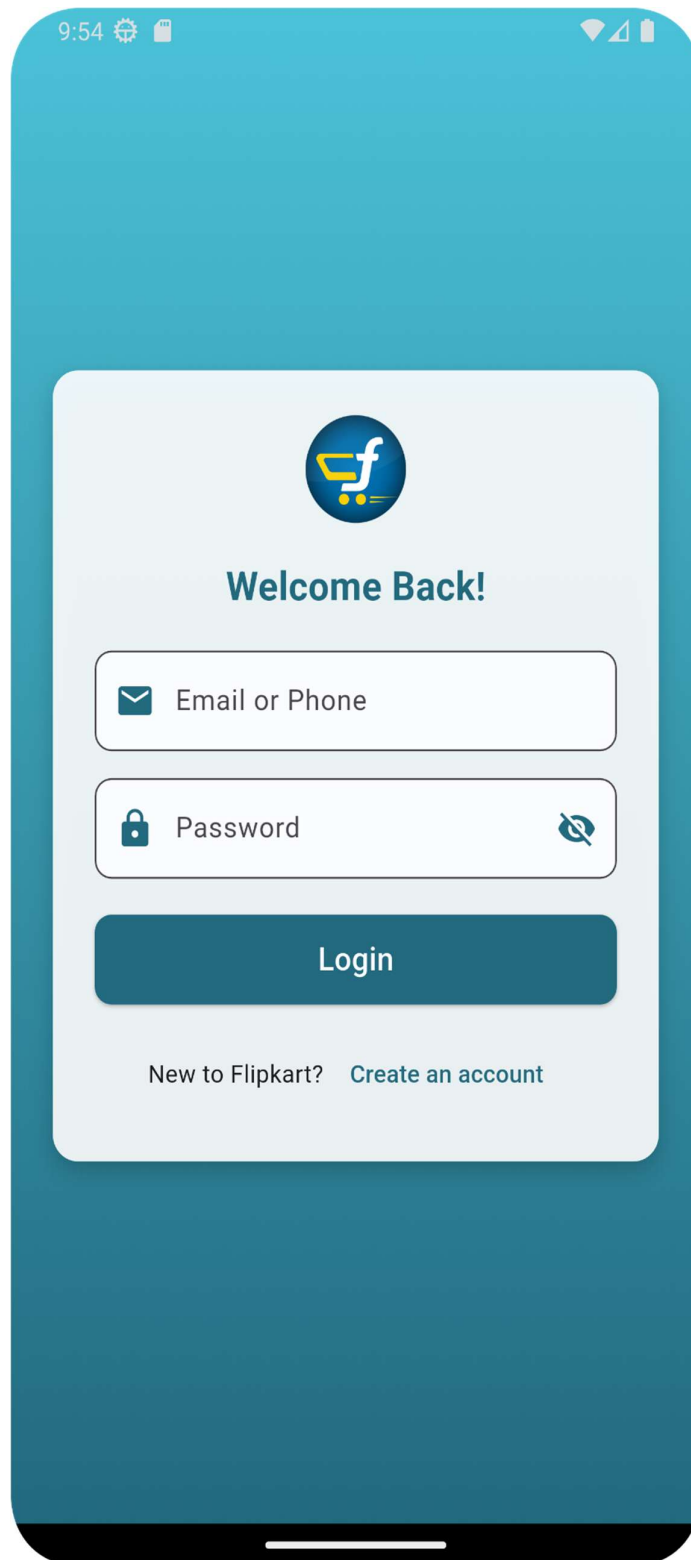
  void _register() {
    String email = _emailController.text.trim();
    String password =
_passwordController.text.trim();
    String confirmPassword =
_confirmPasswordController.text.trim();

    if (email.isEmpty || !email.contains("@")) {
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(content: Text("Please enter a
valid email")),
      );
      return;
    }
    if (password.isEmpty || password.length < 6) {
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(content: Text("Password must
be at least 6 characters")),
      );
      return;
    }
    if (confirmPassword.isEmpty) {
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(content: Text("Please confirm
your password")),
      );
      return;
    }
  }
}
```

```
const SizedBox(height: 20),
SizedBox(
  width: double.infinity,
  height: 50,
  child: ElevatedButton(
    onPressed: _register,
    style: ElevatedButton.styleFrom(
      backgroundColor: const
Color(0xFF003580),
      shape: RoundedRectangleBorder(
        borderRadius:
BorderRadius.circular(8)),
    ),
    child: Text("Register",
      style: GoogleFonts.montserrat(
        fontSize: 16,
        fontWeight: FontWeight.w400,
        color: Colors.white)),
    ),
  ),
),
),
),
);
}

InputDecoration _inputDecoration(String
hintText) {
  return InputDecoration(
    hintText: hintText,
    hintStyle: GoogleFonts.montserrat(color:
Colors.grey[500]),
    border: OutlineInputBorder(borderRadius:
BorderRadius.circular(8)),
    focusedBorder: OutlineInputBorder(
      borderRadius: BorderRadius.circular(8),
      borderSide: const BorderSide(color:
Colors.blue, width: 2)),
    contentPadding: const
EdgeInsets.symmetric(vertical: 12, horizontal: 15),
  );
}
EdgeInsets.symmetric(vertical
contentPadding: const
EdgeInsets.symmetric(vertical: 12, horizontal: 15),
suffixIcon: IconButton(
  icon: Icon(isVisible ? Icons.visibility :
Icons.visibility_off,
    color: Colors.grey),
  onPressed: toggleVisibility,
),
),
}
```

OUTPUT :





Create Account

 Full Name

 Email

 Password

Sign Up

Already have an account? [Login](#)

