EXPERIMENT NO: - 04

Name:- Manav Punjabi **Class:-** D15A **Roll:No: -** 44

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

Theory: -

A form in Flutter is a structured container that collects user input through various fields like text fields, dropdowns, checkboxes, and buttons. It plays a crucial role in applications that require user data entry, such as login pages, registration forms, and feedback submissions. Flutter provides the Form widget, which works alongside TextFormField and other input elements to manage validation, state handling, and error messages efficiently. By using form validation techniques, developers can ensure data accuracy and enhance user experience.

When you create a form, it is necessary to provide the GlobalKey. This key uniquely identifies the form and allows you to do any validation in the form fields. The form widget uses child widget TextFormField to provide the users to enter the text field. This widget renders a material design text field and also allows us to display validation errors when they occur.

Creation of a Form

- While creating a form in Flutter, the **Form widget** is essential as it acts as a container for grouping multiple form fields and managing validation.

- A **GlobalKey<FormState>** is required to uniquely identify the form and enable validation or data retrieval from the form fields.

- The **TextFormField widget** is used to provide input fields where users can enter data such as names, phone numbers, or email addresses.

- To enhance the appearance and usability of input fields, **InputDecoration** is used, allowing customization of labels, icons, borders, and hint text.

- Validation plays a crucial role in forms, and the **validator property** within **TextFormField** ensures user input meets specific criteria before submission.

- Different types of input require appropriate **keyboard types**, such as TextInputType.number for numeric fields or TextInputType.emailAddress for email fields.

- Proper **state management** is needed to store and retrieve user input, ensuring the form data is processed correctly.

- A **submit button** is necessary to trigger form validation and submit the collected data for further processing.

Some Properties of Form Widget

- **key:** A GlobalKey that uniquely identifies the Form. You can use this key to interact with the form, such as validating, resetting, or saving its state.
- **child:** The child widget that contains the form fields. Typically, this is a Column, ListView, or another widget that allows you to arrange the form fields vertically.
- **autovalidateMode:** An enum that specifies when the form should automatically validate its fields.

Some Methods of Form Widget

- **validate():** This method is used to trigger the validation of all the form fields within the Form. It returns true if all fields are valid, otherwise false. You can use it to check the overall validity of the form before submitting it.
- **save():** This method is used to save the current values of all form fields. It invokes the onSaved callback for each field. Typically, this method is called after validation succeeds.
- **reset():** Resets the form to its initial state, clearing any user-entered data.
- **currentState:** A getter that returns the current FormState associated with the Form.

Code: -

login_page.dart

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

class LoginScreen extends StatefulWidget {
  const LoginScreen({super.key});

  @override
  State<LoginScreen> createState() =>
_LoginScreenState();
}

class _LoginScreenState extends
State<LoginScreen> {
  bool isLogin = true;
  final _emailController =
TextEditingController();
  final _passwordController =
TextEditingController();

  @override
  void dispose() {
    _emailController.dispose();
    _passwordController.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: SingleChildScrollView(
          child: Padding(
            padding: const EdgeInsets.all(16.0),
            child: Column(
              mainAxisAlignment:
MainAxisAlignment.center,
              children: [
                const SizedBox(height: 50),
                // App Logo
                Container(
                  height: 100,
                  width: 100,
                  decoration: BoxDecoration(
                    color: Colors.blue,
                    borderRadius:
BorderRadius.circular(20),
                  ),
                  child: const Icon(
                    Icons.chat,
                    size: 60,
                    color: Colors.white,
                  ),
                ),
                const SizedBox(height: 30),

                // Welcome Text
                Text(
                  isLogin ? 'Welcome Back' : 'Create
Account',
                  style: const TextStyle(
                    fontSize: 28,
                    fontWeight: FontWeight.bold,
                  ),
                ),
                const SizedBox(height: 10),
                Text(
                  isLogin ? 'Login to continue' : 'Sign
up to get started',
                  style: TextStyle(
                    fontSize: 16,
                    color: Colors.grey[600],
                  ),
                ),
                const SizedBox(height: 30),

                // Email Field
                TextField(
                  controller: _emailController,
                  decoration: InputDecoration(
                    labelText: 'Email',
                    prefixIcon: const Icon(Icons.email),
                    border: OutlineInputBorder(
                      borderRadius:
BorderRadius.circular(12),
                    ),
                    enabledBorder: OutlineInputBorder(
                      borderRadius:
BorderRadius.circular(12),
                      borderSide: const
BorderSide(color: Colors.grey),
                    ),
                    focusedBorder: OutlineInputBorder(
                      borderRadius:
BorderRadius.circular(12),
                      borderSide: const
BorderSide(color: Colors.blue),
                    ),
                  ),
                  keyboardType:
TextInputType.emailAddress,
                ),
                const SizedBox(height: 16),

                // Password Field
                TextField(
                  controller: _passwordController,
                  decoration: InputDecoration(
                    labelText: 'Password',
                    prefixIcon: const Icon(Icons.lock),
                    border: OutlineInputBorder(
                      borderRadius:
BorderRadius.circular(12),
                    ),
                    enabledBorder: OutlineInputBorder(
                      borderRadius:
BorderRadius.circular(12),
                      borderSide: const
BorderSide(color: Colors.grey),
                    ),
                    focusedBorder: OutlineInputBorder(
                      borderRadius:
BorderRadius.circular(12),
                      borderSide: const
BorderSide(color: Colors.blue),
                    ),
                  ),
                  obscureText: true,
                ),
                const SizedBox(height: 20),

                // Forgot Password Button
                if (isLogin)
                  Align(
                    alignment: Alignment.centerRight,
                    child: TextButton(
                      onPressed: () {
                        // Forgot password logic
                      },
                      child: const Text('Forgot
Password?'),
                    ),
                  ),
                const SizedBox(height: 20),

                // Login/Signup Button
                SizedBox(
                  width: double.infinity,
                  height: 50,
                  child: ElevatedButton(
                    onPressed: () {
                      // Login/Signup logic ke baad
                      Navigator.pushReplacement(
                        context,
                        MaterialPageRoute(builder:
(context) => const HomeScreen()),
                      );
                    },
                    style: ElevatedButton.styleFrom(
                      shape: RoundedRectangleBorder(
                        borderRadius:
BorderRadius.circular(12),
                      ),
                    ),
                    child: Text(
                      isLogin ? 'Login' : 'Sign Up',
                      style: const TextStyle(fontSize:
16),
                    ),
                  ),
                ),
                const SizedBox(height: 20),

                // Toggle Login/Signup
                Row(
                  mainAxisAlignment:
MainAxisAlignment.center,
                  children: [
                    Text(
                      isLogin ? 'Don\'t have an account?'
: 'Already have an account?',
                      style: TextStyle(color:
Colors.grey[600]),
                    ),
                    TextButton(
                      onPressed: () {
                        setState(() {
                          isLogin = !isLogin;
                        });
                      },
                      child: Text(
                        isLogin ? 'Sign Up' : 'Login',
                        style: const TextStyle(fontWeight:
FontWeight.bold),
                      ),
                    ),
                  ],
                ),
              ],
            ),
          ),
        ),
      ),
    );
  }
```

```
}
```

signup_screen.dart

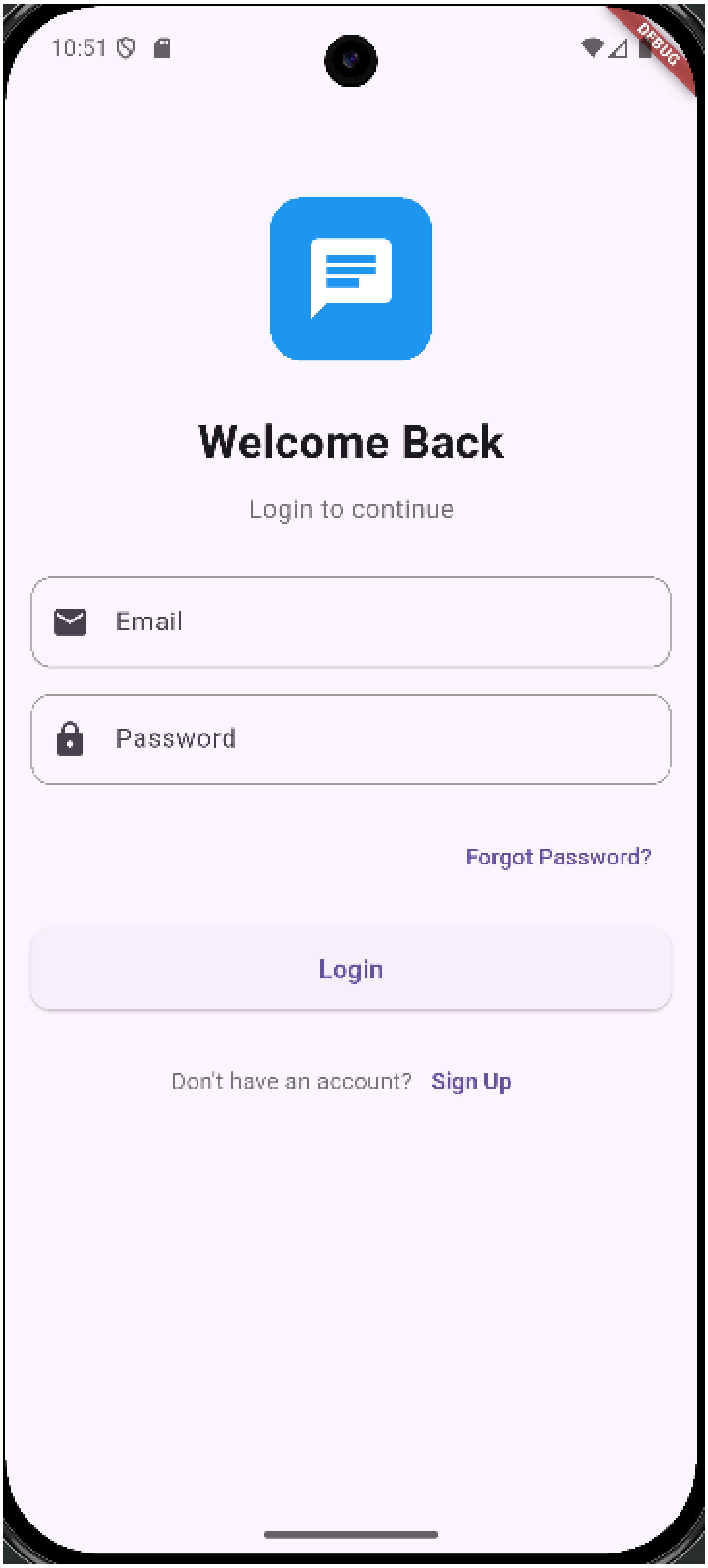
```
import 'package:flutter/material.dart';

class SignupScreen extends StatefulWidget {
  const SignupScreen({super.key});

  @override
  State<SignupScreen> createState() => _SignupScreenState();
}

class _SignupScreenState extends State<SignupScreen> {
  final _formKey = GlobalKey<FormState>();
  final _nameController = TextEditingController();
  final _emailController = TextEditingController();
  final _phoneController = TextEditingController();
  final _passwordController = TextEditingController();
  String? _selectedGender;
  DateTime? _selectedDate;
  String? _profileImage;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Create Account'),
      ),
      body: SingleChildScrollView(
        padding: const EdgeInsets.all(16),
        child: Form(
          key: _formKey,
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              Center(
                child: Stack(
                  children: [
                    CircleAvatar(
                      radius: 50,
                      backgroundColor: Theme.of(context).colorScheme.primary.withAlpha(30),
                      backgroundImage: _profileImage != null
                          ? NetworkImage(_profileImage!)
                          : null,
                      child: _profileImage == null
                          ? const Icon(Icons.person, size: 50)
                          : null,
                    ),
                    Positioned(
                      bottom: 0,
                      right: 0,
                      child: CircleAvatar(
          backgroundColor: Theme.of(context).colorScheme.primary,
          radius: 18,
          child: IconButton(
            icon: const Icon(Icons.camera_alt, size: 18),
            color: Colors.white,
            onPressed: _pickImage,
          ),
        ),
      ),
    ],
  ),
),
const SizedBox(height: 24),
TextFormField(
  controller: _nameController,
  decoration: const InputDecoration(
    labelText: 'Full Name',
    prefixIcon: Icon(Icons.person_outline),
  ),
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter your name';
    }
    return null;
  },
),
const SizedBox(height: 16),
TextFormField(
  controller: _emailController,
  decoration: const InputDecoration(
    labelText: 'Email',
    prefixIcon: Icon(Icons.email_outlined),
  ),
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter your email';
    }
    if (!value.contains('@')) {
      return 'Please enter a valid email';
    }
    return null;
  },
),
const SizedBox(height: 16),
TextFormField(
  controller: _phoneController,
  decoration: const InputDecoration(
    labelText: 'Phone Number',
    prefixIcon: Icon(Icons.phone_outlined),
  ),
  keyboardType: TextInputType.phone,
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter your phone number';
    }
    if (value.length < 10) {
      return 'Please enter a valid phone number';
    }
    return null;
  },
),
const SizedBox(height: 16),
TextFormField(
  controller: _passwordController,
  decoration: const InputDecoration(
    labelText: 'Password',
    prefixIcon: Icon(Icons.lock_outline),
  ),
  obscureText: true,
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter a password';
    }
    if (value.length < 6) {
      return 'Password must be at least 6 characters';
    }
    return null;
  },
),
const SizedBox(height: 16),
DropdownButtonFormField<String>(
  value: _selectedGender,
  decoration: const InputDecoration(
    labelText: 'Gender',
    prefixIcon: Icon(Icons.people_outline),
  ),
  items: ['Male', 'Female', 'Other'].map((gender) {
    return DropdownMenuItem(
      value: gender,
      child: Text(gender),
    );
  }).toList(),
  onChanged: (value) {
    setState(() {
      _selectedGender = value;
    });
  },
  validator: (value) {
    if (value == null) {
      return 'Please select your gender';
    }
    return null;
```

```
              },
            ),
            const SizedBox(height: 16),
            InkWell(
              onTap: _selectDate,
              child: InputDecorator(
                decoration: const InputDecoration(
                  labelText: 'Date of Birth',
                  prefixIcon: Icon(Icons.calendar_today_outlined),
                ),
                child: Text(
                  _selectedDate != null
                      ? '${_selectedDate!.day}/${_selectedDate!.month}/${_selectedDate!.year}'
                      : 'Select Date',
                ),
              ),
            ),
            const SizedBox(height: 24),
            SizedBox(
              width: double.infinity,
              height: 48,
              child: ElevatedButton(
                onPressed: _submitForm,
                child: const Text('Create Account'),
              ),
            ),
          ],
        ),
      ),
    ),
  );
}

Future<void> _pickImage() async {
  // Implement image picker
}

Future<void> _selectDate() async {
  final DateTime? picked = await showDatePicker(
    context: context,
    initialDate: DateTime.now(),
    firstDate: DateTime(1900),
    lastDate: DateTime.now(),
  );
  if (picked != null) {
    setState(() {
      _selectedDate = picked;
    });
  }
}
```

```
    void _submitForm() {
      if (_formKey.currentState!.validate()) {
        // Create user object
        final user = {
          'name': _nameController.text,
          'email': _emailController.text,
          'phone': _phoneController.text,
          'gender': _selectedGender,
          'dateOfBirth': _selectedDate?.toIso8601String(),
          'profileImage': _profileImage,
        };

        // Save user data and navigate
        print(user); // Replace with actual API call
        Navigator.pushReplacementNamed(context, '/home');
      }
    }

    @override
    void dispose() {
      _nameController.dispose();
      _emailController.dispose();
      _phoneController.dispose();
      _passwordController.dispose();
      super.dispose();
    }
  }
TextStyle(fontSize: 18)), Column(
            children: myCrops.map((crop) => Text(crop)).toList(),
          ),
        ],
      ),
    );
  }

  Widget _buildTextField({ required String label, required String initialValue,
    required Function(String?) onSaved,
  }) {
    return TextFormField( initialValue: initialValue, decoration: InputDecoration(
      labelText: label,
        border: OutlineInputBorder(),
      ),
      onSaved: onSaved,

    );
  }
}
```

Manav - D15A /44

OUTPUT: -



10:51

DEBUG

Welcome Back

Login to continue

Email

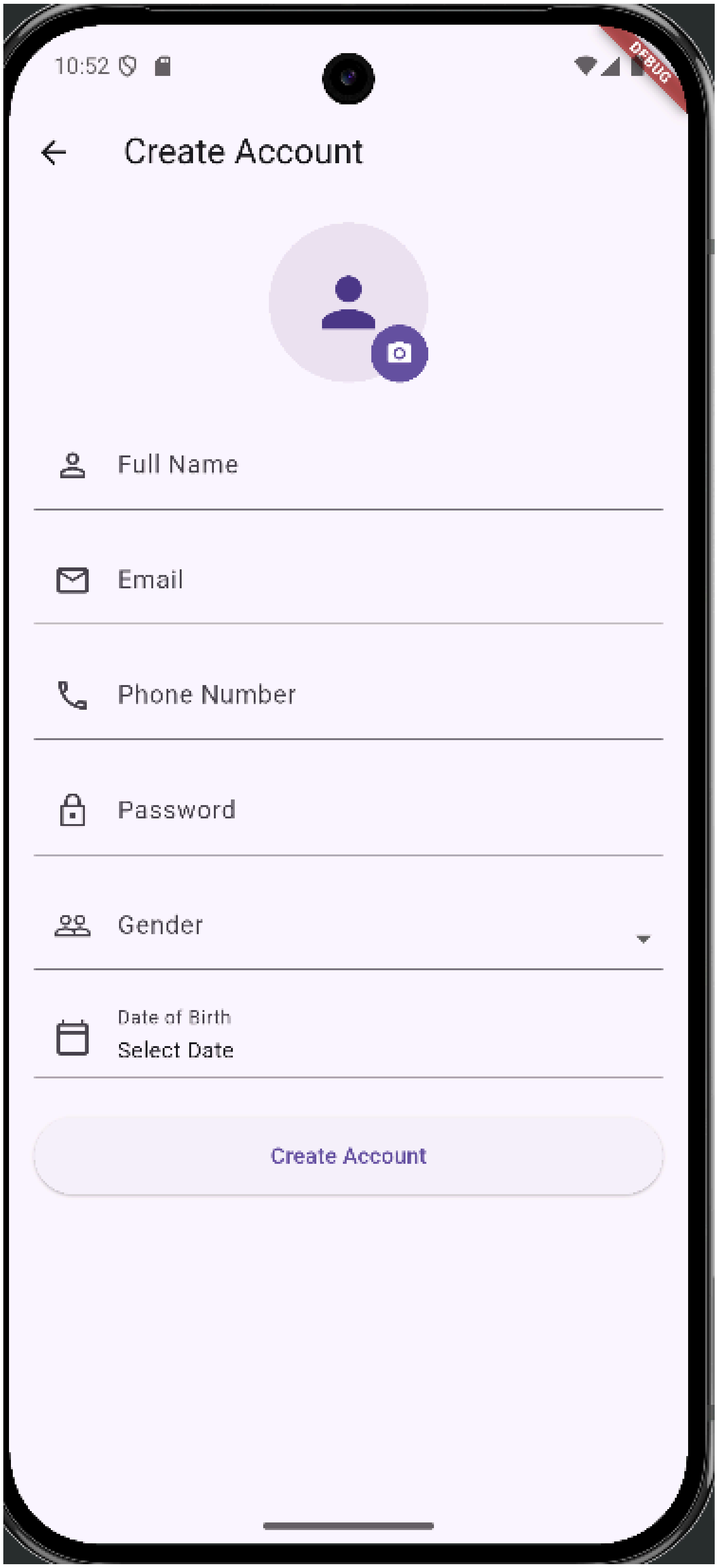
Password

Forgot Password?

Login

Don't have an account? Sign Up

Manav - D15A /44



Create Account

Full Name

Email

Phone Number

Password

Gender

Date of Birth
Select Date

Create Account