

EXPERIMENT 4

AIM:To create an interactive Form using form widget

THEORY:**Form Widget:**

It groups multiple input fields like `TextFormField` and manages their state together. It helps in validating input and prevents submission if any field is invalid.

TextFormField:

Used for text input. Supports validation, styling, and works with controllers to read and manage user data.

Validation and Submission:

You can add `validator` functions to check conditions like empty or wrongly formatted input. Form submission is typically done using an `ElevatedButton` or `TextButton`.

GlobalKey:

A `GlobalKey<FormState>` is used to manage the form's state, perform validation, and reset the form

CODE:

```
import 'package:flutter/material.dart';
import 'dashboard.dart'; // Import Dashboard for navigation

class DonatePage extends StatefulWidget {
  // Static list to store donors across app sessions
  static List<Map<String, String>> donors = [];

  @override
  _DonatePageState createState() => _DonatePageState();
}

class _DonatePageState extends State<DonatePage> {
  final _formKey = GlobalKey<FormState>();

  // Controllers for form fields
  final TextEditingController nameController = TextEditingController();
  final TextEditingController ageController = TextEditingController();
```

```
final TextEditingController areaController = TextEditingController();
final TextEditingController contactController = TextEditingController();
String? selectedBloodGroup;

// List of blood groups
final List<String> bloodGroups = ['A+', 'A-', 'B+', 'B-', 'O+', 'O-', 'AB+', 'AB-'];

void _submitForm() {
  if (_formKey.currentState!.validate()) {
    // Store form data in the static donors list
    DonatePage.donors.add({
      'name': nameController.text,
      'age': ageController.text,
      'bloodGroup': selectedBloodGroup!,
      'area': areaController.text,
      'contact': contactController.text,
    });

    // Show success dialog
    showDialog(
      context: context,
      builder: (BuildContext context) {
        return AlertDialog(
          title: Text('Thank You!'),
          content: Text('Your donation information has been submitted successfully.'),
          actions: <Widget>[
            TextButton(
              child: Text('OK'),
              onPressed: () {
                // Close the dialog
                Navigator.of(context).pop();
                // Navigate back to Dashboard
                Navigator.of(context).pushAndRemoveUntil(
                  MaterialPageRoute(builder: (context) => Dashboard(uid: 'uid')),
                  (Route<dynamic> route) => false
                );
              },
            ),
          ],
        );
      },
    );
  }
}
```

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text("Donate Blood"),
      backgroundColor: Colors.red,
    ),
    body: Form(
      key: _formKey,
      child: SingleChildScrollView(
        child: Padding(
          padding: const EdgeInsets.all(16.0),
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.stretch,
            children: [
              // Header
              Center(
                child: Column(
                  children: const [
                    Icon(Icons.bloodtype, size: 80, color: Colors.red),
                    SizedBox(height: 10),
                    Text(
                      "Donate Blood, Save Lives",
                      style: TextStyle(
                        fontSize: 24,
                        fontWeight: FontWeight.bold,
                        color: Colors.redAccent,
                      ),
                    ),
                  ],
                ),
              ),
              const SizedBox(height: 20),

              // Name Field
              TextFormField(
                controller: nameController,
                decoration: InputDecoration(
                  labelText: "Full Name",
                  border: OutlineInputBorder(borderRadius: BorderRadius.circular(10)),
                  prefixIcon: const Icon(Icons.person, color: Colors.red),
                ),
                validator: (value) {
```

```
        if (value == null || value.isEmpty) {
            return "Please enter your name";
        }
        return null;
    },
),
const SizedBox(height: 16),

// Age Field
TextFormField(
  controller: ageController,
  keyboardType: TextInputType.number,
  decoration: InputDecoration(
    labelText: "Age",
    border: OutlineInputBorder(borderRadius: BorderRadius.circular(10)),
    prefixIcon: const Icon(Icons.cake, color: Colors.red),
  ),
  validator: (value) {
    if (value == null || value.isEmpty) {
      return "Please enter your age";
    }
    if (int.tryParse(value) == null || int.parse(value) < 18) {
      return "You must be at least 18 years old";
    }
    return null;
  },
),
const SizedBox(height: 16),


// Blood Group Dropdown
DropdownButtonFormField<String>(
  decoration: InputDecoration(
    labelText: "Blood Group",
    border: OutlineInputBorder(borderRadius: BorderRadius.circular(10)),
    prefixIcon: const Icon(Icons.bloodtype, color: Colors.red),
  ),
  value: selectedBloodGroup,
  onChanged: (value) {
    setState(() {
      selectedBloodGroup = value;
    });
  },
  validator: (value) => value == null ? "Please select your blood group" : null,
  items: bloodGroups.map((group) {
```


```
        return DropdownMenuItem<String>(  
          value: group,  
          child: Text(group),  
        );  
      }).toList(),  
    ),  
    const SizedBox(height: 16),  
  
    // Area Field  
    TextFormField(  
      controller: areaController,  
      decoration: InputDecoration(  
        labelText: "Area / City",  
        border: OutlineInputBorder(borderRadius: BorderRadius.circular(10)),  
        prefixIcon: const Icon(Icons.location_on, color: Colors.red),  
      ),  
      validator: (value) {  
        if (value == null || value.isEmpty) {  
          return "Please enter your area";  
        }  
        return null;  
      },  
    ),  
    const SizedBox(height: 16),  
  
    // Contact Number Field  
    TextFormField(  
      controller: contactController,  
      keyboardType: TextInputType.phone,  
      decoration: InputDecoration(  
        labelText: "Contact Number",  
        border: OutlineInputBorder(borderRadius: BorderRadius.circular(10)),  
        prefixIcon: const Icon(Icons.phone, color: Colors.red),  
      ),  
      validator: (value) {  
        if (value == null || value.isEmpty) {  
          return "Please enter your contact number";  
        }  
        if (value.length != 10) {  
          return "Enter a valid 10-digit phone number";  
        }  
        return null;  
      },  
    ),  
  ),  
);
```

```
const SizedBox(height: 24),

// Submit Button
ElevatedButton(
  style: ElevatedButton.styleFrom(
    backgroundColor: Colors.red,
    padding: const EdgeInsets.symmetric(vertical: 15),
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(30),
    ),
  ),
  onPressed: _submitForm,
  child: const Text(
    "Submit",
    style: TextStyle(fontSize: 18),
  ),
),
],
),
),
),
),
);
}
```


OUTPUT :

 **Donate Blood**




Donate Blood, Save Lives



Full Name

 atharva


Age

 20


Blood Group

 B+ 

Area / City

 Mumbai

Contact Number

 8455869878

Submit