Experiment – 3: Creation of Forms for mobile Apps.

Aim: To create interactive form using form widget

Objectives: After study of this experiment, the student will be able to

- Develop the App UI by incorporating form widget

Outcomes: After study of this experiment, the student will be able to

- Design and Develop interactive Flutter App by using widgets (L604.3)

Prerequisite: Dart Programming Language

Requirements: Android Studio, Flutter framework, Internet Connection.

Pre-Experiment Exercise:
Brief Theory:

- The Form widget is an optional container for grouping together multiple form field widgets.
- The benefit of using a Form widget is to validate each text field as a group. ● You can group TextFormField widgets to manually or automatically validate them. ● The TextFormField widget wraps a TextField widget to provide validation when enclosed in a Form widget.
- If all text fields pass the FormState validate method, then it returns true. If any text fields contain errors, it displays the appropriate error message for each text field, and the FormState validate method returns false. This process gives you the ability to use FormState to check for any validation errors instead of checking each text field for errors and not allowing the posting of invalid data.
- The Form widget needs a unique key to identify it and is created by using GlobalKey. This GlobalKey value is unique across the entire app.
- We can create and validate a form using the following steps:
  1. Create a Form with a GlobalKey.
  2. Add a TextFormField with validation logic.
  3. Create a button to validate and submit the form.

Laboratory Exercise

A. Program

1. Create a form for the mobile app having two text fields: person's name and contact number. The form should be able to accept the input when the user submits the form. Also validate the form for null values and display appropriate messages.

B. Result/Observation

1. Print out of program code and output.

Post-Experiments Exercise

A. Questions:

1. Add one more text field along with validation, and show the output. Also validate the 'phone number' field to ensure that only numeric value is accepted in the 'phone number' field.

B. Conclusion:

1. Write what you have learnt in the experiment.

C. References:

1. https://api.flutter.dev/flutter/widgets/Form-class.html
2. https://docs.flutter.dev/cookbook/forms/validation

Name: Vishal Rajesh Mahajan
Class: TE IT A

MAD PWA EXP 03
Roll No: 56

Laboratory Exercise

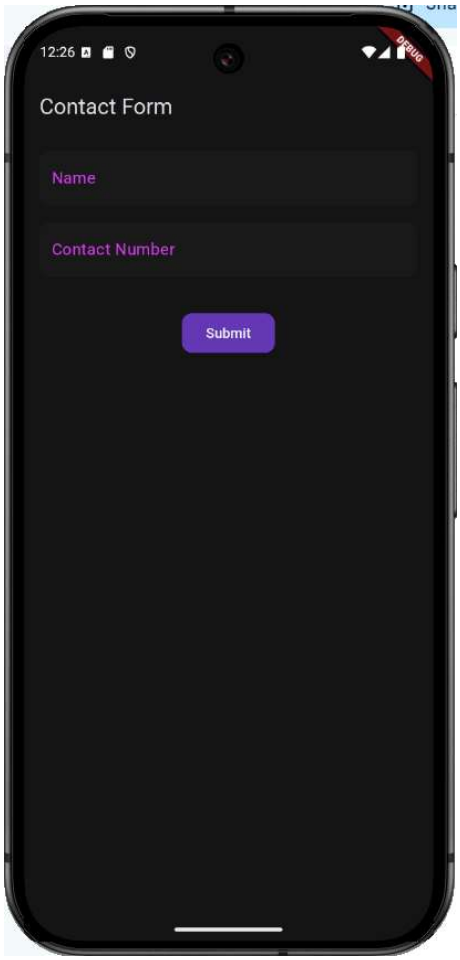
Create a form for the mobile app having two text fields: person's name and contact number. The form should be able to accept the input when the user submits the form. Also validate the form for null values and display appropriate messages.

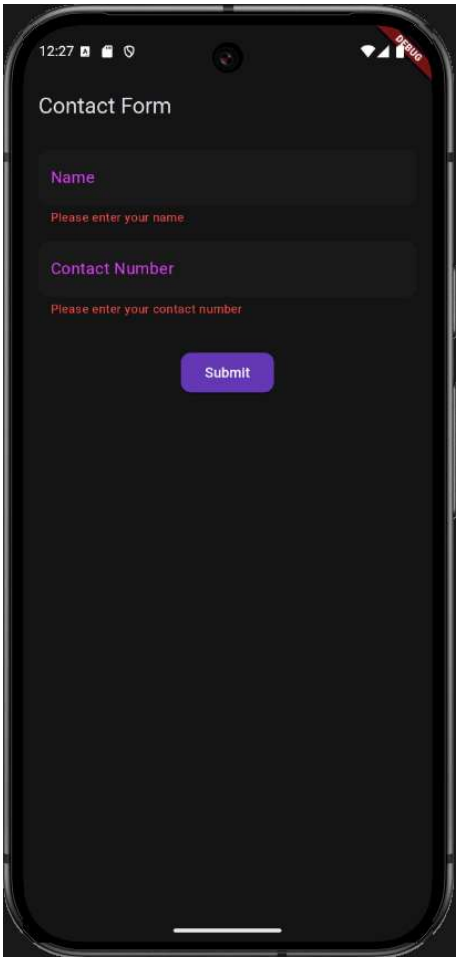
```dart
import 'package:flutter/material.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Modern Form App',
      theme: ThemeData(
        brightness: Brightness.dark,
        primaryColor: Colors.deepPurple,
        hintColor: Colors.purpleAccent,
        scaffoldBackgroundColor: Color(0xFF121212),
        inputDecorationTheme: InputDecorationTheme(
          filled: true,
          fillColor: Color(0xFF1E1E1E),
          border: OutlineInputBorder(
            borderRadius: BorderRadius.circular(10.0),
            borderSide: BorderSide.none,
          ),
          focusedBorder: OutlineInputBorder(
            borderRadius: BorderRadius.circular(10.0),
            borderSide: BorderSide(
              color: Colors.purpleAccent,
              width: 2.0,
            ),
          ),
          labelStyle: TextStyle(color: Colors.purpleAccent),
          errorStyle: TextStyle(color: Colors.redAccent),
        ),
```

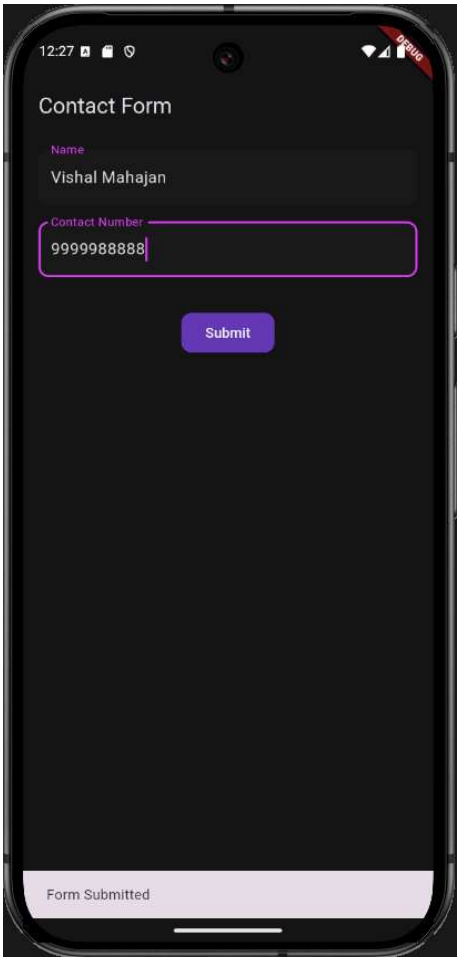
```
        textSelectionTheme: TextSelectionThemeData(
          cursorColor: Colors.purpleAccent,
          selectionColor: Colors.purple,
          selectionHandleColor: Colors.purpleAccent,
        ),
        elevatedButtonTheme: ElevatedButtonThemeData(
          style: ElevatedButton.styleFrom(
            foregroundColor: Colors.white,
            backgroundColor: Colors.deepPurple,
            shape: RoundedRectangleBorder(
              borderRadius: BorderRadius.circular(10.0),
            ),
          ),
        ),
      ),
      home: FormScreen(),
    );
  }
}

class FormScreen extends StatefulWidget {
  @override
  _FormScreenState createState() => _FormScreenState();
}

class _FormScreenState extends State<FormScreen> {
  final _formKey = GlobalKey<FormState>();
  final _nameController = TextEditingController();
  final _contactController = TextEditingController();

  @override
  void dispose() {
    _nameController.dispose();
    _contactController.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Contact Form'),
```

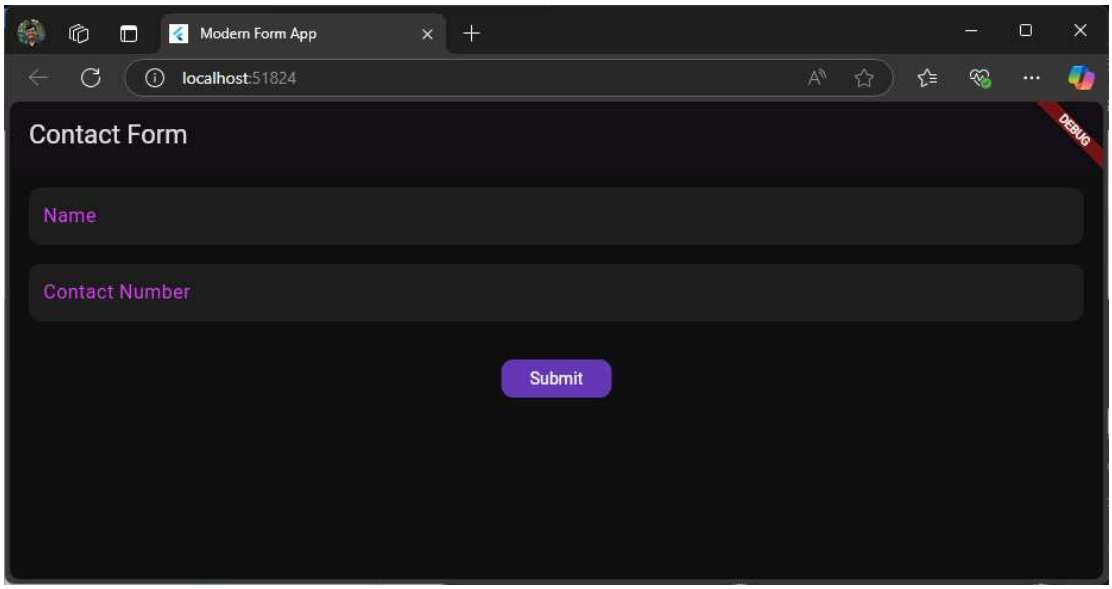
```
    ),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Form(
        key: _formKey,
        child: Column(
          children: [
            TextFormField(
              controller: _nameController,
              decoration: InputDecoration(
                labelText: 'Name',
              ),
              validator: (value) {
                if (value == null || value.isEmpty) {
                  return 'Please enter your name';
                }
                return null;
              },
            ),
            SizedBox(height: 16.0),
            TextFormField(
              controller: _contactController,
              decoration: InputDecoration(
                labelText: 'Contact Number',
              ),
              keyboardType: TextInputType.phone,
              validator: (value) {
                if (value == null || value.isEmpty) {
                  return 'Please enter your contact number';
                }
                return null;
              },
            ),
            SizedBox(height: 32.0),
            ElevatedButton(
              onPressed: () {
                if (_formKey.currentState!.validate()) {
                  ScaffoldMessenger.of(context).showSnackBar(
                    SnackBar(content: Text('Form Submitted')),
                  );
                }
              },
```

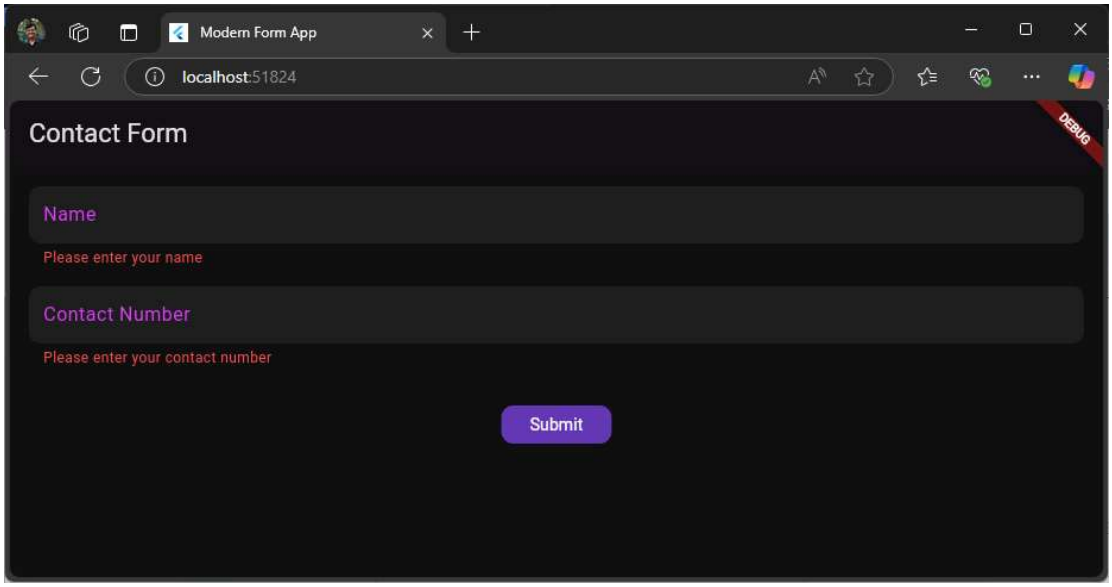
```
                child: Text('Submit'),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

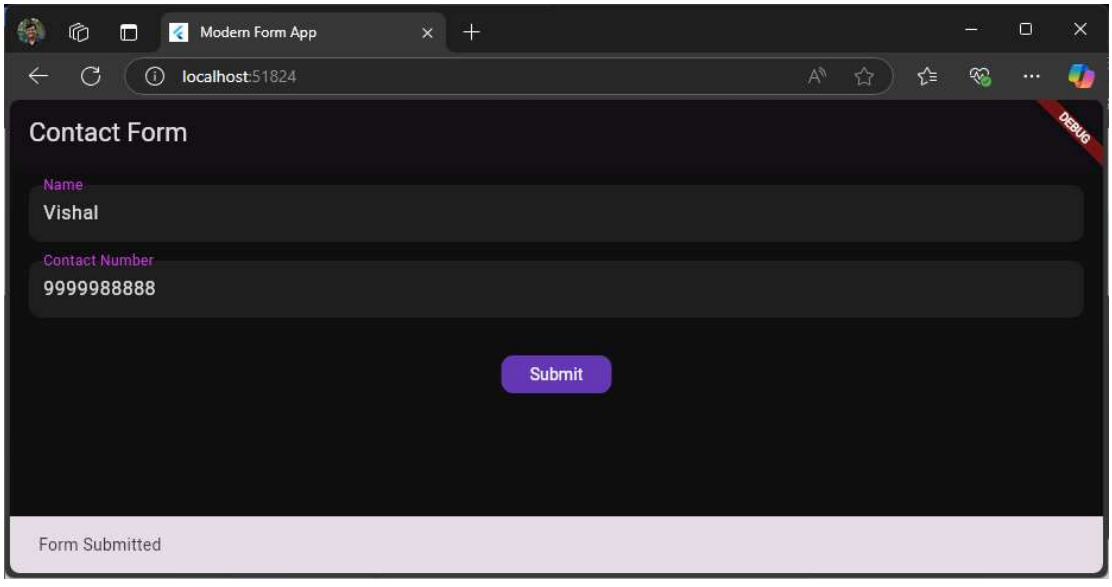







Modern Form App
localhost:51824
Contact Form
DEBUG
Name
Contact Number
Submit


Modern Form App
localhost:51824
Contact Form
DEBUG
Name
Please enter your name
Contact Number
Please enter your contact number
Submit


Modern Form App
localhost:51824
Contact Form
DEBUG
Name
Vishal
Contact Number
9999988888
Submit
Form Submitted

Post Experiment Exercise

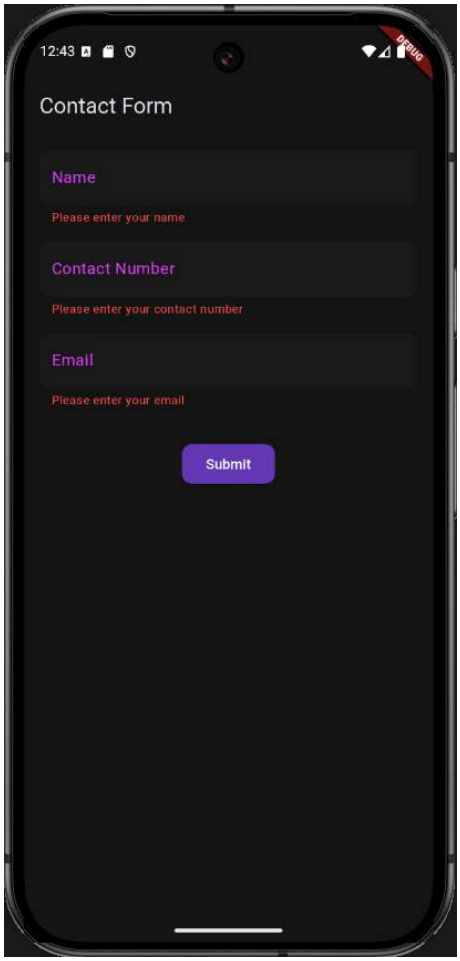
Add one more text field along with validation, and show the output. Also validate the ‘phone number’ field to ensure that only numeric value is accepted in the ‘phone number’ field.

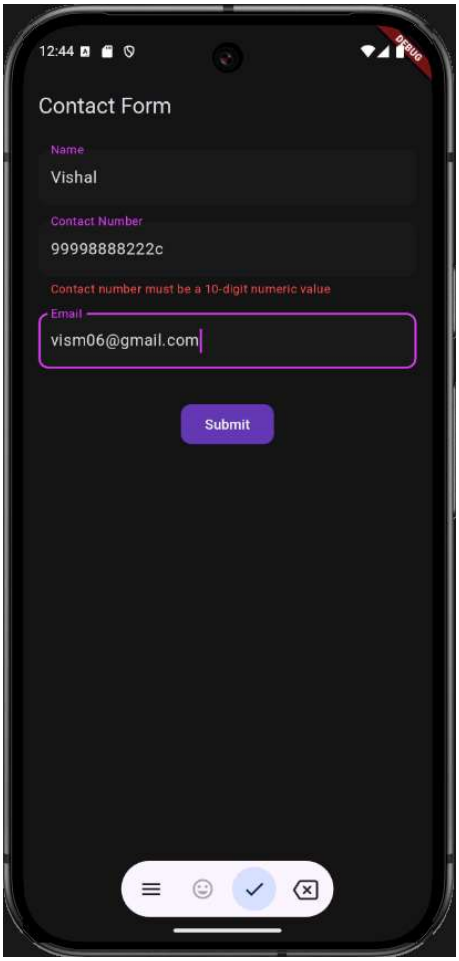
```dart
import 'package:flutter/material.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Modern Form App',
      theme: ThemeData(
        brightness: Brightness.dark,
        primaryColor: Colors.deepPurple,
        hintColor: Colors.purpleAccent,
        scaffoldBackgroundColor: Color(0xFF121212),
        inputDecorationTheme: InputDecorationTheme(
          filled: true,
          fillColor: Color(0xFF1E1E1E),
          border: OutlineInputBorder(
            borderRadius: BorderRadius.circular(10.0),
            borderSide: BorderSide.none,
          ),
          focusedBorder: OutlineInputBorder(
            borderRadius: BorderRadius.circular(10.0),
            borderSide: BorderSide(
              color: Colors.purpleAccent,
              width: 2.0,
            ),
          ),
          labelStyle: TextStyle(color: Colors.purpleAccent),
          errorStyle: TextStyle(color: Colors.redAccent),
        ),
        textSelectionTheme: TextSelectionThemeData(
          cursorColor: Colors.purpleAccent,
          selectionColor: Colors.purple,
          selectionHandleColor: Colors.purpleAccent,
        ),
        elevatedButtonTheme: ElevatedButtonThemeData(
          style: ElevatedButton.styleFrom(
```

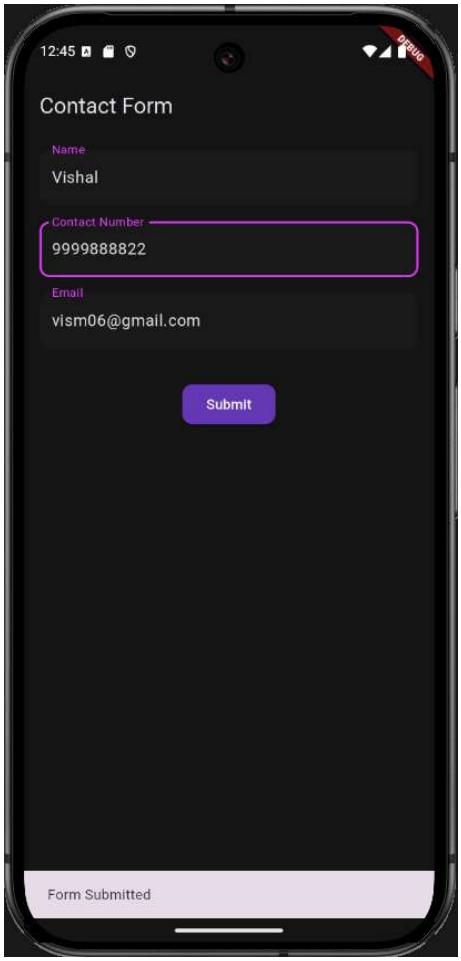
```
            foregroundColor: Colors.white,
            backgroundColor: Colors.deepPurple,
            shape: RoundedRectangleBorder(
              borderRadius: BorderRadius.circular(10.0),
            ),
          ),
        ),
      ),
      home: FormScreen(),
    );
  }
}

class FormScreen extends StatefulWidget {
  @override
  _FormScreenState createState() => _FormScreenState();
}

class _FormScreenState extends State<FormScreen> {
  final _formKey = GlobalKey<FormState>();
  final _nameController = TextEditingController();
  final _contactController = TextEditingController();
  final _emailController = TextEditingController();

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Contact Form'),
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Form(
          key: _formKey,
          child: Column(
            children: [
              TextFormField(
                controller: _nameController,
```

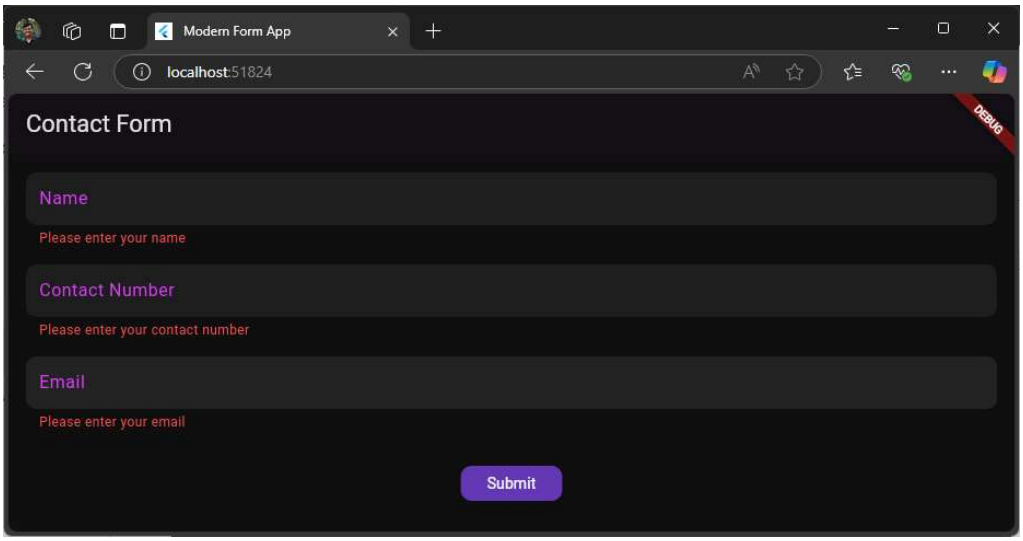
```
            decoration: InputDecoration(
              labelText: 'Name',
            ),
            validator: (value) {
              if (value == null || value.isEmpty) {
                return 'Please enter your name';
              }
              return null;
            },
          ),
          SizedBox(height: 16.0),
          TextFormField(
            controller: _contactController,
            decoration: InputDecoration(
              labelText: 'Contact Number',
            ),
            keyboardType: TextInputType.phone,
            validator: (value) {
              if (value == null || value.isEmpty) {
                return 'Please enter your contact number';
              } else if (!RegExp(r'^[0-9]{10}$').hasMatch(value)) {
                return 'Contact number must be a 10-digit numeric value';
              }
              return null;
            },
          ),
          SizedBox(height: 16.0),
          TextFormField(
            controller: _emailController,
            decoration: InputDecoration(
              labelText: 'Email',
            ),
            keyboardType: TextInputType.emailAddress,
            validator: (value) {
              if (value == null || value.isEmpty) {
                return 'Please enter your email';
              } else if (!RegExp(
                      r'^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$')
                  .hasMatch(value)) {
                return 'Please enter a valid email address';
              }
              return null;
            },
          ),
          SizedBox(height: 32.0),
          ElevatedButton(
```

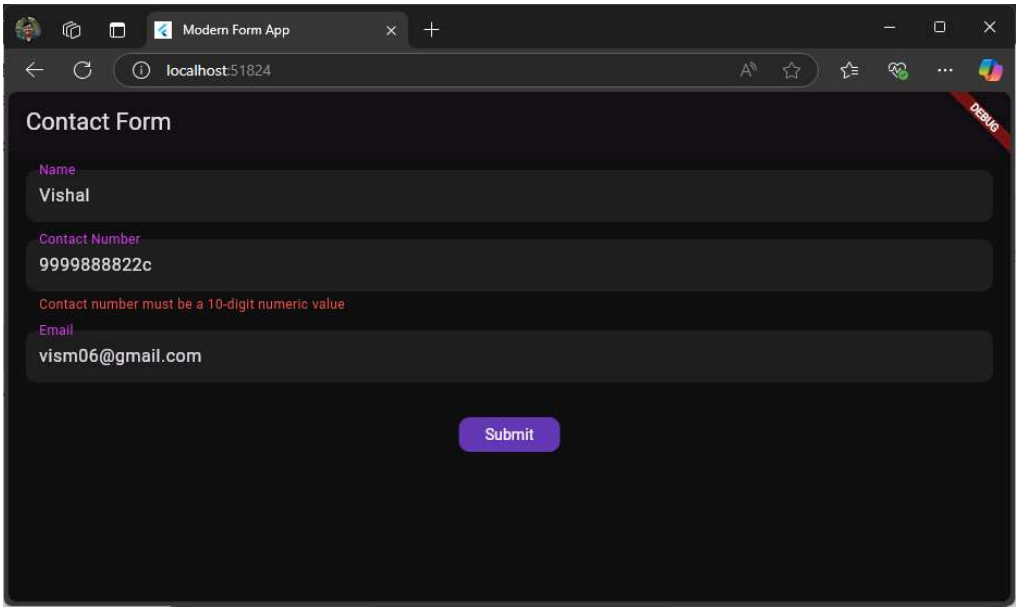
```
                onPressed: () {
                  if (_formKey.currentState!.validate()) {
                    ScaffoldMessenger.of(context).showSnackBar(
                      SnackBar(content: Text('Form Submitted')),
                    );
                  }
                },
                child: Text('Submit'),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

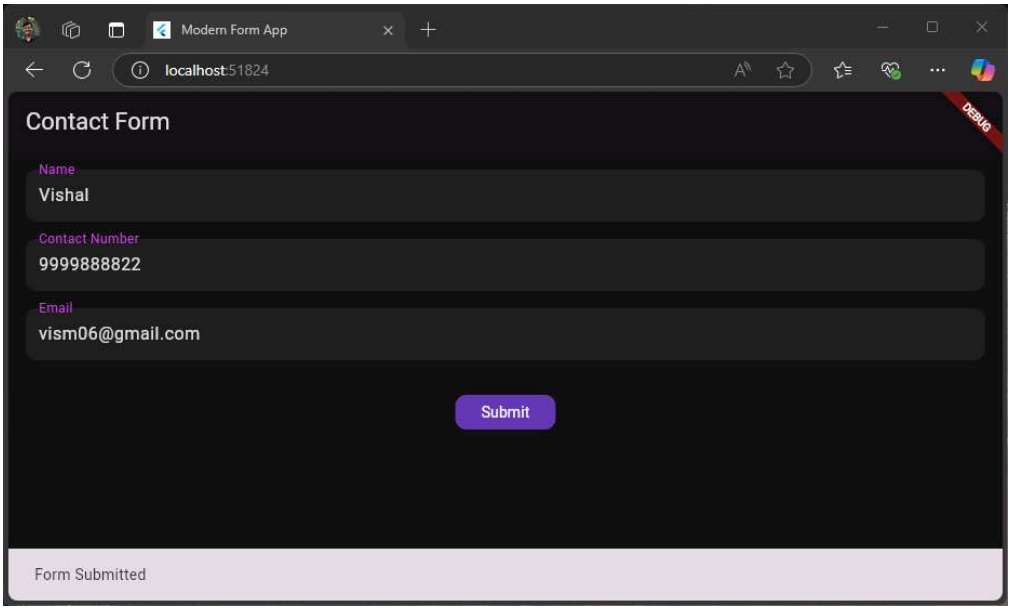







Modern Form App
localhost:51824
Contact Form
DEBUG
Name
Please enter your name
Contact Number
Please enter your contact number
Email
Please enter your email
Submit


Modern Form App
localhost:51824
Contact Form
DEBUG
Name
Vishal
Contact Number
9999888822c
Contact number must be a 10-digit numeric value
Email
vism06@gmail.com
Submit


Modern Form App
localhost:51824
Contact Form
DEBUG
Name
Vishal
Contact Number
9999888822
Email
vism06@gmail.com
Submit
Form Submitted