

**Name: - Anmol Vaswani**

**Div: -D15A**

**Roll No.: - 67**

## **Experiment-4**

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

### **Theory: -**

Form Widget:

- In Flutter, a Form widget is used to group multiple form fields together.
- It provides functionality for managing form state, validation, and submission.
- The Form widget keeps track of the form's state using a Form State object.

Form Fields:

- Form fields represent input elements within the form, such as text fields, checkboxes, radio buttons, dropdowns, etc.
- Each form field typically has properties for input validation and handling user input changes.

Form State Management:

- The Form State object is responsible for managing the state of a form.
- It provides methods for validating the form, saving form data, resetting the form, and more.

User Interaction:

- User interaction with the form involves entering data into form fields, interacting with input elements, and submitting the form.

### **The Code is of login\_page.dart:**

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

import 'package:unacademy_app/home_page.dart';

class LoginPage extends StatefulWidget {
  const LoginPage({Key? key}) : super(key: key);

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

class _LoginPageState extends State<LoginPage> {
  String? _selectedCountry;
  String? _phoneNumber;
  final GlobalKey<FormState> _formKey = GlobalKey<FormState>();

  final List<String> countries = [
    'AF Afghanistan - +93',
    'AM Armenia - +374',
    'AZ Azerbaijan - +994',
    'BH Bahrain - +973',
    'BD Bangladesh - +880',
    'BT Bhutan - +975',
    'IO British Indian Ocean Territory - +246',
  ];
}
```

'BN Brunei - +673',  
'KH Cambodia - +855',  
'CN China - +86',  
'CY Cyprus - +357',  
'EG Egypt - +20',  
'GE Georgia - +995',  
'HK Hong Kong - +852',  
'IN India - +91',  
'ID Indonesia - +62',  
'IR Iran - +98',  
'IQ Iraq - +964',  
'IL Israel - +972',  
'JP Japan - +81',  
'JO Jordan - +962',  
'KZ Kazakhstan - +7',  
'KP North Korea - +850',  
'KR South Korea - +82',  
'KW Kuwait - +965',  
'KG Kyrgyzstan - +996',  
'LA Laos - +856',  
'LB Lebanon - +961',  
'MO Macau - +853',  
'MY Malaysia - +60',  
'MV Maldives - +960',  
'MN Mongolia - +976',  
'MM Myanmar - +95',  
'NP Nepal - +977',  
'OM Oman - +968',  
'PK Pakistan - +92',  
'PS State of Palestine - +970',  
'PH Philippines - +63',  
'QA Qatar - +974',  
'RU Russia - +7',  
'SA Saudi Arabia - +966',  
'SG Singapore - +65',  
'LK Sri Lanka - +94',  
'SY Syria - +963',  
'TW Taiwan - +886',  
'TJ Tajikistan - +992',  
'TH Thailand - +66',  
'TL Timor-Leste - +670',  
'TR Turkey - +90',  
'TM Turkmenistan - +993',  
'AE United Arab Emirates - +971',  
'UZ Uzbekistan - +998',  
'VN Vietnam - +84',  
'YE Yemen - +967',  
'AB Abkhazia - +7 840',

```

'cx Christmas Island - +61',
'cc Cocos (Keeling) Islands - +61',
'Az Nagorno-Karabakh - +374',
'cy Northern Cyprus - +90 392',
'GE South Ossetia - +995 34',
];

@Override
Widget build(BuildContext context) {
  return Scaffold(
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Form(
        key: _formKey,
        child: Stack(
          children: [
            Column(
              mainAxisAlignment: MainAxisAlignment.start,
              children: [
                // Upper Logo and Heading
                Column(
                  mainAxisAlignment: MainAxisAlignment.start,
                  children: [
                    Image.asset('assets/images/your_logo.png', height: 60, width: 60),
                    const SizedBox(height: 16),
                    RichText(
                      text: TextSpan(
                        style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold, color: Colors.black),
                        children: [
                          TextSpan(text: 'Welcome to\n', style: TextStyle(fontSize: 25, fontWeight: FontWeight.bold, color: Colors.black)),
                          TextSpan(text: 'Unacademy\n', style: TextStyle(fontSize: 25, fontWeight: FontWeight.bold, color: Colors.black)),
                          TextSpan(text: 'Over', style: TextStyle(fontSize: 15, color: Colors.grey)),
                          TextSpan(text: '10 crore', style: TextStyle(fontSize: 15, fontWeight: FontWeight.bold, color: Colors.green)), // Highlight "10 crore" in green
                          TextSpan(text: ' learners trust\n', style: TextStyle(fontSize: 15, color: Colors.grey)), // Grey color for "learners trust"
                          TextSpan(text: 'us for online and offline\n', style: TextStyle(fontSize: 15, color: Colors.grey)), // Grey color for "us for online and offline"
                          TextSpan(text: 'coaching', style: TextStyle(fontSize: 15, color: Colors.grey)), // Grey color for "coaching"
                        ],
                      ),
                    ),
                  ],
                ),
              ],
            ),
            const SizedBox(height: 16),
            // Country Selection
            DropdownButton<String>(
              value: _selectedCountry,
              onChanged: (String? newValue) {
                setState(() {

```

```

        _selectedCountry = newValue;
    });
},
items: countries.map<DropdownMenuItem<String>>((String value) {
    return DropdownMenuItem<String>(
        value: value,
        child: Text(value),
    );
}).toList(),
selectedItemBuilder: (BuildContext context) {
    return countries.map<Widget>((String item) {
        final String flag = item.split(' ')[0];
        final String countryName = item.substring(item.indexOf(' ') + 1);
        return Row(
            children: [
                Text(flag),
                const SizedBox(width: 8),
                Text(countryName),
            ],
        );
    }).toList();
},
hint: Text('Select your country'),
),
const SizedBox(height: 16),

// Mobile Number Input
 TextFormField(
    keyboardType: TextInputType.phone,
    decoration: InputDecoration(
        labelText: 'Enter your mobile number',
    ),
    validator: (value) {
        if (value == null || value.isEmpty) {
            return 'Please enter your mobile number';
        }
        if (value.length != 10) {
            return 'Mobile number must be exactly 10 digits';
        }
        if (!RegExp(r'^[0-9]+\$').hasMatch(value)) {
            return 'Please enter only numeric digits';
        }
        return null;
    },
    onSaved: (newValue) {
        _phoneNumber = newValue;
    },
),
const SizedBox(height: 16),

// Continue Button at the bottom
 Expanded(
    child: Align(

```

```
        alignment: Alignment.bottomCenter,
        child: ElevatedButton(
            onPressed: () {
                if (_selectedCountry == null) {
                    ScaffoldMessenger.of(context).showSnackBar(
                        SnackBar(
                            content: Text('Please select your country'),
                            backgroundColor: Colors.red,
                        ),
                    );
                } else if (_formKey.currentState!.validate()) {
                    _formKey.currentState!.save();
                    // Handle continue button press
                    // Navigate to the next screen or perform necessary actions
                    Navigator.push(
                        context,
                        MaterialPageRoute(
                            builder: (context) => VerifyOTPPage(
                                phoneNumber: _phoneNumber ?? '',
                                email: '',
                            ),
                        ),
                    );
                }
            },
            child: Text('Continue'),
        ),
    ],
),
Positioned(
    top: 0,
    right: 0,
    child: InkWell(
        onTap: () {
            Navigator.push(
                context,
                MaterialPageRoute(builder: (context) => LoginWithEmailPage()),
            );
        },
        child: Padding(
            padding: const EdgeInsets.all(8.0),
            child: Text(
                'LOGIN WITH EMAIL>',
                style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
            ),
        ),
    ),
),
],
),
),
),
);
});
```

```
}

class LoginWithEmailPage extends StatefulWidget {
  @override
  _LoginWithEmailPageState createState() => _LoginWithEmailPageState();
}

class _LoginWithEmailPageState extends State<LoginWithEmailPage> {
  final TextEditingController _emailController = TextEditingController();
  final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
  bool _emailValidated = true;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Your email address'),
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Form(
          key: _formKey,
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              Text(
                'We\'ll send an OTP for verification',
                style: TextStyle(fontSize: 16),
              ),
              SizedBox(height: 16),
              TextFormField(
                controller: _emailController,
                keyboardType: TextInputType.emailAddress,
                decoration: InputDecoration(
                  labelText: 'Enter your email address',
                  errorText: _emailValidated ? null : 'This email address is invalid',
                ),
                validator: (value) {
                  if (value == null || value.isEmpty) {
                    return 'This field is required';
                  }
                  if (!value.contains('@gmail.com')) {
                    setState(() {
                      _emailValidated = false;
                    });
                    return null;
                  } else {
                    setState(() {
                      _emailValidated = true;
                    });
                    return null;
                  }
                },
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

```

        SizedBox(height: 10),
        Row(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
                TextButton(
                    onPressed: () {
                        Navigator.pop(context);
                    },
                    child: Text('LOGIN WITH MOBILE NUMBER >'),
                ),
            ],
        ),
        SizedBox(height: 16),
        Expanded(
            child: Align(
                alignment: Alignment.bottomCenter,
                child: ElevatedButton(
                    onPressed: () {
                        if (_formKey.currentState!.validate()) {
                            _formKey.currentState!.save();
                            // Handle continue button press
                            // Navigate to the next screen or perform necessary actions
                            Navigator.push(
                                context,
                                MaterialPageRoute(
                                    builder: (context) => VerifyOTPPage(
                                        phoneNumber: '',
                                        email: _emailController.text,
                                    ),
                                ),
                            );
                        }
                    },
                    child: Text('Continue'),
                ),
            ),
        ),
    );
}
}

```

```

class VerifyOTPPage extends StatefulWidget {
    final String phoneNumber;
    final String email;

    VerifyOTPPage({required this.phoneNumber, required this.email});

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

```

```

class _VerifyOTPPageState extends State<VerifyOTPPage> {
  TextEditingController _otpController = TextEditingController();
  bool _isResendingOTP = true;
  int _resendTimeout = 50;

  @override
  void initState() {
    super.initState();
    // Start countdown for resend OTP
    startResendTimer();
  }

  void startResendTimer() {
    const oneSec = const Duration(seconds: 1);
    Timer.periodic(oneSec, (Timer timer) {
      setState(() {
        if (_resendTimeout < 1) {
          _isResendingOTP = true;
          timer.cancel();
        } else {
          _resendTimeout--;
        }
      });
    });
  }

  void resendOTP() {
    // Implement OTP resend logic here
    // You can call your API or perform any necessary actions
    setState(() {
      _isResendingOTP = true;
      _resendTimeout = 50; // Reset resend timeout
    });
    startResendTimer(); // Start countdown again
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Verify OTP'),
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.start,
          children: [
            Text(
              'We\'ve sent an OTP to ${widget.phoneNumber.isNotEmpty ? widget.phoneNumber : widget.email}',
              style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
            ),
            SizedBox(height: 16),
            Row(

```



```

        },
        child: Center(
            child: Text(
                'Having trouble? Write to us at help@unacademy.com',
                style: TextStyle(fontWeight: FontWeight.bold, color: Colors.blue),
            ),
        ),
    ),
),
SizedBox(height: 16),
Expanded(
    child: Align(
        alignment: Alignment.bottomCenter,
        child: ElevatedButton(
            onPressed: () {
                Navigator.push( // Navigating to BookVisit page
                    context,
                    MaterialPageRoute(builder: (context) => const HomePage()),
                );
            },
            child: Text('Verify'),
        ),
    ),
),
],
),
),
);
}
}

```

### **The Code is of VerifyOTPPage.dart:**

```

import 'dart:async'; // Add this import statement

import 'package:flutter/material.dart';
import 'package:unacademy_app/home_page.dart';

class VerifyOTPPage extends StatefulWidget {
    final String phoneNumber;
    final String email;

    VerifyOTPPage({required this.phoneNumber, required this.email});

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

class _VerifyOTPPageState extends State<VerifyOTPPage> {
    TextEditingController _otpController = TextEditingController();
    bool _isResendingOTP = false;
    int _resendTimeOut = 50;
    bool _isPhoneVerification = false;

    @override
    void initState() {

```

```

super.initState();
// Start countdown for resend OTP
startResendTimer();
}

void startResendTimer() {
const oneSec = const Duration(seconds: 1);
Timer.periodic(oneSec, (Timer timer) {
setState(() {
if (_resendTimeout < 1) {
_isResendingOTP = false;
timer.cancel();
} else {
_resendTimeout--;
}
});
});
}

void resendOTP() {
// Implement OTP resend logic here
// You can call your API or perform any necessary actions
setState(() {
_isResendingOTP = true;
_resendTimeout = 50; // Reset resend timeout
});
startResendTimer(); // Start countdown again
}

void verifyOTP() {
// Implement OTP verification logic here
// Check if the entered OTP is correct or not
// For demonstration, let's assume the correct OTP is '453347'
String correctOTP = '453347';
String enteredOTP = _otpController.text;
if (enteredOTP == correctOTP) {
// OTP verified successfully
ScaffoldMessenger.of(context).showSnackBar(
SnackBar(
content: Text('OTP verified successfully. You are now a verified user.'),
backgroundColor: Colors.green,
),
);
// Navigate to HomePage
print("Navigating to HomePage");
Navigator.pushAndRemoveUntil(
context,
MaterialPageRoute(builder: (context) => HomePage()),
(route) => false, // Remove all routes until the new route is pushed
);
} else {
// Incorrect OTP
ScaffoldMessenger.of(context).showSnackBar(
SnackBar(
content: Text('Incorrect OTP. Please try again.'),
)
);
}
}

```

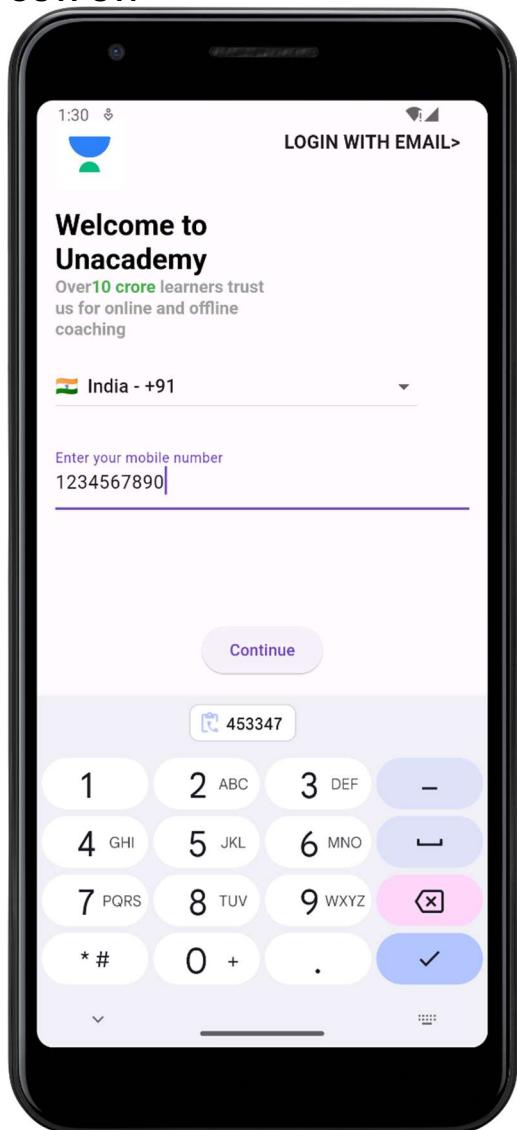
```

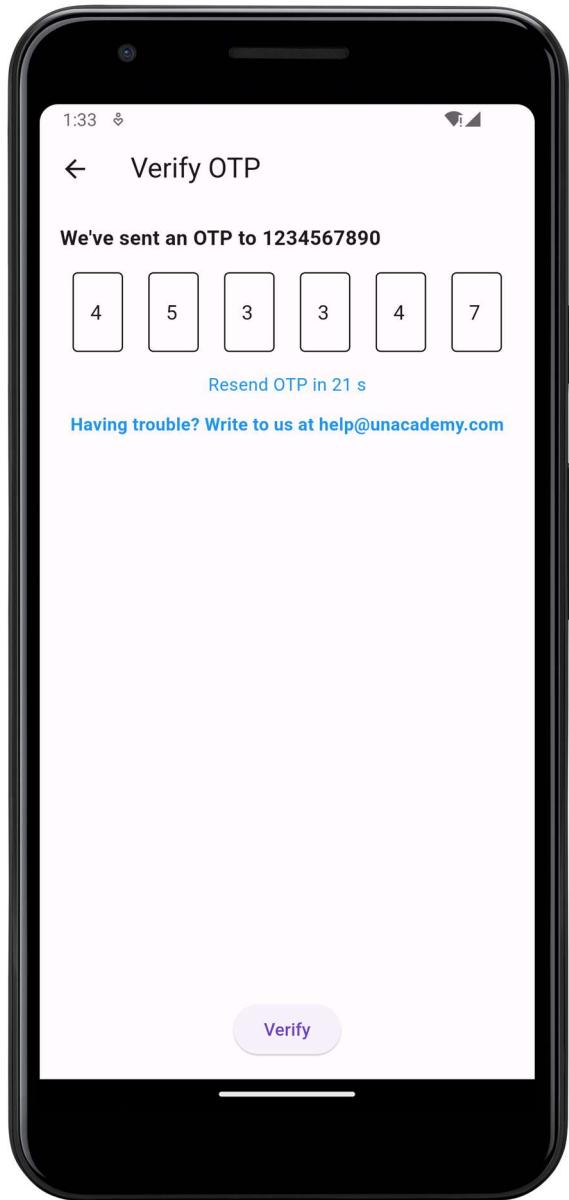
        backgroundColor: Colors.red,
    ),
);
}
}

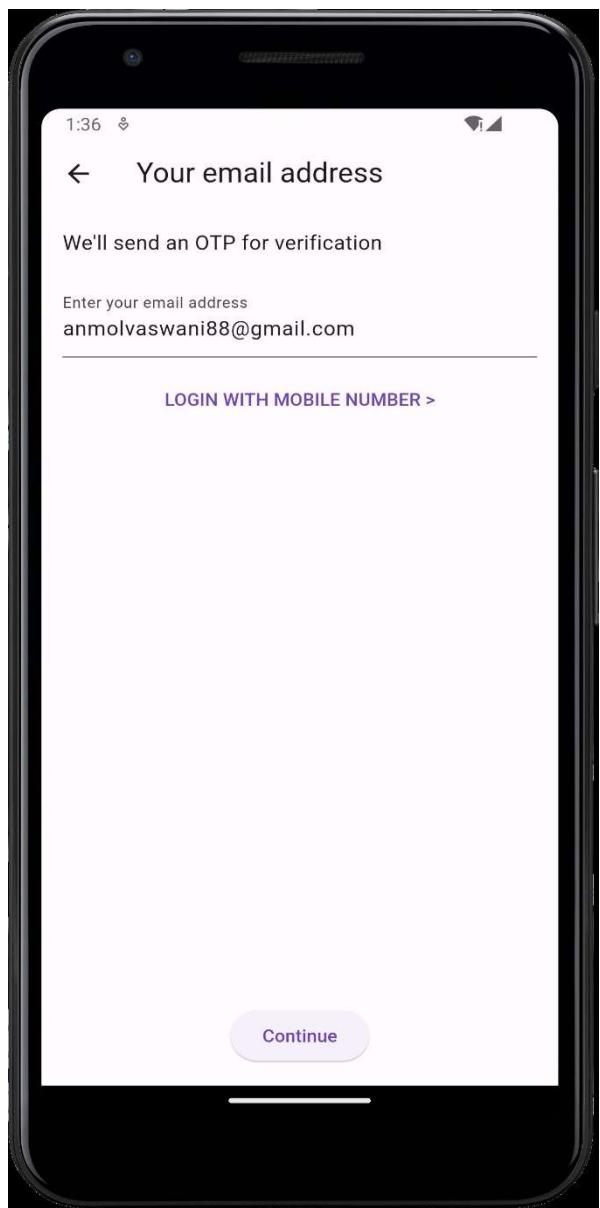
@Override
Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(
            title: Text('Verify OTP'),
        ),
        body: Padding(
            padding: const EdgeInsets.all(16.0),
        child: Column(
            mainAxisAlignment: MainAxisAlignment.start,
            children: [
                Text(
                    _isPhoneVerification
                    ? 'We\'ve sent an OTP to your phone number ${widget.phoneNumber}'
                    : 'We\'ve sent an OTP to your email ${widget.email}',
                    style: TextStyle(fontSize: 16, fontWeight: FontWeight.bold),
                ),
                SizedBox(height: 16),
                Row(
                    mainAxisAlignment: MainAxisAlignment.spaceAround,
                    children: List.generate(
                        6,
                        (index) => SizedBox(
                            width: 40,
                            child: TextFormField(
                                controller: _otpController,
                                maxLength: 1,
                                keyboardType: TextInputType.number,
                                textAlign: TextAlign.center,
                                decoration: InputDecoration(
                                    counterText: "",
                                    enabledBorder: OutlineInputBorder(
                                        borderSide: BorderSide(width: 1),
                                    ),
                                    focusedBorder: OutlineInputBorder(
                                        borderSide: BorderSide(width: 1),
                                    ),
                                ),
                                onChanged: (value) {
                                    // Automatically move to the next box when a digit is entered
                                    if (value.isNotEmpty) {
                                        if (index < 5) {
                                            FocusScope.of(context).nextFocus();
                                        } else {
                                            FocusScope.of(context).unfocus();
                                        }
                                    }
                                },
                            ),
                        ),
                    ),
                ),
            ],
        ),
    );
}

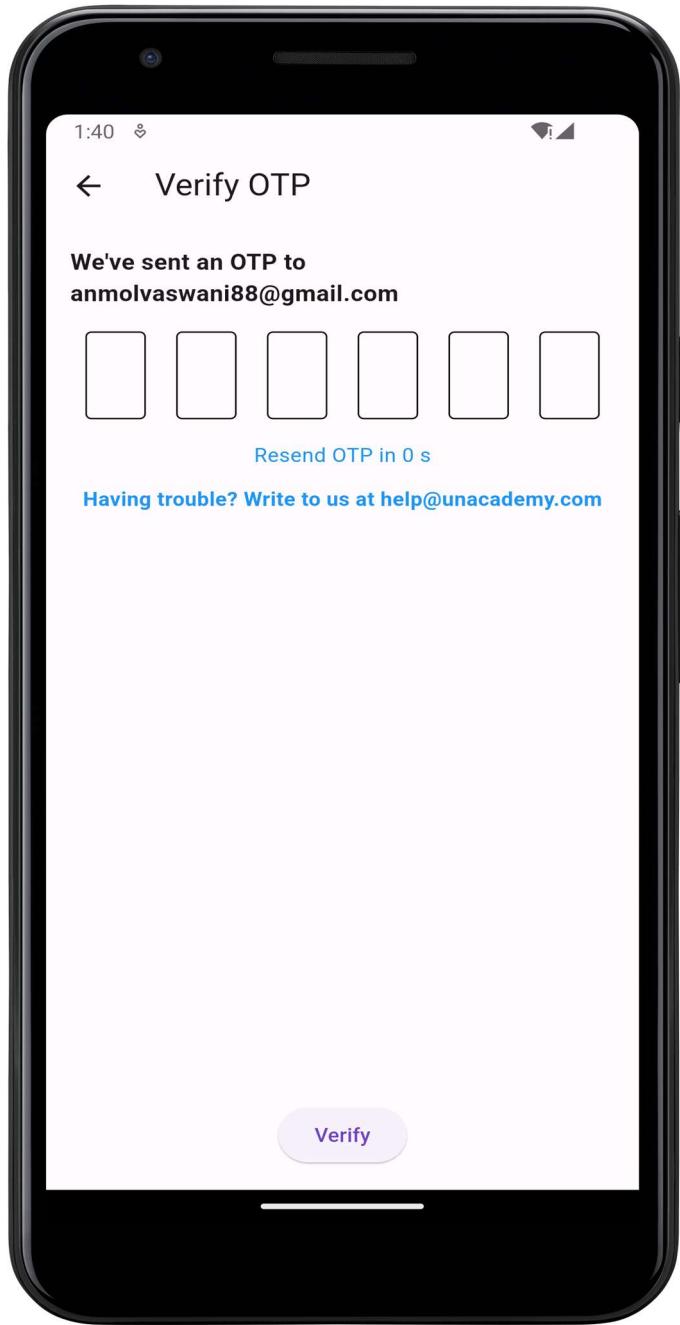
```



**OUTPUT: -**







**Conclusion:** - We have successfully implemented the form in Flutter UI using widgets