



Experiment	4
Aim	Develop an app for TPO Office at SPIT
Objective	<ul style="list-style-type: none">• To Create TPO App using UI Component• To Develop SideDrawer Navigation• To Use 60-30-10 rule
Name	Abhijit Khambekar
UCID	2024510027
Class	FYMCA
Batch	D
Date of Submission	19/02/2025

Technology used	Flutter
Task	<p>Develop an app for TPO with the following requirement (Use side navigation drawer) :</p> <ol style="list-style-type: none">1. Registration Form (Name, E-mail id, Contact, Roll no, HSC (Which College, Year of Passing, total, out of, should calculate percentage), SSC marks (Which College, Year of Passing, total, out of, should calculate percentage), Marks in sem 1-5 (CGPA and Percentage), upload Resume, Any Additional Courses).2. User should be able to view the uploaded Resume. (optional)3. Company details should be visible to the user. (Name of the company, Location, Range of the salary be given for internship/Placement, Short Profile).4. Student should be able to view their own details.
Code with proper label	Main.dart Code:



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

import 'home_page.dart';

import 'registration_form_page.dart';

import 'student_details_page.dart';

import 'company_page.dart';


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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'TPO App',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: HomePage(),
      routes: {
        '/home': (context) => HomePage(),
        '/registration': (context) => RegistrationFormPage(),
        '/studentDetails': (context) => StudentDetailsPage(),
        '/company': (context) => CompanyPage(),
      },
    );
  }
}
```



Home_page.dart Code:

```
import 'package:flutter/material.dart';
import 'package:google_fonts/google_fonts.dart'; // Import Google Fonts

class HomePage extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Welcome to TPO'),
      ),
      drawer: AppDrawer(),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            // TPO!!! in a big, stylish, and funky font
            Text(
              'TPO!!!',
              style: GoogleFonts.pacifico(
                fontSize: 72,
                fontWeight: FontWeight.bold,
                color: Colors.blue,
              ),
            ),
            SizedBox(height: 20),
          ],
        ),
      ),
    );
  }
}
```



```
// Tagline

Text(
  'Where students are laboured LEGALLY!!!',
  style: GoogleFonts.lato(
    fontSize: 20,
    fontStyle: FontStyle.italic,
    color: Colors.grey[700],
  ),
),
],
),
),
);
}
}

class AppDrawer extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Drawer(
      child: ListView(
        padding: EdgeInsets.zero,
        children: <Widget>[
          DrawerHeader(
            decoration: BoxDecoration(
              color: Colors.blue,
            ),
          ),
        ],
      ),
    );
  }
}
```



	<pre>child: Text('Welcome to TPO', style: TextStyle(color: Colors.white, fontSize: 24,),),),), ListTile(leading: Icon(Icons.home), title: Text('Home'), onTap: () { Navigator.pushReplacementNamed(context, '/home'); },), ListTile(leading: Icon(Icons.app_registration), title: Text('Registration Form'), onTap: () { Navigator.pushReplacementNamed(context, '/registration'); },), ListTile(leading: Icon(Icons.person), title: Text('Student Details'), onTap: () { Navigator.pushReplacementNamed(context, '/studentDetails');</pre>
--	--



```
    },  
  ),  
  ListTile(  
    leading: Icon(Icons.business),  
    title: Text('Company'),  
    onTap: () {  
      Navigator.pushReplacementNamed(context, '/company');  
    },  
  ),  
],  
),  
);  
}  
}
```

Registration_form_page.dart Code:

```
import 'package:flutter/material.dart';  
import 'global_data.dart'; // Import the global list  
import 'home_page.dart';  
  
class RegistrationFormPage extends StatefulWidget {  
  @override  
  _RegistrationFormPageState createState() =>  
    _RegistrationFormPageState();  
}  
  
class _RegistrationFormPageState extends State<RegistrationFormPage> {
```



```
final _formKey = GlobalKey<FormState>();  
final _nameController = TextEditingController();  
final _emailController = TextEditingController();  
final _contactController = TextEditingController();  
final _rollNoController = TextEditingController();  
final _hscCollegeController = TextEditingController();  
final _hscMarksController = TextEditingController();  
final _hscOutOfController = TextEditingController();  
final _bachelorCourseController = TextEditingController();  
final _bachelorCollegeController = TextEditingController();  
double _percentage = 0.0;  
  
void _calculatePercentage() {  
  if (_hscMarksController.text.isNotEmpty &&  
      _hscOutOfController.text.isNotEmpty) {  
    double marks = double.parse(_hscMarksController.text);  
    double outOf = double.parse(_hscOutOfController.text);  
    setState(() {  
      _percentage = (marks / outOf) * 100;  
    });  
  } else {  
    setState(() {  
      _percentage = 0.0;  
    });  
  }  
}
```



```
// Validate Contact Number

String? _validateContact(String? value) {
    if (value == null || value.isEmpty) {
        return 'Contact number is required';
    }
    if (value.length != 10 || !RegExp(r'^[0-9]+$').hasMatch(value)) {
        return 'Contact number must be 10 digits';
    }
    return null;
}

// Validate Roll No (UCID)

String? _validateRollNo(String? value) {
    if (value == null || value.isEmpty) {
        return 'Roll No is required';
    }
    if (!RegExp(r'^[0-9]+$').hasMatch(value)) {
        return 'Roll No must contain only numbers';
    }
    return null;
}

// Validate Marks Obtained and Out Of

String? _validateMarks(String? value) {
    if (value == null || value.isEmpty) {
        return 'This field is required';
    }
}
```




```
if (!RegExp(r'^[0-9]+$').hasMatch(value)) {  
    return 'This field must contain only numbers';  
}  
return null;  
}  
  
// Validate Email ID  
String? _validateEmail(String? value) {  
    if (value == null || value.isEmpty) {  
        return 'Email ID is required';  
    }  
    if (!RegExp(r'^[\w-\.\.]+\@([\w-]+\.\.)+[\w-]{2,4}$').hasMatch(value)) {  
        return 'Enter a valid email address';  
    }  
    return null;  
}  
  
void _register() {  
    if (_formKey.currentState!.validate()) {  
        // Save data to a map  
        Map<String, dynamic> studentData = {  
            'Name': _nameController.text,  
            'Email ID': _emailController.text,  
            'Contact': _contactController.text,  
            'Roll No': _rollNoController.text,  
            'HSC College': _hscCollegeController.text,  
            'HSC Marks Obtained': _hscMarksController.text,
```



```
'HSC Out Of': _hscOutOfController.text,

'Percentage': _percentage.toStringAsFixed(2),

'Bachelor Course': _bachelorCourseController.text,

'Bachelor College': _bachelorCollegeController.text,

};

// Add student data to the global list
students.add(studentData);

// Clear form fields
_formKey.currentState!.reset();

setState() {
  _percentage = 0.0;
});

// Show success message
ScaffoldMessenger.of(context).showSnackBar(
  SnackBar(content: Text('Registration Successful!')),
);

// Navigate to Student Details Page
Navigator.pushNamed(
  context,
  '/studentDetails',
);
}
}
```



```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('Welcome to TPO'),
    ),
    drawer: AppDrawer(),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Form(
        key: _formKey,
        child: ListView(
          children: [
            _buildTextField('Name', _nameController, TextInputType.name),
            _buildTextField('Email ID', _emailController,
              TextInputType.emailAddress, validator: _validateEmail),
            _buildTextField('Contact', _contactController,
              TextInputType.phone, validator: _validateContact),
            _buildTextField('Roll No', _rollNoController,
              TextInputType.number, validator: _validateRollNo),
            _buildTextField('HSC College', _hscCollegeController,
              TextInputType.text),
            _buildTextField('HSC Marks Obtained', _hscMarksController,
              TextInputType.number, validator: _validateMarks, onChanged: (_) =>
              _calculatePercentage()),
            _buildTextField('HSC Out Of', _hscOutOfController,
              TextInputType.number, validator: _validateMarks, onChanged: (_) =>
              _calculatePercentage()),
```



```
        _buildReadOnlyField('Percentage',
        _percentage.toStringAsFixed(2)),
        _buildTextField('Bachelor Course', _bachelorCourseController,
        TextInputType.text),
        _buildTextField('Bachelor College', _bachelorCollegeController,
        TextInputType.text),
        SizedBox(height: 20),
        ElevatedButton(
          onPressed: _register,
          child: Text('Register'),
        ),
      ],
    ),
  ),
),
);
}
```

```
Widget _buildTextField(String label, TextEditingController controller,
  TextInputType keyboardType, {Function(String)? onChanged, String?
  Function(String)? validator}) {
```

```
  return Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      Text(label),
      TextFormField(
        controller: controller,
        keyboardType: keyboardType,
        decoration: InputDecoration(
```



```
        hintText: 'Enter $label',
      ),
      validator: validator,
      onChanged: onChanged,
    ),
    SizedBox(height: 10),
  ],
);
}

Widget _buildReadOnlyField(String label, String value) {
  return Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      Text(label),
      TextFormField(
        readOnly: true,
        decoration: InputDecoration(
          hintText: value,
        ),
      ),
      SizedBox(height: 10),
    ],
  );
}
```



Global_data.dart Code:

```
List<Map<String, dynamic>> students = [];
```

```
List<Map<String, dynamic>> companies = [];
```

Student_details_page.dart Code:

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

```
import 'global_data.dart'; // Import the global list
```

```
import 'home_page.dart';
```

```
class StudentDetailsPage extends StatefulWidget {
```

```
  @override
```

```
  _StudentDetailsPageState createState() => _StudentDetailsPageState();
```

```
}
```

```
class _StudentDetailsPageState extends State<StudentDetailsPage> {
```

```
  void _deleteStudent(int index) {
```

```
    setState() {
```

```
      students.removeAt(index);
```

```
    });
```

```
}
```

```
  @override
```

```
  Widget build(BuildContext context) {
```

```
    return Scaffold(
```

```
      appBar: AppBar(
```

```
        title: Text('Welcome to TPO'),
```

```
      ),
```



```
drawer: AppDrawer(),  
body: students.isEmpty  
  ? Center(child: Text('No student details available'))  
  : ListView.builder(  
    padding: EdgeInsets.all(16),  
    itemCount: students.length,  
    itemBuilder: (context, index) {  
      final student = students[index];  
      return Card(  
        margin: EdgeInsets.only(bottom: 16),  
        child: Padding(  
          padding: const EdgeInsets.all(16.0),  
          child: Column(  
            crossAxisAlignment: CrossAxisAlignment.start,  
            children: [  
              Text('Name: ${student['Name']}', style: TextStyle(fontSize:  
16)),  
              SizedBox(height: 8),  
              Text('Email ID: ${student['Email ID']}', style:  
TextStyle(fontSize: 16)),  
              SizedBox(height: 8),  
              Text('Roll No: ${student['Roll No']}', style:  
TextStyle(fontSize: 16)),  
              SizedBox(height: 8),  
              Text('Contact: ${student['Contact']}', style:  
TextStyle(fontSize: 16)),  
              SizedBox(height: 8),  
              Text('HSC College: ${student['HSC College']}', style:  
TextStyle(fontSize: 16)),
```



```
SizedBox(height: 8),

Text('HSC Marks: ${student['HSC Marks
Obtained']}/${student['HSC Out Of']}', style: TextStyle(fontSize: 16)),

SizedBox(height: 8),

Text('Percentage: ${student['Percentage']}%', style:
TextStyle(fontSize: 16)),

SizedBox(height: 8),

Text('Bachelor Course: ${student['Bachelor Course']}', style:
TextStyle(fontSize: 16)),

SizedBox(height: 8),

Text('Bachelor College: ${student['Bachelor College']}',
style: TextStyle(fontSize: 16)),

SizedBox(height: 16),

Align(
  alignment: Alignment.centerRight,
  child: ElevatedButton(
    onPressed: () => _deleteStudent(index),
    child: Text('Delete'),
    style: ElevatedButton.styleFrom(
      backgroundColor: Colors.red,
      foregroundColor: Colors.white,
    ),
  ),
),
),
);
```




```
    },  
    ),  
);  
}  
}  
  
Company_page.dart Code:  
import 'package:flutter/material.dart';  
import 'global_data.dart'; // Import the global list  
import 'home_page.dart';  
  
class CompanyPage extends StatefulWidget {  
  @override  
  _CompanyPageState createState() => _CompanyPageState();  
}  
  
class _CompanyPageState extends State<CompanyPage> {  
  final _companyNameController = TextEditingController();  
  final _companyDetailsController = TextEditingController();  
  final _jobRoleController = TextEditingController();  
  final _jobRequirementController = TextEditingController();  
  
  void _addCompany() {  
    showDialog(  
      context: context,  
      builder: (context) {  
        return AlertDialog(  

```



	<pre>title: Text('Add Company'), content: SingleChildScrollView(child: Column(mainAxisAlignment: MainAxisAlignment.min, children: [_buildTextField('Company Name', _companyNameController), _buildTextField('Company Details', _companyDetailsController), _buildTextField('Job Role', _jobRoleController), _buildTextField('Job Requirement', _jobRequirementController),],),), actions: [TextButton(onPressed: () { Navigator.pop(context); // Close the dialog }, child: Text('Cancel'),), TextButton(onPressed: () { if (_companyNameController.text.isNotEmpty && _companyDetailsController.text.isNotEmpty && _jobRoleController.text.isNotEmpty && _jobRequirementController.text.isNotEmpty) { // Save company data to a map Map<String, dynamic> companyData = {</pre>
--	---



```
'Company Name': _companyNameController.text,  
'Company Details': _companyDetailsController.text,  
'Job Role': _jobRoleController.text,  
'Job Requirement': _jobRequirementController.text,  
};  
  
// Add company data to the global list  
setState() {  
  companies.add(companyData);  
});  
  
// Clear text fields  
_companyNameController.clear();  
_companyDetailsController.clear();  
_jobRoleController.clear();  
_jobRequirementController.clear();  
  
// Close the dialog  
Navigator.pop(context);  
} else {  
  ScaffoldMessenger.of(context).showSnackBar(  
    SnackBar(content: Text('Please fill all fields')),  
  );  
}  
},  
child: Text('Create'),  
,
```



```
    ],  
    );  
  },  
  );  
}  
  
void _deleteCompany(int index) {  
  setState(() {  
    companies.removeAt(index);  
  });  
}  
  
@override  
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(  
      title: Text('Welcome to TPO'),  
      actions: [  
        IconButton(  
          onPressed: _addCompany,  
          icon: Icon(Icons.add),  
          tooltip: 'Add Company', // Tooltip for the button  
        ),  
      ],  
    ),  
    drawer: AppDrawer(),  
    body: companies.isEmpty
```



```
? Center(child: Text('No company details available'))

: ListView.builder(
  padding: EdgeInsets.all(16),
  itemCount: companies.length,
  itemBuilder: (context, index) {
    final company = companies[index];
    return Card(
      margin: EdgeInsets.only(bottom: 16),
      child: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Text('Company Name: ${company['Company Name']}',
style: TextStyle(fontSize: 16)),
            SizedBox(height: 8),
            Text('Company Details: ${company['Company Details']}',
style: TextStyle(fontSize: 16)),
            SizedBox(height: 8),
            Text('Job Role: ${company['Job Role']}', style:
TextStyle(fontSize: 16)),
            SizedBox(height: 8),
            Text('Job Requirement: ${company['Job Requirement']}',
style: TextStyle(fontSize: 16)),
            SizedBox(height: 16),
            Align(
              alignment: Alignment.centerRight,
              child: ElevatedButton(
```



```
onPressed: () => _deleteCompany(index),
child: Text('Delete'),
style: ElevatedButton.styleFrom(
  backgroundColor: Colors.red,
  foregroundColor: Colors.white,
),
),
),
],
),
),
);
},
),
);
}

Widget _buildTextField(String label, TextEditingController controller) {
  return Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      Text(label),
      TextFormField(
        controller: controller,
        decoration: InputDecoration(
          hintText: 'Enter $label',
        ),
      ),
    ],
  );
}
```



```
),  
    SizedBox(height: 10),  
  ],  
);  
}  
}  
  
import 'package:flutter/material.dart';  
import 'global_data.dart'; // Import the global list  
import 'home_page.dart';  
  
class CompanyPage extends StatefulWidget {  
  @override  
  _CompanyPageState createState() => _CompanyPageState();  
}  
  
class _CompanyPageState extends State<CompanyPage> {  
  final _companyNameController = TextEditingController();  
  final _companyDetailsController = TextEditingController();  
  final _jobRoleController = TextEditingController();  
  final _jobRequirementController = TextEditingController();  
  
  void _addCompany() {  
    showDialog(  
      context: context,  
      builder: (context) {  
        return AlertDialog(  
          title: Text('Add Company'),  
          content: Column(  
            children: [  
              TextFormField(controller: _companyNameController),  
              TextFormField(controller: _companyDetailsController),  
              TextFormField(controller: _jobRoleController),  
              TextFormField(controller: _jobRequirementController),  
            ],  
          ),  
        );  
      },  
    );  
  }  
}
```



	<pre>content: SingleChildScrollView(child: Column(mainAxisAlignment: MainAxisAlignment.min, children: [_buildTextField('Company Name', _companyNameController), _buildTextField('Company Details', _companyDetailsController), _buildTextField('Job Role', _jobRoleController), _buildTextField('Job Requirement', _jobRequirementController),],),), actions: [TextButton(onPressed: () { Navigator.pop(context); // Close the dialog }, child: Text('Cancel'),), TextButton(onPressed: () { if (_companyNameController.text.isNotEmpty && _companyDetailsController.text.isNotEmpty && _jobRoleController.text.isNotEmpty && _jobRequirementController.text.isNotEmpty) { // Save company data to a map Map<String, dynamic> companyData = { 'Company Name': _companyNameController.text,</pre>
--	--



	<pre>'Company Details': _companyDetailsController.text, 'Job Role': _jobRoleController.text, 'Job Requirement': _jobRequirementController.text, }; // Add company data to the global list setState() { companies.add(companyData); }); // Clear text fields _companyNameController.clear(); _companyDetailsController.clear(); _jobRoleController.clear(); _jobRequirementController.clear(); // Close the dialog Navigator.pop(context); } else { ScaffoldMessenger.of(context).showSnackBar(SnackBar(content: Text('Please fill all fields')),); }, child: Text('Create'),),],</pre>
--	---



```
);  
},  
);  
}  
  
void _deleteCompany(int index) {  
    setState(() {  
        companies.removeAt(index);  
    });  
}  
  
@override  
Widget build(BuildContext context) {  
    return Scaffold(  
        appBar: AppBar(  
            title: Text('Welcome to TPO'),  
            actions: [  
                IconButton(  
                    onPressed: _addCompany,  
                    icon: Icon(Icons.add),  
                    tooltip: 'Add Company', // Tooltip for the button  
                ),  
            ],  
        ),  
        drawer: AppDrawer(),  
        body: companies.isEmpty  
            ? Center(child: Text('No company details available'))
```



```
: ListView.builder(  
  padding: EdgeInsets.all(16),  
  itemCount: companies.length,  
  itemBuilder: (context, index) {  
    final company = companies[index];  
    return Card(  
      margin: EdgeInsets.only(bottom: 16),  
      child: Padding(  
        padding: const EdgeInsets.all(16.0),  
        child: Column(  
          crossAxisAlignment: CrossAxisAlignment.start,  
          children: [  
            Text('Company Name: ${company['Company Name']}',  
style: TextStyle(fontSize: 16)),  
            SizedBox(height: 8),  
            Text('Company Details: ${company['Company Details']}',  
style: TextStyle(fontSize: 16)),  
            SizedBox(height: 8),  
            Text('Job Role: ${company['Job Role']}', style:  
TextStyle(fontSize: 16)),  
            SizedBox(height: 8),  
            Text('Job Requirement: ${company['Job Requirement']}',  
style: TextStyle(fontSize: 16)),  
            SizedBox(height: 16),  
            Align(  
              alignment: Alignment.centerRight,  
              child: ElevatedButton(  
                onPressed: () => _deleteCompany(index),
```



```
child: Text('Delete'),  
style: ElevatedButton.styleFrom(  
  backgroundColor: Colors.red,  
  foregroundColor: Colors.white,  
),  
,  
,  
],  
,  
,  
);  
},  
,  
);  
}  
  
Widget _buildTextField(String label, TextEditingController controller) {  
  return Column(  
    crossAxisAlignment: CrossAxisAlignment.start,  
    children: [  
      Text(label),  
      TextFormField(  
        controller: controller,  
        decoration: InputDecoration(  
          hintText: 'Enter $label',  
        ),  
      ),  
    ],  
  );  
}
```

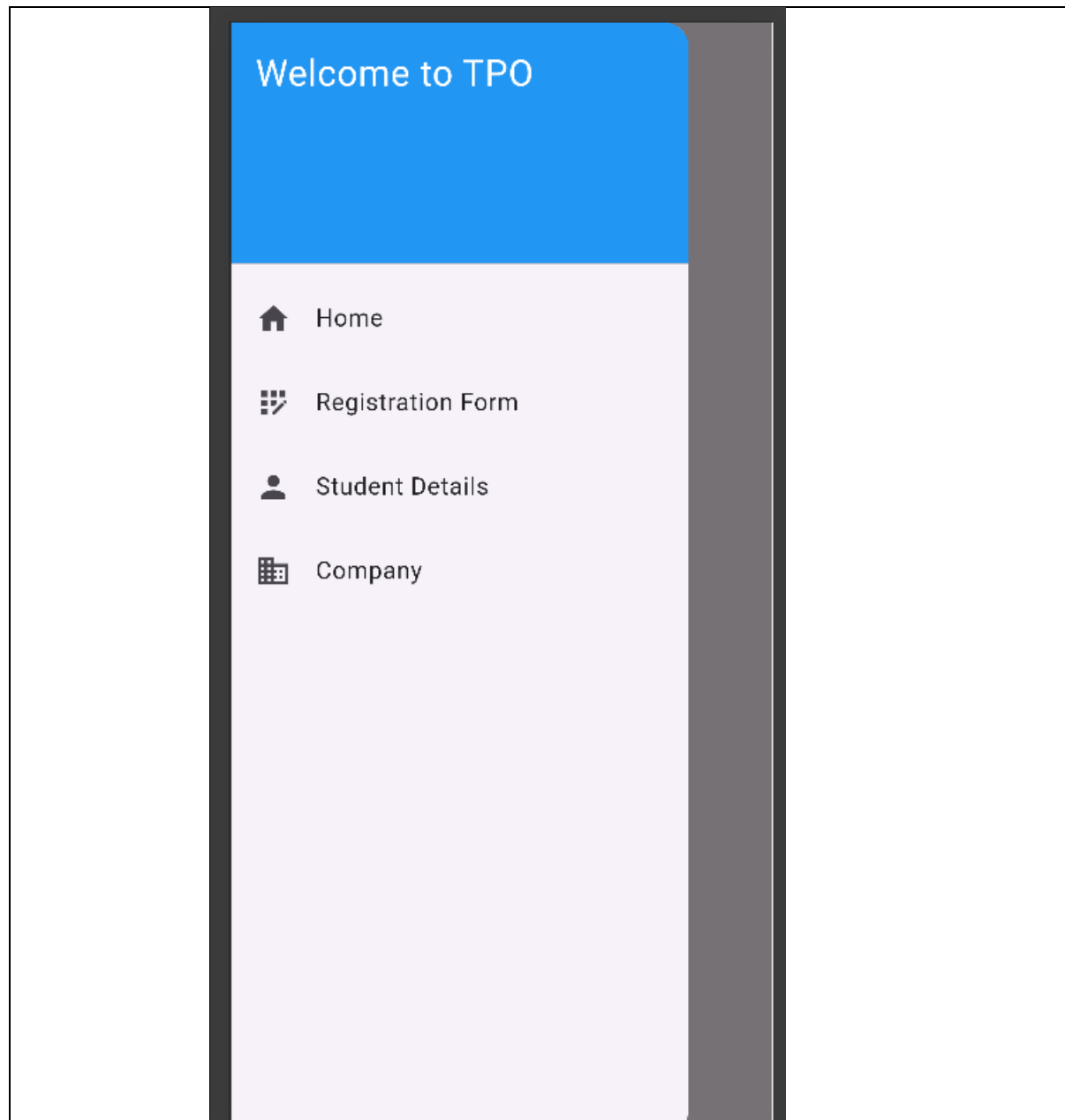


	<pre>SizedBox(height: 10),],); } }</pre>
--	--



Screenshots







☰ Welcome to TPO

Name

Enter Name

Email ID

Enter Email ID

Contact

Enter Contact

Roll No

Enter Roll No

HSC College

Enter HSC College

HSC Marks Obtained

Enter HSC Marks Obtained

HSC Out Of

Enter HSC Out Of

Percentage

0.00

Bachelor Course



Welcome to TPO

Enter Contact

Roll No

Enter Roll No

HSC College

Enter HSC College

HSC Marks Obtained

Enter HSC Marks Obtained

HSC Out Of

Enter HSC Out Of

Percentage

0.00

Bachelor Course

Enter Bachelor Course

Bachelor College

Enter Bachelor College

Register



☰

Welcome to TPO

Enter Contact

Contact number is required

Roll No

Enter Roll No

Roll No is required

HSC College

Enter HSC College

HSC Marks Obtained

Enter HSC Marks Obtained

This field is required

HSC Out Of

Enter HSC Out Of

This field is required

Percentage

0.00

Bachelor Course

Enter Bachelor Course

Bachelor College

Enter Bachelor College



☰

Welcome to TPO

Contact number is required

Roll No

2024510027

Roll No is required

HSC College

RKT

HSC Marks Obtained

500

This field is required

HSC Out Of

600

This field is required

Percentage

83.33

Bachelor Course

Bsc Computer science

Bachelor College

CHM College



☰ Welcome to TPO

Name

Abhijit Khambekar

Email ID

test@ok.com

Email ID is required

Contact

7845124547

Contact number is required

Roll No

2024510027

Roll No is required

HSC College

RKT

HSC Marks Obtained

500

This field is required

HSC Out Of

600

This field is required

-



Welcome to TPO

Name: Abhijit Khambekar

Email ID: test@ok.com

Roll No: 2024510027

Contact: 7845124547

HSC College: RKT

HSC Marks: 500/600

Percentage: 83.33%

Bachelor Course: Bsc Computer science

Bachelor College: CHM College

Delete







☰

Welcome to TPO

+

Add Company

Company Name

TCS

Company Details

Awesome

Job Role

QA

Job Requirement

None

Cancel

Create



Welcome to TPO

+

Company Name: TCS

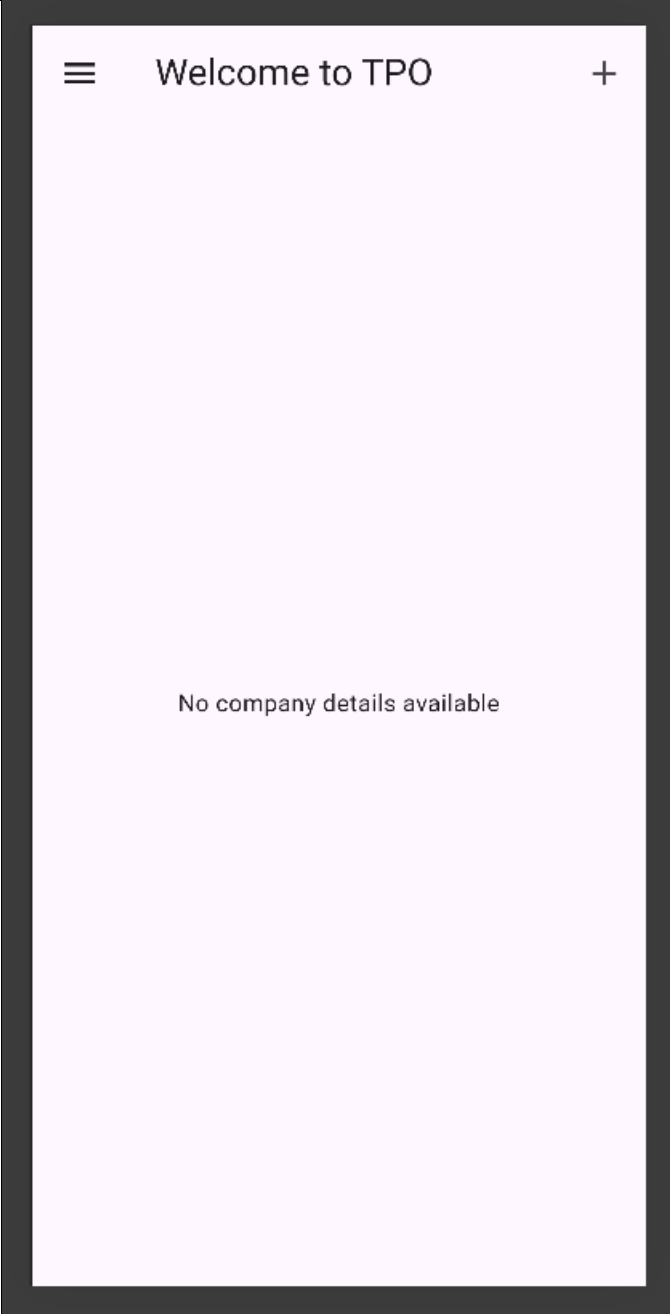
Company Details: Awesome

Job Role: QA

Job Requirement: None

Delete



	
Question and Answers	<p>Answer the following Questions:</p> <ol style="list-style-type: none">1. How to create SideDrawer Navigation? <p>Solu: To create a SideDrawer Navigation in an app, you can use the Drawer widget in Flutter. It is usually placed inside the Scaffold widget using the drawer property. Inside the drawer, you can add a ListView with ListTile widgets for each navigation option. When a user taps an option, it takes</p>



Bharatiya Vidya Bhavan's
SARDAR PATEL INSTITUTE OF TECHNOLOGY

(Autonomous Institute Affiliated to University of Mumbai) Munshi Nagar, Andheri (W), Mumbai
– 400 058. Department of Computer Science and Engineering

	<p>them to the corresponding page using <code>Navigator.push()</code>. You can also add a <code>DrawerHeader</code> to display a title or user info at the top.</p> <p>2. How did you use 60-30-10 rule in your application?</p> <p>Solu: The 60-30-10 rule helps maintain a balanced color scheme in an app. In my application, I used 60% for the primary background color, 30% for secondary elements like buttons and cards, and 10% for accent colors like icons and highlights. This makes the UI look clean and organized. The primary color creates the base, the secondary color supports it, and the accent color adds contrast. This approach improves both appearance and user experience.</p> <p>3. Which new elements did you use for creating UI components?</p> <p>Solu: I used several new UI components while creating the app. These include the Card widget for organizing content, the Snackbar for showing quick messages, and the AlertDialog for important notifications. I also used the ListTile widget for easy navigation options. The TextFormField with input validation helped make the forms more user-friendly and secure.</p>
Conclusion	Learned how to create a Side Navigation Drawer, Use of Google fonts, and how to store data temporary.