NAME : SOHAM SHETYE ROLL NO. : 54
DIV : D15A BATCH : C

MAD PWA LAB

LAB 5

Aim: To apply navigation, routing and gestures in Flutter app.

THEORY :

Flutter provides robust tools for crafting intuitive and user-friendly app navigation experiences. Here's a theoretical breakdown of the key concepts:

Navigation:

- Refers to the app's structure and how users move between different screens.
- Involves concepts like hierarchical navigation (think nested screens) and bottom navigation bars.

Routing:

- Defines the mechanism for transitioning between screens, handling the "how" of navigation.
- Utilizes widgets like Navigator and methods like push and pop to manage the navigation stack.
- Can use named routes for clarity and easier maintenance.

Gestures:

- Provide ways for users to interact with the app interface and trigger navigation actions.
- Common gestures include taps, swipes, drags, and long presses.
- Flutter's gesture recognition system allows you to define custom gestures for specific behaviors.

Understanding the interplay:

- Gestures trigger routing actions managed by the Navigator.
- Navigation decisions define the app's flow and screen transitions.
- Consider:

- Use relevant gestures based on platform conventions and user expectations.
- Design smooth and intuitive transitions between screens.
- Implement accessibility features for diverse user needs.

Further Exploration:

- Dive deeper into the official Flutter documentation for navigation and routing:https://docs.flutter.dev/ui/navigation
- Explore gesture recognition in Flutter: <invalid URL removed>
- Experiment with different navigation patterns and gesture interactions to find the perfect fit for your app.

By actively exploring and practicing these concepts, we'll create a truly engaging and user-friendly navigation experience for your Flutter app.

CODE :

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

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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'OLA Taxi Booking App',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: LoginPage(),
      routes: {
        '/home': (context) => HomePage(),
        '/destination': (context) => DestinationPage(),
        '/account': (context) => AccountPage(),
        '/register': (context) => RegisterPage(),
        '/phone_verification': (context) => PhoneVerificationPage(),
      },
    );
```

```
  }
}

class LoginPage extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('OLA Account Login'),
      ),
      body: GestureDetector(
        onTap: () {
          // Dismiss the keyboard when tapping outside the text field
          FocusScope.of(context).unfocus();
        },
        child: Container(
          decoration: BoxDecoration(
            image: DecorationImage(
              image: AssetImage('assets/images/projects.png'),
              fit: BoxFit.cover,
            ),
          ),
          child: Center(
            child: Column(
              mainAxisAlignment: MainAxisAlignment.center,
              children: [
                Expanded(
                  child: SizedBox(),
                ),
                Text(
                  'Login Page',
                  style: TextStyle(
                    color: Colors.white,
                    fontSize: 24,
                    fontWeight: FontWeight.bold,
                  ),
                ),
                SizedBox(height: 20),
                ElevatedButton(
                  onPressed: () {
                    Navigator.pushNamed(context, '/home');
```

```
              },
              style: ElevatedButton.styleFrom(
                primary: Colors.green,
                onPrimary: Colors.white,
              ),
              child: Padding(
                padding: EdgeInsets.symmetric(horizontal: 20),
                child: Text(
                  'Login',
                  style: TextStyle(fontSize: 18),
                ),
              ),
            ),
            SizedBox(height: 10),
            TextButton(
              onPressed: () {
                Navigator.pushNamed(context, '/register');
              },
              style: TextButton.styleFrom(
                primary: Color.fromARGB(255, 5, 185, 29),
              ),
              child: Text(
                'Register',
                style: TextStyle(
                  fontSize: 16,
                  decoration: TextDecoration.underline,
                ),
              ),
            ),
          ],
        ),
      ),
    ),
  );
 }
}

class RegisterPage extends StatefulWidget {
 @override
 _RegisterPageState createState() => _RegisterPageState();
```

```
}

class _RegisterPageState extends State<RegisterPage> {
  TextEditingController nameController = TextEditingController();
  TextEditingController phoneNumberController = TextEditingController();
  TextEditingController emailController = TextEditingController();
  TextEditingController genderController = TextEditingController();
  TextEditingController addressController = TextEditingController();
  TextEditingController ageController = TextEditingController();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Register'),
        backgroundColor: const Color.fromARGB(255, 150, 255, 154),
      ),
      body: GestureDetector(
        onTap: () {
          // Dismiss the keyboard when tapping outside the text field
          FocusScope.of(context).unfocus();
        },
        child: SingleChildScrollView(
          child: Padding(
            padding: const EdgeInsets.all(16.0),
            child: Column(
              mainAxisAlignment: MainAxisAlignment.center,
              children: [
                Text(
                  'Registration Page',
                  style: TextStyle(
                    fontSize: 20.0,
                    fontWeight: FontWeight.bold,
                    color: const Color.fromARGB(255, 150, 255, 154),
                  ),
                ),
                SizedBox(height: 20.0),
                buildTextField(nameController, 'Name'),
                buildTextField(phoneNumberController, 'Phone Number'),
                buildTextField(emailController, 'Email'),
                buildTextField(genderController, 'Gender'),
```

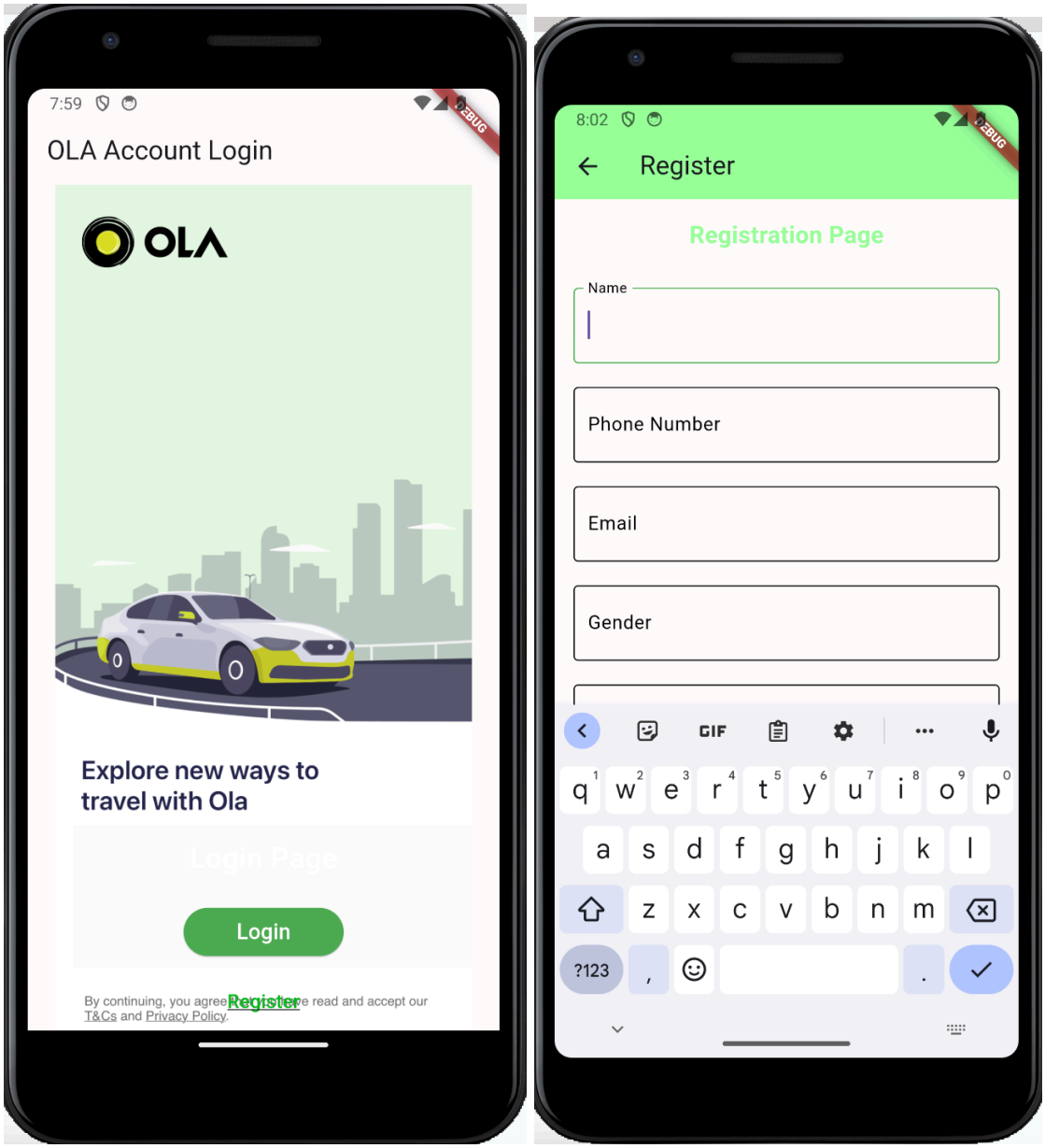
```
            buildTextField(addressController, 'Address'),
            buildTextField(ageController, 'Age'),
            SizedBox(height: 20.0),
            ElevatedButton(
              onPressed: () {
                Navigator.pushNamed(context, '/phone_verification');
              },
              style: ElevatedButton.styleFrom(
                primary: Color.fromARGB(255, 155, 255, 255, 159),
              ),
              child: Text('Continue to Phone Verification'),
            ),
            SizedBox(height: 10.0),
            TextButton(
              onPressed: () {
                Navigator.pop(context); // Navigate back to the previous screen
              },
              style: TextButton.styleFrom(
                primary: Colors.white,
              ),
              child: Text(
                'Back',
                style: TextStyle(
                  fontSize: 16,
                  decoration: TextDecoration.underline,
                ),
              ),
            ),
          ],
        ),
      ),
    ),
  );
}

Widget buildTextField(TextEditingController controller, String labelText) {
  return Padding(
    padding: const EdgeInsets.symmetric(vertical: 10.0),
    child: TextField(
      controller: controller,
```

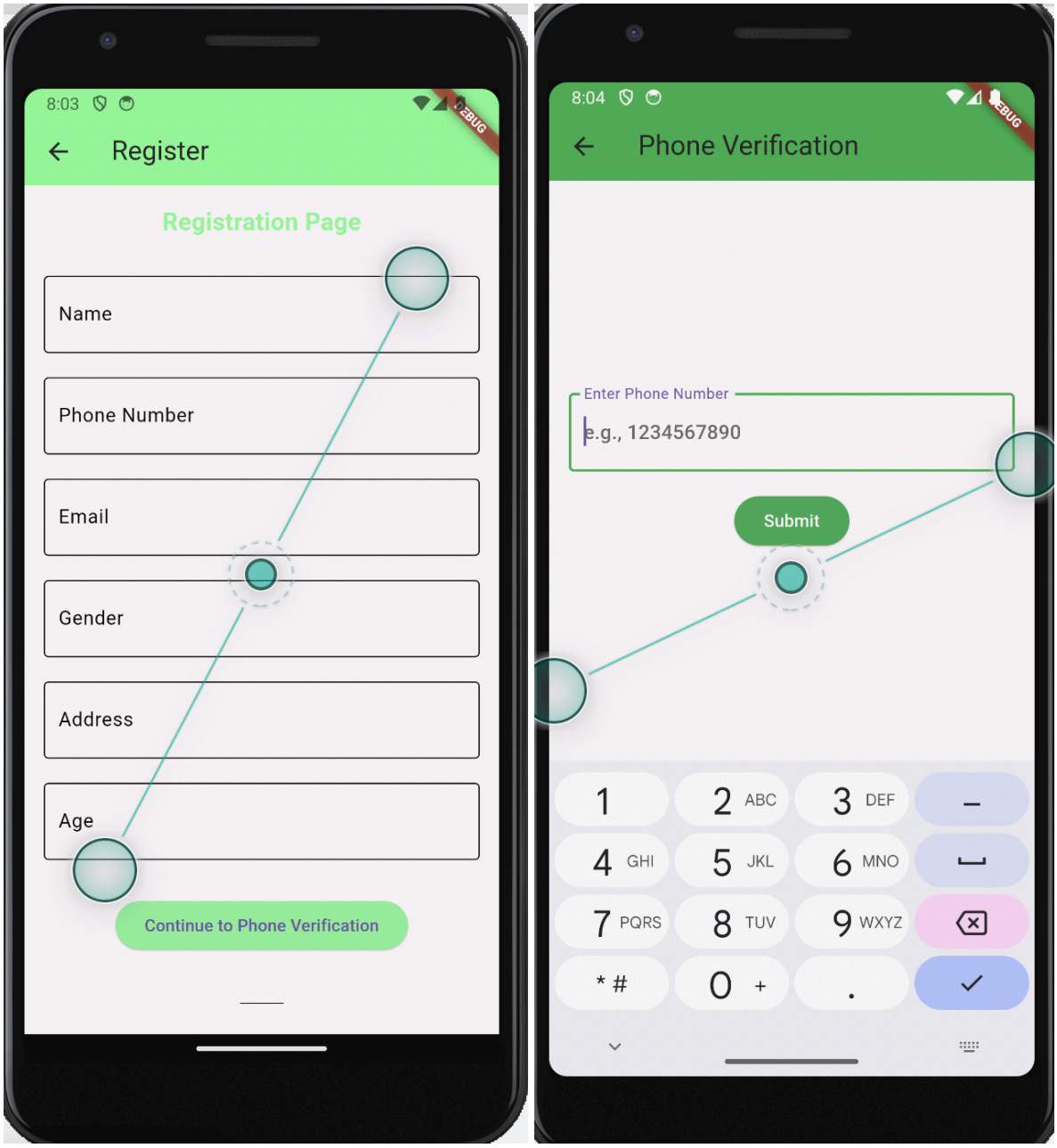
```
      decoration: InputDecoration(
       labelText: labelText,
       labelStyle: TextStyle(color: Color.fromARGB(255, 0, 0, 0)),
       focusedBorder: OutlineInputBorder(
        borderSide: BorderSide(color: Colors.green),
       ),
       enabledBorder: OutlineInputBorder(
        borderSide: BorderSide(color: Colors.black),
       ),
      ),
     ),
    );
  }
}

class PhoneVerificationPage extends StatelessWidget {
  final TextEditingController _phoneNumberController = TextEditingController();

  @override
  Widget build(BuildContext context) {
   return Scaffold(
    appBar: AppBar(
     title: Text('Phone Verification'),
     backgroundColor: Colors.green,
    ),
    body: GestureDetector(
     onTap: () {
      // Dismiss the keyboard when tapping outside the text field
      FocusScope.of(context).unfocus();
     },
     child: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Column(
       mainAxisAlignment: MainAxisAlignment.center,
       children: [
        TextField(
         controller: _phoneNumberController,
         keyboardType: TextInputType.phone,
         maxLength: 10,
         decoration: InputDecoration(
          labelText: 'Enter Phone Number',
```

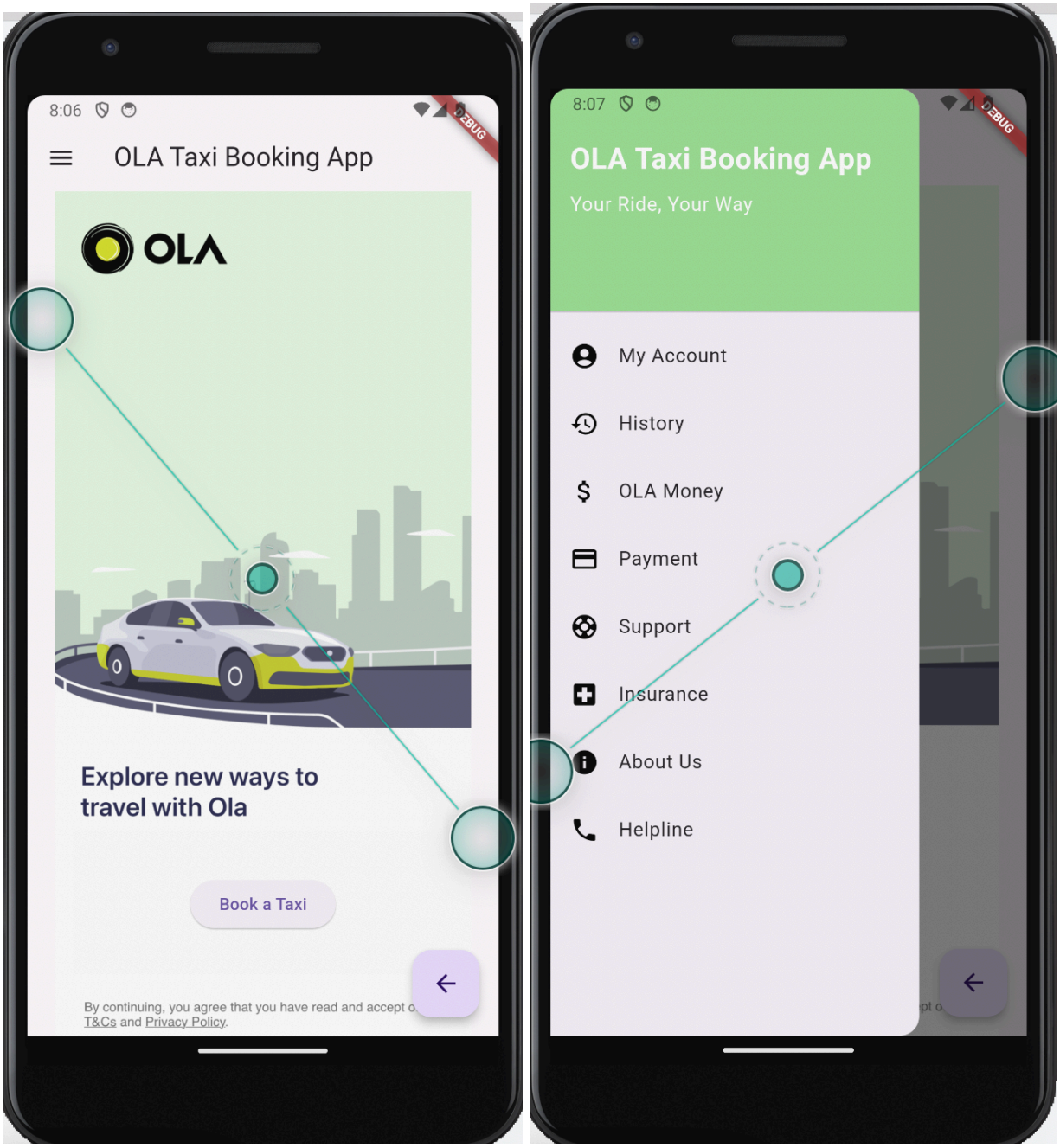
```
            hintText: 'e.g., 1234567890',
            counterText: '',
            enabledBorder: OutlineInputBorder(
              borderSide: BorderSide(color: Colors.green),
            ),
            focusedBorder: OutlineInputBorder(
              borderSide: BorderSide(color: Colors.green, width: 2.0),
            ),
          ),
        ),
        SizedBox(height: 16.0),
        ElevatedButton(
          onPressed: () {
            if (_phoneNumberController.text.length == 10) {
              ScaffoldMessenger.of(context).showSnackBar(
                SnackBar(
                  content: Text('Successfully Verified'),
                  duration: Duration(seconds: 2),
                ),
              );
            } else {
              ScaffoldMessenger.of(context).showSnackBar(
                SnackBar(
                  content: Text('Please enter 10 digits'),
                  duration: Duration(seconds: 2),
                ),
              );
            }
          },
          style: ElevatedButton.styleFrom(
            primary: Colors.green,
            onPrimary: Colors.white,
          ),
          child: Text('Submit'),
        ),
      ],
    ),
  ),
 );
}
```

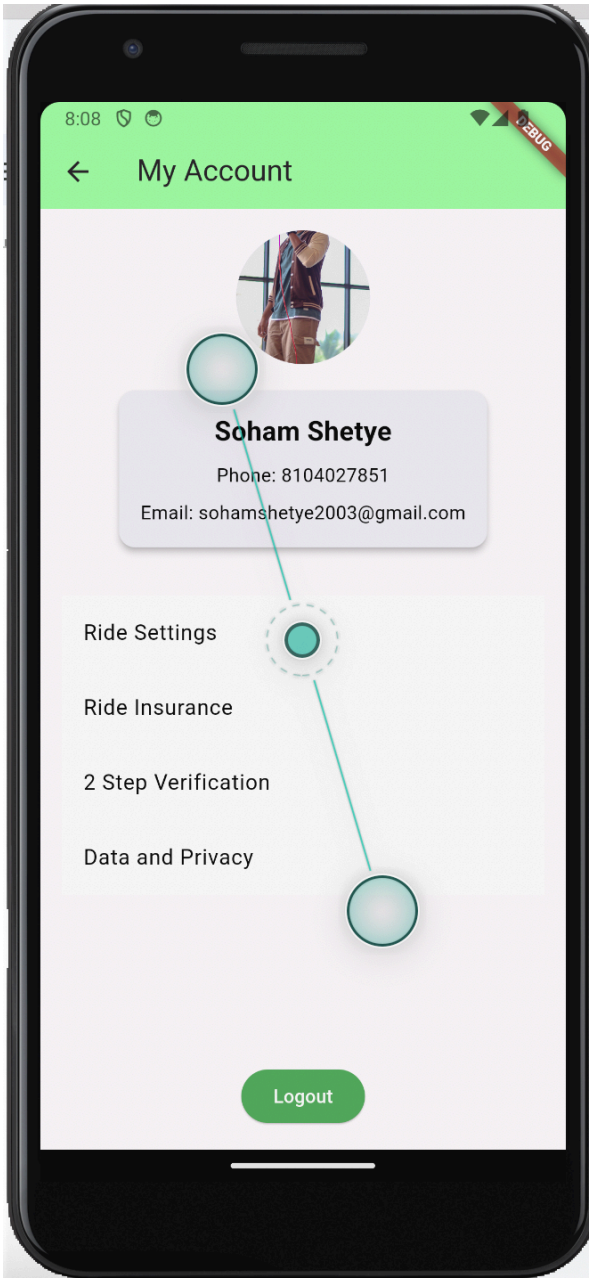
}

Screenshots :




8:03
Register
Registration Page
Name
Phone Number
Email
Gender
Address
Age
Continue to Phone Verification
8:04
Phone Verification
Enter Phone Number
e.g., 1234567890
Submit


8:06
OLA Taxi Booking App
OLA
Explore new ways to travel with Ola
Book a Taxi
By continuing, you agree that you have read and accept o
T&Cs and Privacy Policy.
8:07
OLA Taxi Booking App
Your Ride, Your Way
My Account
History
OLA Money
Payment
Support
Insurance
About Us
Helpline



Navigation and Routing:

Main Route Configuration (MyApp):

- The `MaterialApp` widget is used to define the main navigation configuration.
- The `routes` property is used to define named routes and their corresponding widgets.

Login Page (LoginPage):

- The `ElevatedButton` on the login page is configured to navigate to the `/home` route using `Navigator.pushNamed(context, '/home')`.

Register Page (RegisterPage):

- The `ElevatedButton` on the register page is configured to navigate to the `/phone_verification` route using `Navigator.pushNamed(context, '/phone_verification')`.
- The `TextButton` is configured to navigate back to the previous screen using `Navigator.pop(context)`.

Phone Verification Page (PhoneVerificationPage):

- No explicit navigation in this page, but it is part of the overall route configuration.

Gestures:

Login Page (LoginPage):

- The `GestureDetector` wraps the entire `Container` to handle gestures. The `onTap` callback is used to dismiss the keyboard when tapping outside the text field.

Register Page (RegisterPage):

- The `GestureDetector` wraps the entire `SingleChildScrollView` to handle gestures. The `onTap` callback is used to dismiss the keyboard when tapping outside the text field.

Phone Verification Page (PhoneVerificationPage):

- The `GestureDetector` wraps the entire `Padding` widget to handle gestures. The `onTap` callback is used to dismiss the keyboard when tapping outside the text field.

CONCLUSION :

Navigation and routing are used when transitioning between different pages using named routes (Navigator.pushNamed). Gestures are used to dismiss the keyboard when tapping outside text fields (GestureDetector with onTap callback).