Since 1962

Vivekanand Education Society's

Institute of Technology

(Affiliated to University of Mumbai, Approved by AICTE & Recognized by Govt. of Maharashtra)

Department of Information Technology

A.Y. 22-23

MAD & PWA Lab

Journal

Experiment No.	04
Experiment Title.	To create an interactive Form using form widget
Roll No.	70
Name	MAYURI SHRIDATTA YERANDE
Class	D15B
Subject	MAD & PWA Lab
Lab Outcome	LO2: Design and Develop interactive Flutter App by using widgets, layouts, gestures and animation
Grade:	

EXPERIMENT – 04

AIM: To create an Interactive form using form widget.

THEORY:

- A form widget is provided by the flutter to create forms.
- This form widget acts as a container, which allows us to group multiple form fields.
- To create a form, we have to provide a GlobalKey to uniquely identify the form, which will enable us to validate form fields.
- The form widget uses child widget TextFormField to provide the users to enter the textfield. This widget renders a material design textfield and also allows us to display validation errors when they occur.
- We define a global key as _formKey.
- This key holds a FormState and can use to retrieve the form widget.
- A validator is created to ensure Text is not empty and then we display error message.

```
TextFormField (
    validator : (value) {
      if (value == null || value.isEmpty) {
        return 'Please enter some text';
      }
      return null; },
),
```

- We also check if form is valid or not when the user submits the form and we display success message.

=>
```
ElevatedButton(
  onPressed: () {
    if (_formKey.currentState!.validate()) {
    ScaffoldMessenger.of(context).showSnackBar(
      const SnackBar(content: Text('Processing Data')),
    );
  }
},
```

- The FormState class contains validate() method. When validate() method is called, it runs the validator() function for each text field in the form. If error occurs, then displays respective message, returns false.

CONCLUSION: Thus validation for login, signup and Case Information page is implemented. After entering correct details in sign up page, the user is directed to case information page and then login page. When successfully logged in, the user is directed to main screen. Thus validation for forms in OLA App has been successfully implemented.

EXPERIMENT - 04

AIM: To create an interactive Form using form widget

IMPLEMENTATION:

SCREEN 1: LOGIN SCREEN

CODE:

```
import 'package:drivers_app/authentication/signup_screen.dart';
import 'package:flutter/material.dart';

import '../mainScreens/main_screen.dart';

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

  @override
  State<LoginScreen> createState() => _LoginScreenState();
}

class _LoginScreenState extends State<LoginScreen> {
  TextEditingController nameTextEditingController =
TextEditingController();
  TextEditingController emailTextEditingController =
TextEditingController();
  TextEditingController phoneTextEditingController =
TextEditingController();
  TextEditingController passwordTextEditingController =
TextEditingController();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        elevation: 0,

        backgroundColor: Colors.white,
      ),
      body: MyStatefulWidget(
      ),
    );
```

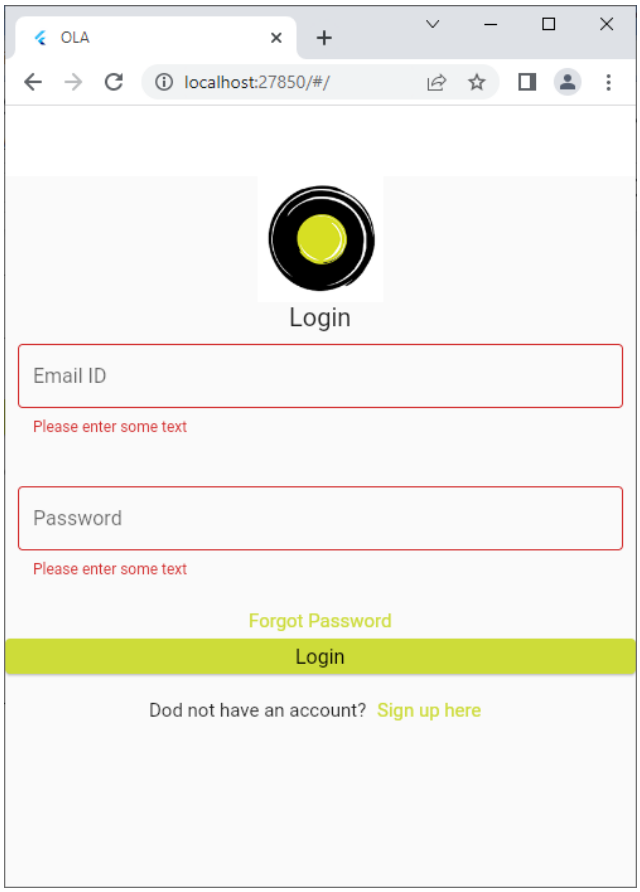
```
  }
}

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

  @override
  State<MyStatefulWidget> createState() => _MyStatefulWidgetState();
}

class _MyStatefulWidgetState extends State<MyStatefulWidget> {
  TextEditingController nameController = TextEditingController();
  TextEditingController passwordController = TextEditingController();
  final _formKey = GlobalKey<FormState>();

  @override
  Widget build(BuildContext context) {
    return Form(

        key: _formKey,
        child: ListView(

          children: <Widget>[
            Container(
                width: 100,
                height: 100,
                decoration: BoxDecoration(
                  image: DecorationImage(
                    image: AssetImage("images/ola.png"),
                  ),
                )
            ),
            Container(
              child: Center(
                child: Text("Login",style:
TextStyle(fontSize:(20)),),
              ),
            ),
            Container(
              padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
              child: Column(
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  TextFormField(
                    controller: nameController,
```

```
        decoration: const InputDecoration(
          border: OutlineInputBorder(),
          labelText: 'Email ID',
        ),
        validator: (value) {
          if (value == null || value.isEmpty) {
            return "Please enter some text";
          }
          return null;
        },
      ),
    ],
  ),
),
Container(
  padding: const EdgeInsets.fromLTRB(10, 30, 10, 20),
  child: TextFormField(
    obscureText: true,
    controller: passwordController,
    decoration: const InputDecoration(
      border: OutlineInputBorder(),
      labelText: 'Password',
    ),
    validator: (value) {
      if (value == null || value.isEmpty) {
        return "Please enter some text";
      }
      return null;
    },
  ),
),
TextButton(
  onPressed: () {
    //forgot password screen
  },
  child: const Text(
    'Forgot Password',
    style: TextStyle(fontSize: (15),color: Colors.lime),
  ),
),
Container(
    child: ElevatedButton(

      style: ElevatedButton.styleFrom(
        backgroundColor: Colors.lime,
```

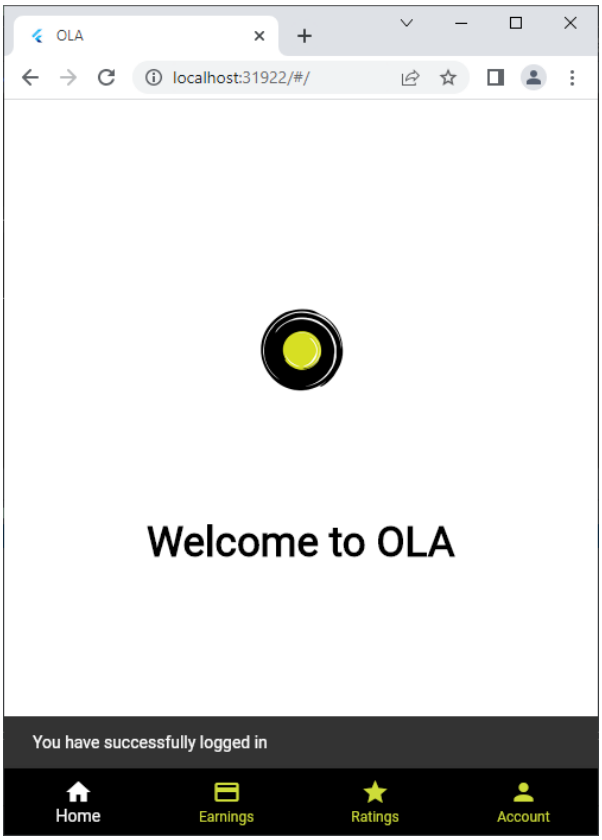
```
                ),

                child: const Text(
                  'Login',
                  style: TextStyle(fontSize: (16),color:
Colors.black),
                ),
                onPressed: () {
                  if (_formKey.currentState!.validate()) {
                    ScaffoldMessenger.of(context).showSnackBar(
                        const SnackBar(content: Text("Processing
data, Your account has been created")));

                  }
                  Navigator.push(context,
MaterialPageRoute(builder: (context) => const MainScreen())
                  );

                  print(nameController.text);
                  print(passwordController.text);
                },

              )),
            const SizedBox(height: 15),
            Row(
              mainAxisAlignment: MainAxisAlignment.center,
              children: <Widget>[
                const Text(
                  'Do not have an account?',
                  style: TextStyle(
                    fontSize: (15),
                  ),
                ),
                TextButton(
                  child: const Text(
                    'Sign up here',
                    style: TextStyle(fontSize: 15, color:
Colors.lime),
                  ),
                  onPressed: () {
                    Navigator.push(
                      context,
                      MaterialPageRoute(builder: (context) => const
SignUpScreen()),
                    );
```

```
                    },
                  ),
                ],
              ),
            ],
          ));
    }
}
```

OUTPUT:

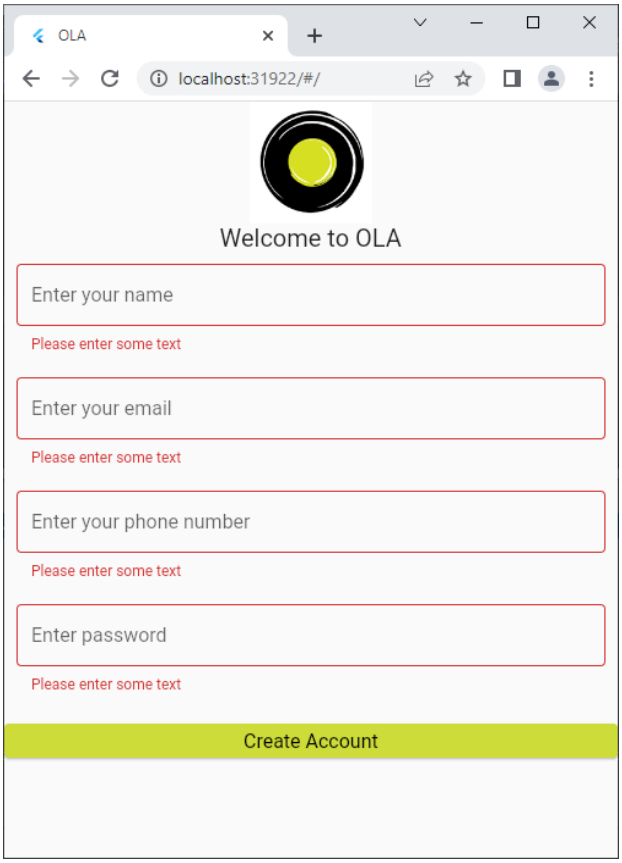


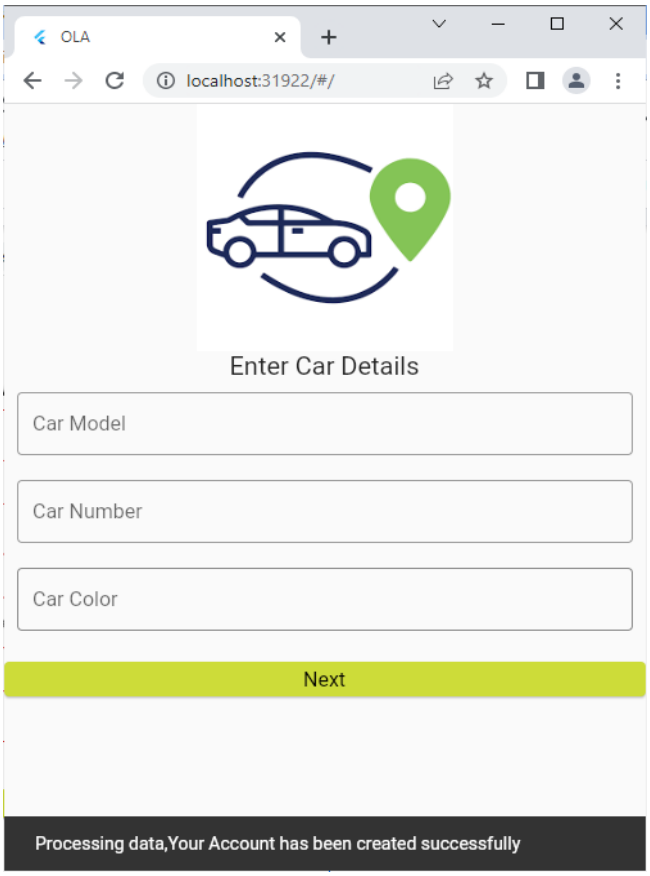
If the user does not enter any text, then a message saying “please enter some text” will be flashed.



Once You have entered the details correctly Then a message will be flashed saying "You have successfully logged in". Thus Validation for Login Page is done.

SCREEN 2: SIGNUP SCREEN:-





SCREEN 3: CAR INFORMATION
CODE:

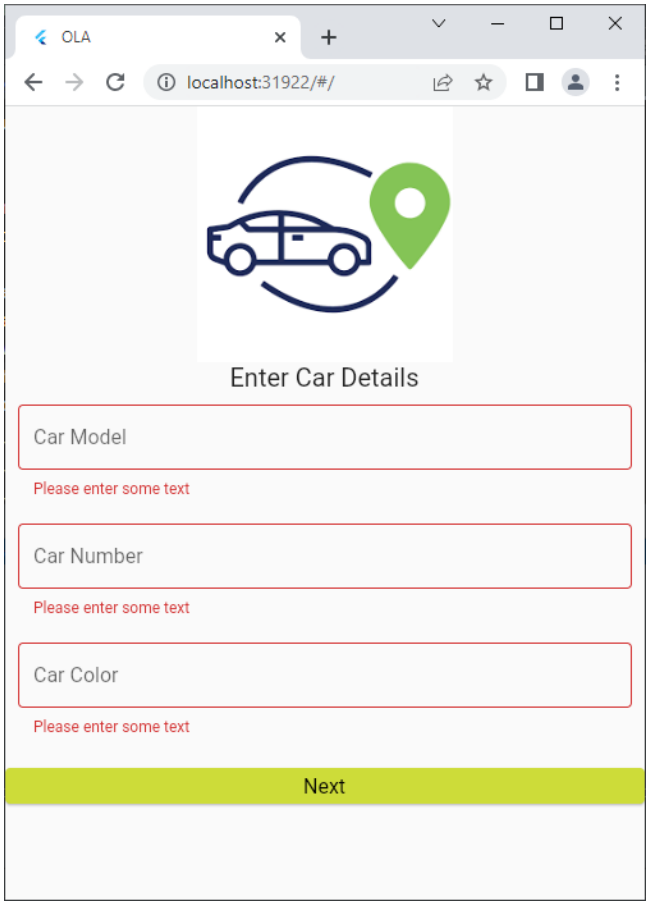
```
import 'package:drivers_app/authentication/signup_screen.dart';
import 'package:flutter/material.dart';

import '../mainScreens/main_screen.dart';
import 'login_screen.dart';

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

  @override
  State<CarInfoScreen> createState() => _CarInfoScreenState();
}

class _CarInfoScreenState extends State<CarInfoScreen> {
  TextEditingController nameTextEditingController =
TextEditingController();
  TextEditingController emailTextEditingController =
TextEditingController();
  TextEditingController phoneTextEditingController =
TextEditingController();
  TextEditingController passwordTextEditingController =
TextEditingController();

  @override
  Widget build(BuildContext context) {
    return Scaffold(

      body: MyStatefulWidget(
      ),
    );
  }
}

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

  @override
  State<MyStatefulWidget> createState() => _MyStatefulWidgetState();
}

class _MyStatefulWidgetState extends State<MyStatefulWidget> {
```

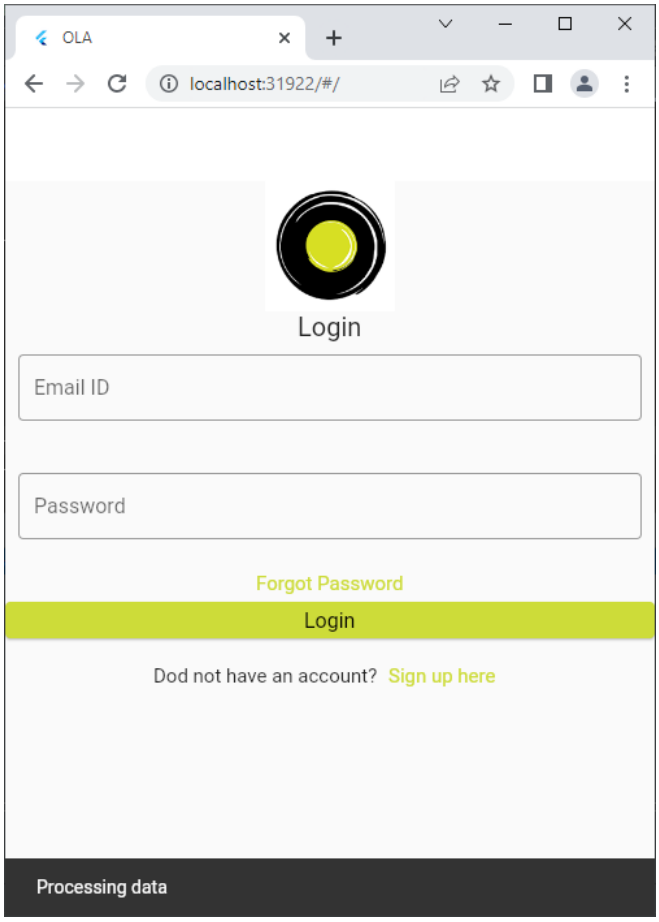
```
  TextEditingController carModelTextEditingController =
TextEditingController();
  TextEditingController carNumberTextEditingController =
TextEditingController();
  TextEditingController carColorTextEditingController =
TextEditingController();
  final _formKey = GlobalKey<FormState>();

  @override
  Widget build(BuildContext context) {
    return Form(

        key: _formKey,
        child: ListView(

          children: <Widget>[
            Container(
                width: 600,
                height: 200,
                decoration: BoxDecoration(
                  image: DecorationImage(
                    image: AssetImage("images/car_info.png"),
                  ),
                )
            ),
            Container(
              child: Center(
                child: Text("Enter Car Details",style:
TextStyle(fontSize:(20)),),
              ),
            ),
            Container(
              padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
              child: Column(
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  TextFormField(
                    controller: carModelTextEditingController,
                    decoration: const InputDecoration(
                      border: OutlineInputBorder(),
                      labelText: 'Car Model',
                    ),
                    validator: (value) {
                      if (value == null || value.isEmpty) {
                        return "Please enter some text";
```

```
          }
          return null;
        },
      ),
    ],
  ),
),
Container(
  padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
  child: TextFormField(
    obscureText: true,
    controller: carNumberTextEditingController,
    decoration: const InputDecoration(
      border: OutlineInputBorder(),
      labelText: 'Car Number',
    ),
    validator: (value) {
      if (value == null || value.isEmpty) {
        return "Please enter some text";
      }
      return null;
    },
  ),
),
Container(
  padding: const EdgeInsets.fromLTRB(10, 10, 10, 10),
  child: TextFormField(
    obscureText: true,
    controller: carColorTextEditingController,
    decoration: const InputDecoration(
      border: OutlineInputBorder(),
      labelText: 'Car Color',
    ),
    validator: (value) {
      if (value == null || value.isEmpty) {
        return "Please enter some text";
      }
      return null;
    },
  ),
),
const SizedBox(height: 15),
Container(
    child: ElevatedButton(
```

```
                    style: ElevatedButton.styleFrom(
                      backgroundColor: Colors.lime,
                    ),

                    child: const Text(
                      'Next',
                      style: TextStyle(fontSize: (16),color:
Colors.black),
                    ),
                    onPressed: () {
                      if (_formKey.currentState!.validate()) {
                       ScaffoldMessenger.of(context).showSnackBar(
                           const SnackBar(content: Text("Processing
data")));
                        Navigator.push(context,
MaterialPageRoute(builder: (context) => const LoginScreen())
                        );
                      }



                    },

                  )),
              const SizedBox(height: 15),

            ],
          ));
    }
}
```

OUTPUT:





CONCLUSION: Thus Validation for Login,Signup and Car Information page is implemented. After entering correct details in the Sign Up page, the user is directed to the Car Information Page and then to the Login page. When successfully Logged in, the user is directed to the Main Screen. Thus Validation for forms in OLA App has been successfully implemented.