MAD and PWA Lab

Name: Sarvadnya Awaghad   Class: D15A   Roll no:04

EXP 4

Aim : To create an interactive Form using form widget

Theory:

Form Widgets:

Form widgets are essential components of interactive forms, offering a range of input elements such as text fields, checkboxes, radio buttons, dropdown menus, and more. These widgets empower developers to design forms that cater to specific data input requirements. The flexibility of form widgets allows for the creation of dynamic and user-friendly interfaces, ensuring that the form adapts to the user's needs.

Form Inputs:

Text Fields:

Purpose: Allow users to input general text information.
Attributes: May include specifications such as maximum length, placeholder text, and input type (e.g., email, password).

Checkboxes:

Purpose: Enable users to make multiple selections from a list of options.
Attributes: Each checkbox typically represents a distinct option, and users can choose multiple checkboxes simultaneously.

Radio Buttons:

Purpose: Provide users with exclusive choices within a group.
Attributes: Users can select only one option from the group, making radio buttons suitable for mutually exclusive selections.

Dropdown Menus:

Purpose: Offer a space-efficient way to present a list of options for selection.
Attributes: Users click on a dropdown menu to reveal a list of choices, selecting one option from the list.

Textareas:

Purpose: Allow users to input multiline text, suitable for longer responses or comments.
Attributes: Can include settings for the number of rows and columns to determine the size of the textarea.

Date Pickers:

Purpose: Facilitate the selection of dates.
Attributes: Users can choose a specific date from a calendar interface, helping to ensure accurate date input.

File Upload:

Purpose: Enable users to submit files (e.g., images, documents).
Attributes: May include file type restrictions, maximum file size, and a browse button for users to locate and upload files from their device.

Code

```dart
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'package:tinder/authenticationScreen/registration_screen.dart';
import 'package:tinder/widgets/custom_text_field_widget.dart';

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

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

class _LoginScreenState extends State<LoginScreen> {
 TextEditingController emailTextEditingController = TextEditingController();
 TextEditingController passwordTextEditingController = TextEditingController();
 bool showProgressBar = false;

 @override
 Widget build(BuildContext context) {
  return Scaffold(
   body: Container(
    decoration: BoxDecoration(
     gradient: LinearGradient(
      colors: [Colors.pink, Colors.orange],
      begin: Alignment.topLeft,
      end: Alignment.bottomRight,
     ),
    ),
    child: SingleChildScrollView(
     child: Center(
      child: Column(
       children: [
        const SizedBox(
         height: 220,
        ),
        Image.asset(
         'images/tinder_logo.png',
         width: 100,
        ),
        const SizedBox(
         height: 60,
        ),
        Text(
         "By clicking Login",
         style: TextStyle(
          fontSize: 13,
         ),
        ),
        Text(
         " you agree with our terms and conditions.",
         style: TextStyle(
```

```dart
      fontSize: 13,
    ),
  ),
  Text(
    " Learn how we process your data in Privacy Policy and cookies policy",
    style: TextStyle(
      fontSize: 12,
    ),
  ),
  const SizedBox(
    height: 25,
  ),
  GestureDetector(
    onTap: () {
      // Implement your Google login logic here
    },
    child: Container(
      width: MediaQuery.of(context).size.width - 36,
      height: 55,
      decoration: BoxDecoration(
        color: Colors.white,
        borderRadius: BorderRadius.circular(10),
      ),
      child: Row(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          const SizedBox(width: 25),
          Image.asset(
            'images/google.jpeg', // Replace with the actual path to your Google logo image
            width: 24, // Adjust the width as needed
            height: 24, // Adjust the height as needed
          ),
          const SizedBox(width: 15),
          Text(
            "Continue with Google",
            style: TextStyle(
              fontSize: 20,
              color: Colors.black,
              fontWeight: FontWeight.bold,
            ),
          ),
        ],
      ),
    ),
  ),
  const SizedBox(
    height: 25,
  ),
  SizedBox(
    width: MediaQuery.of(context).size.width - 36,
    height: 55,
    child: CustomTextFieldWidget(
```

```dart
      editingController: emailTextEditingController,
      labelText: 'Email',
      iconData: Icons.email_outlined,
      isObscure: false,
    ),
  ),
  const SizedBox(
    height: 20,
  ),
  SizedBox(
    width: MediaQuery.of(context).size.width - 36,
    height: 55,
    child: CustomTextFieldWidget(
      editingController: passwordTextEditingController,
      labelText: 'Password',
      iconData: Icons.lock_outline,
      isObscure: true,
    ),
  ),
  const SizedBox(
    height: 20,
  ),
  GestureDetector(
    onTap: () {
      // Implement your login logic here
      // For example, you can use Get.to(HomeScreen());
    },
    child: Container(
      width: MediaQuery.of(context).size.width - 36,
      height: 55,
      decoration: BoxDecoration(
        gradient: LinearGradient(
          colors: [Colors.pink, Colors.orange],
          begin: Alignment.topLeft,
          end: Alignment.bottomRight,
        ),
        borderRadius: BorderRadius.circular(10),
      ),
      child: const Center(
        child: Text(
          "Login",
          style: TextStyle(
            fontSize: 20,
            color: Colors.white,
            fontWeight: FontWeight.bold,
          ),
        ),
      ),
    ),
  ),
  const SizedBox(
    height: 20,
```

```dart
        ),
        Row(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            Text(
              'Don\'t have an Account?',
              style: TextStyle(
                fontSize: 16,
                color: Colors.black,
              ),
            ),
            InkWell(
              onTap: () {
                Get.to(RegistrationScreen());
              },
              child: const Text(
                'Create Here',
                style: TextStyle(
                  fontSize: 16,
                  color: Colors.black,
                  fontWeight: FontWeight.bold,
                ),
              ),
            )
          ],
        ),
        const SizedBox(
          height: 40,
        ),
        showProgressBar
            ? const CircularProgressIndicator(
                valueColor: AlwaysStoppedAnimation<Color>(Colors.pink),
              )
            : Container()
      ],
    ),
  ),
),
);
}
}
```
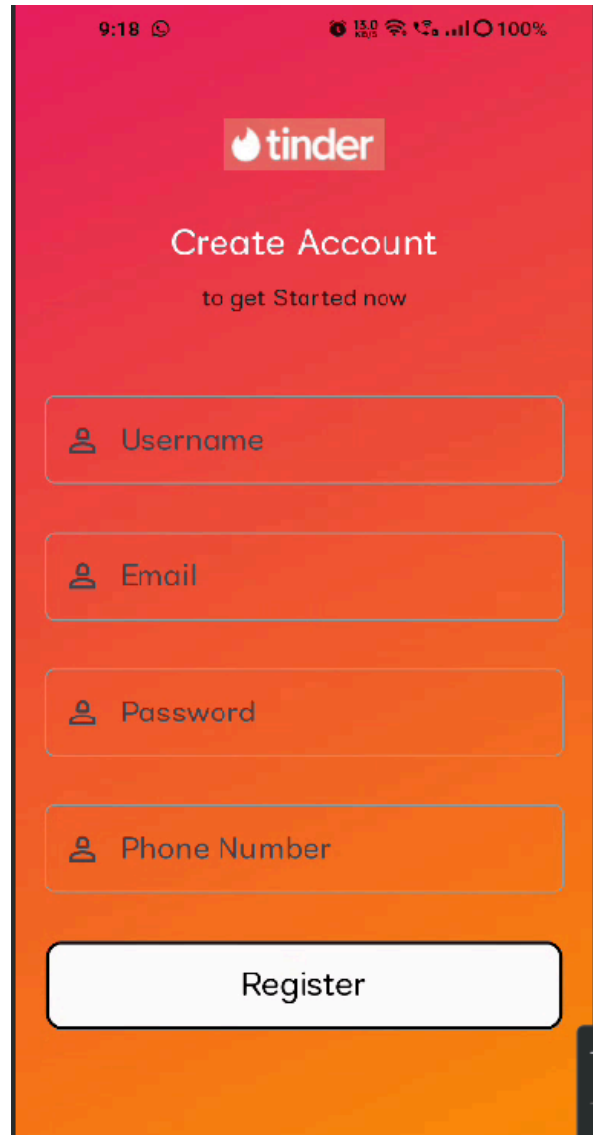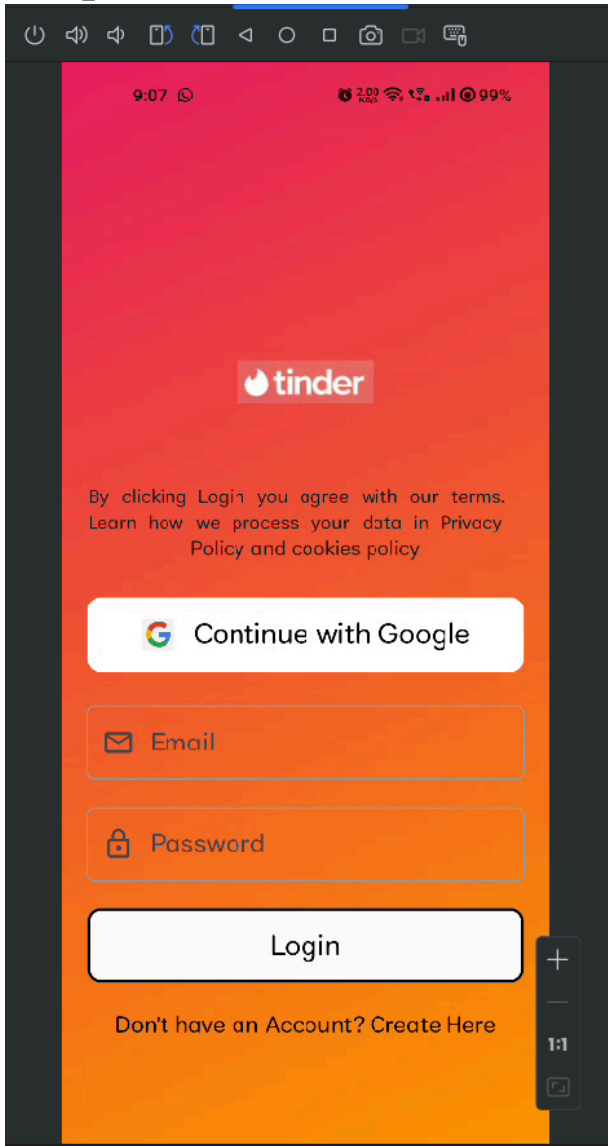
Output:

Conclusion: In this experiment , we have successfully created form using form widget and create a login page for my clone application, various properties of form are implemented successfully in the above experiment.