

## **MAD LAB EXPT 5**

**Name:Swarali Dhobale\_D15A\_13**

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

### **Output:**

```
import 'package:amazon_clone/resources/authentication_methods.dart';
import 'package:amazon_clone/screens/sign_up_screen.dart';
import 'package:amazon_clone/utils/color_themes.dart';
import 'package:amazon_clone/utils/constants.dart';
import 'package:amazon_clone/utils/utils.dart';
import 'package:amazon_clone/widgets/custom_main_button.dart';
import 'package:amazon_clone/widgets/text_field_widget.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

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

  @override
  State<SignInScreen> createState() => _SignInScreenState();
}

class _SignInScreenState extends State<SignInScreen> {
  TextEditingController emailController = TextEditingController();
  TextEditingController passwordController = TextEditingController();
  AuthenticationMethods authenticationMethods = AuthenticationMethods();
  bool isLoading = false;

  @override
  void dispose() {
    super.dispose();
    emailController.dispose();
    passwordController.dispose();
  }

  @override
  Widget build(BuildContext context) {
    Size screenSize = Utils().getScreenSize();
    return Scaffold(
      backgroundColor: Colors.white,
      body: SingleChildScrollView(
```

```
child: SizedBox(
  height: screenSize.height,
  width: screenSize.width,
  child: Padding(
    padding: const EdgeInsets.symmetric(horizontal: 10, vertical: 20),
    child: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.spaceEvenly,
        crossAxisAlignment: CrossAxisAlignment.center,
        children: [
          Image.network(
            amazonLogo,
            height: screenSize.height * 0.10,
          ),
          Container(
            height: screenSize.height * 0.6,
            width: screenSize.width * 0.8,
            padding: const EdgeInsets.all(25),
            decoration: BoxDecoration(
              border: Border.all(
                color: Colors.grey,
                width: 1,
              ),
            ),
          ),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.min,
            mainAxisAlignment: MainAxisAlignment.spaceBetween,
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              const Text(
                "Sign-In",
                style: TextStyle(
                  fontWeight: FontWeight.w500, fontSize: 33),
              ),
              TextFieldWidget(
                title: "Email",
                controller: emailController,
                obscureText: false,
                hintText: "Enter your email",
              ),
              TextFieldWidget(
                title: "Password",
                controller: passwordController,
                obscureText: true,
```

```

        hintText: "Enter your password",
      ),
      Align(
        alignment: Alignment.center,
        child: CustomMainButton(
          child: const Text(
            "Sign In",
            style: TextStyle(
              letterSpacing: 0.6, color: Colors.black),
          ),
          color: yellowColor,
          isLoading: isLoading,
          onPressed: () async {
            setState(() {
              isLoading = true;
            });

            String output =
              await authenticationMethods.signInUser(
                email: emailController.text,
                password: passwordController.text);
            setState(() {
              isLoading = false;
            });
            if (output == "success") {
              //functions
            } else {
              //error
              Utils().showSnackBar(
                context: context, content: output);
            }
          }
        ),
      ],
    ),
  ),
  Row(
    children: [
      Expanded(
        child: Container(
          height: 1,
          color: Colors.grey,
        ),
      ),
    ],
  ),

```

```

const Padding(
  padding: EdgeInsets.symmetric(horizontal: 10),
  child: Text(
    "New to Amazon?",
    style: TextStyle(color: Colors.grey),
  ),
),
Expanded(
  child: Container(
    height: 1,
    color: Colors.grey,
  ),
),
],
),
CustomMainButton(
  child: const Text(
    "Create an Amazon Account",
    style: TextStyle(
      letterSpacing: 0.6,
      color: Colors.black,
    ),
  ),
),
color: Colors.grey[400]!,
isLoading: false,
onPressed: () {
  Navigator.pushReplacement(context,
    MaterialPageRoute(builder: (context) {
      return const SignUpScreen();
    }));
})
],
),
),
),
),
),
);
}
}

```

```

class UserDetailsBar extends StatelessWidget {
  final double offset;
  const UserDetailsBar({
    Key? key,
    required this.offset,
  }) : super(key: key);

  @override
  Widget build(BuildContext context) {
    Size screenSize = Utils().getScreenSize();
    UserDetailsModel userDetails =
      Provider.of<UserDetailsProvider>(context).userDetails;
    return Positioned(
      top: -offset / 3,
      child: Container(
        height: kAppBarHeight / 2,
        width: screenSize.width,
        decoration: const BoxDecoration(
          gradient: LinearGradient(
            colors: lightBackgroundaGradient,
            begin: Alignment.centerLeft,
            end: Alignment.centerRight,
          ),
        ),
      child: Padding(
        padding: const EdgeInsets.symmetric(
          vertical: 3,
          horizontal: 20,
        ),
        child: Row(
          children: [
            Padding(
              padding: const EdgeInsets.only(right: 8.0),
              child: Icon(
                Icons.location_on_outlined,
                color: Colors.grey[900],
              ),
            ),
            SizedBox(
              width: screenSize.width * 0.7,
              child: Text(
                "Deliver to ${userDetails.name} - ${userDetails.address} ",
                maxLines: 1,
                overflow: TextOverflow.ellipsis,
              ),
            ),
          ],
        ),
      ),
    );
  }
}

```

```

        style: TextStyle(
          color: Colors.grey[900],
        ),
      ),
    ),
  ],
),
),
),
);
}
}

```

## **SIGN OUT-**

```

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

  @override
  State<AccountScreen> createState() => _AccountScreenState();
}

```

```

class _AccountScreenState extends State<AccountScreen> {
  @override
  Widget build(BuildContext context) {
    Size screenSize = Utils().getScreenSize();

    return Scaffold(
      backgroundColor: Colors.white,
      appBar: AccountScreenAppBar(),
      body: SingleChildScrollView(
        child: SizedBox(
          height: screenSize.height,
          width: screenSize.width,
          child: Column(
            children: [
              IntroductionWidgetAccountScreen(),
              Padding(
                padding: const EdgeInsets.all(8.0),
                child: CustomMainButton(
                  child: const Text(
                    "Sign Out",
                    style: TextStyle(color: Colors.black),
                  ),
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }
}

```

```

        color: Colors.orange,
        isLoading: false,
        onPressed: () {
          FirebaseAuth.instance.signOut();
        }},
      ),

```

```

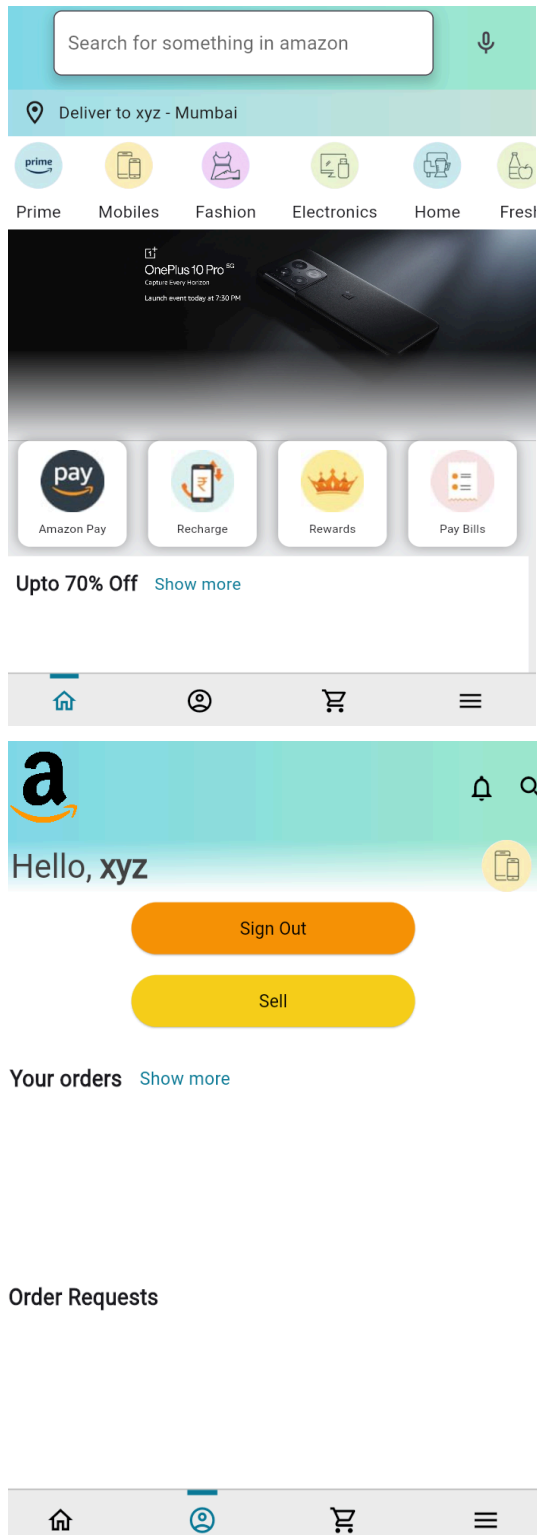
    Padding(
      padding: const EdgeInsets.all(8.0),
      child: CustomMainButton(
        child: const Text("Sell",
          style: TextStyle(color: Colors.black)),
        color: yellowColor,
        isLoading: false,
        onPressed: () {
          Navigator.push(
            context,
            MaterialPageRoute(
              builder: (context) => const SellScreen());
          }},
      ),

```

```

    FutureBuilder(
      future: FirebaseFirestore.instance
        .collection("users")
        .doc(FirebaseAuth.instance.currentUser!.uid)
        .collection("orders")
        .get(),
      builder: (context,
        AsyncSnapshot<QuerySnapshot<Map<String, dynamic>>>
          snapshot) {
        if (snapshot.connectionState == ConnectionState.waiting) {
          return Container();
        } else {
          List<Widget> children = [];
          for (int i = 0; i < snapshot.data!.docs.length; i++) {
            ProductModel model = ProductModel.getModelFromJson(
              json: snapshot.data!.docs[i].data());
            children.add(SimpleProductWidget(productModel: model));
          }
          return ProductsShowcaseListView(
            title: "Your orders", children: children);
        }
      }
    ),

```



**Conclusion: Navigation, routing and gestures have been applied in Flutter Application.**



