PRANAV RAIKAR

D15A 47

Experiment 5

AIM: To apply navigation, routing and gestures in Flutter App

Theory:

In Flutter, applying navigation, routing, and gestures are essential aspects of creating a smooth and intuitive user experience in your app. Let's break down each of these concepts:

Navigation and Routing:

* Navigation in Flutter involves moving between different screens or "routes" within your app.
* Routing is the mechanism by which you define and manage these routes.
* Flutter provides a Navigator class to handle navigation and route management.
* You can use named routes to define the routes in your app, making it easier to navigate between them.
* Navigation and routing are crucial for maintaining a smooth user flow and organizing the content of your app effectively.

Gestures:

* Gestures refer to user interactions such as tapping, swiping, pinching, etc.
* Flutter offers a variety of gesture recognizers to detect and respond to these user actions.
* Common gesture recognizers include GestureDetector, InkWell, DragGestureRecognizer, ScaleGestureRecognizer, etc.
* Gestures play a significant role in making your app interactive and responsive to user input.
* By implementing appropriate gestures, you can enhance the usability and engagement of your app.

Combining Navigation and Gestures:

* Combining navigation and gestures allows you to create interactive user interfaces in your Flutter app.
* For example, you can navigate to a different screen when a specific gesture, such as tapping or swiping, is detected.
* By integrating navigation with gestures, you can create seamless user experiences that are both intuitive and engaging.
* It's essential to consider the context and purpose of each gesture to ensure it aligns with the overall navigation flow of your app.

Code:

Home\_admin:

import 'package:flutter/material.dart';

import 'package:food\_panda/admin/add\_food.dart';

import 'package:food\_panda/widget/widget\_support.dart';

// import 'package:sample\_flutter/admin/add\_food.dart';

// import 'package:sample\_flutter/widget/widget\_support.dart';

class HomeAdmin extends StatefulWidget {

const HomeAdmin({super.key});

@override

State<HomeAdmin> createState() => \_HomeAdminState();

}

class \_HomeAdminState extends State<HomeAdmin> {

@override

Widget build(BuildContext context) {

return Scaffold(

body: Container(

margin: EdgeInsets.only(top: 50.0, left: 20.0, right: 20.0),

child: Column(

children: [

Center(

child: Text(

"Home Admin",

style: AppWidget.HeadlineTextFieldStyle(),

),

),

SizedBox(

height: 50.0,

),

**GestureDetector(**

**onTap: () {**

**Navigator.push(context,**

**MaterialPageRoute(builder: (context) => AddFood()));**

**},**

child: Material(

elevation: 10.0,

borderRadius: BorderRadius.circular(10),

child: Center(

child: Container(

padding: EdgeInsets.all(4),

decoration: BoxDecoration(

color: Colors.black,

borderRadius: BorderRadius.circular(10),

),

child: Row(

children: [

Padding(

padding: EdgeInsets.all(6.0),

child: Image.asset(

"images/food.jpg",

height: 100,

width: 100,

fit: BoxFit.cover,

),

),

SizedBox(

width: 30.0,

),

Text(

"Add Food Items",

style: TextStyle(

color: Colors.white,

fontSize: 20.0,

fontWeight: FontWeight.bold),

)

],

),

),

),

),

)

],

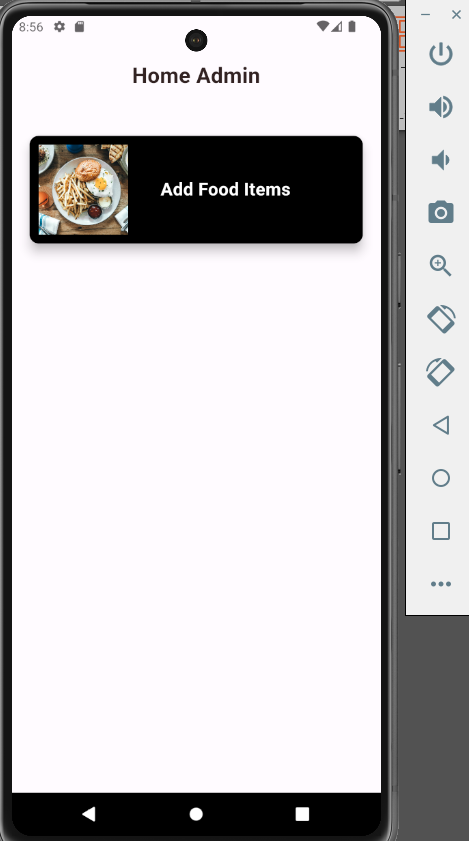
),

),

);

}

}



**Add\_food**:

Code:

import 'dart:io';

import 'package:firebase\_storage/firebase\_storage.dart';

import 'package:flutter/material.dart';

import 'package:food\_panda/service/database.dart';

import 'package:food\_panda/widget/widget\_support.dart';

import 'package:image\_picker/image\_picker.dart';

import 'package:random\_string/random\_string.dart';

// import 'package:sample\_flutter/widget/widget\_support.dart';

class AddFood extends StatefulWidget {

const AddFood({super.key});

@override

State<AddFood> createState() => \_AddFoodState();

}

class \_AddFoodState extends State<AddFood> {

final List<String> fooditems = ['Ice-cream', 'Burger', 'Salad', 'Pizza'];

String? value;

TextEditingController namecontroller = new TextEditingController();

TextEditingController pricecontroller = new TextEditingController();

TextEditingController detailcontroller = new TextEditingController();

final ImagePicker \_picker = ImagePicker();

File? selectedImage;

Future getImage() async {

var image = await \_picker.pickImage(source: ImageSource.gallery);

selectedImage = File(image!.path);

setState(() {});

}

uploadItem() async {

if (selectedImage != null &&

namecontroller.text != "" &&

pricecontroller.text != "" &&

detailcontroller.text != "") {

String addId = randomAlphaNumeric(10);

Reference firebaseStorageRef =

FirebaseStorage.instance.ref().child("blogImages").child(addId);

final UploadTask task = firebaseStorageRef.putFile(selectedImage!);

var downloadUrl = await (await task).ref.getDownloadURL();

Map<String, dynamic> addItem = {

"Image": downloadUrl,

"Name": namecontroller.text,

"Price": pricecontroller.text,

"Detail": detailcontroller.text

};

await DatabaseMethods().addFoodItem(addItem, value!).then((value) {

ScaffoldMessenger.of(context).showSnackBar((SnackBar(

backgroundColor: Colors.redAccent,

content: Text(

"Food item added successfully",

style: TextStyle(fontSize: 20.0),

))));

});

}

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

**leading: GestureDetector(**

**onTap: () {**

**Navigator.pop(context);**

**},**

child: Icon(

Icons.arrow\_back\_ios\_new\_outlined,

color: Color(0xFF373866),

)),

centerTitle: true,

title: Text(

"Add Item",

style: AppWidget.HeadlineTextFieldStyle(),

),

),

body: SingleChildScrollView(

child: Container(

margin:

EdgeInsets.only(left: 20.0, right: 20.0, top: 20.0, bottom: 50.0),

child: Column(

crossAxisAlignment: CrossAxisAlignment.start,

children: [

Text(

"Upload the Item Picture",

style: AppWidget.semiBoldTextFieldStyle(),

),

SizedBox(

height: 20.0,

),

selectedImage == null

? GestureDetector(

onTap: () {

getImage();

},

child: Center(

child: Material(

elevation: 4.0,

borderRadius: BorderRadius.circular(20),

child: Container(

width: 150,

height: 150,

decoration: BoxDecoration(

border:

Border.all(color: Colors.black, width: 1.5),

borderRadius: BorderRadius.circular(20),

),

child: Icon(

Icons.camera\_alt\_outlined,

color: Colors.black,

),

),

),

),

)

: Center(

child: Material(

elevation: 4.0,

borderRadius: BorderRadius.circular(20),

child: Container(

width: 150,

height: 150,

decoration: BoxDecoration(

border: Border.all(color: Colors.black, width: 1.5),

borderRadius: BorderRadius.circular(20),

),

child: Image.file(

selectedImage!,

fit: BoxFit.cover,

),

),

),

),

SizedBox(

height: 30.0,

),

Text(

"Item Name",

style: AppWidget.semiBoldTextFieldStyle(),

),

SizedBox(

height: 10.0,

),

Container(

padding: EdgeInsets.symmetric(horizontal: 20.0),

width: MediaQuery.of(context).size.width,

decoration: BoxDecoration(

color: Color(0xFFececf8),

borderRadius: BorderRadius.circular(10)),

child: TextField(

controller: namecontroller,

decoration: InputDecoration(

border: InputBorder.none,

hintText: "Enter Item Name",

hintStyle: AppWidget.LightTextFieldStyle()),

),

),

SizedBox(

height: 30.0,

),

Text(

"Item Price",

style: AppWidget.semiBoldTextFieldStyle(),

),

SizedBox(

height: 10.0,

),

Container(

padding: EdgeInsets.symmetric(horizontal: 20.0),

width: MediaQuery.of(context).size.width,

decoration: BoxDecoration(

color: Color(0xFFececf8),

borderRadius: BorderRadius.circular(10)),

child: TextField(

controller: pricecontroller,

decoration: InputDecoration(

border: InputBorder.none,

hintText: "Enter Item Price",

hintStyle: AppWidget.LightTextFieldStyle()),

),

),

SizedBox(

height: 30.0,

),

Text(

"Item Detail",

style: AppWidget.semiBoldTextFieldStyle(),

),

SizedBox(

height: 10.0,

),

Container(

padding: EdgeInsets.symmetric(horizontal: 20.0),

width: MediaQuery.of(context).size.width,

decoration: BoxDecoration(

color: Color(0xFFececf8),

borderRadius: BorderRadius.circular(10)),

child: TextField(

maxLines: 6,

controller: detailcontroller,

decoration: InputDecoration(

border: InputBorder.none,

hintText: "Enter Item Detail",

hintStyle: AppWidget.LightTextFieldStyle()),

),

),

SizedBox(

height: 20.0,

),

Text(

"Select Category",

style: AppWidget.semiBoldTextFieldStyle(),

),

SizedBox(

height: 20.0,

),

Container(

padding: EdgeInsets.symmetric(horizontal: 10.0),

width: MediaQuery.of(context).size.width,

decoration: BoxDecoration(

color: Color(0xFFececf8),

borderRadius: BorderRadius.circular(10)),

child: DropdownButtonHideUnderline(

child: DropdownButton<String>(

items: fooditems

.map((item) => DropdownMenuItem<String>(

value: item,

child: Text(

item,

style:

TextStyle(fontSize: 18.0, color: Colors.black),

)))

.toList(),

onChanged: ((value) => setState(() {

this.value = value;

})),

dropdownColor: Colors.white,

hint: Text("Select Category"),

iconSize: 36,

icon: Icon(

Icons.arrow\_drop\_down,

color: Colors.black,

),

value: value,

)),

),

SizedBox(

height: 30.0,

),

GestureDetector(

onTap: () {

uploadItem();

},

child: Center(

child: Material(

elevation: 5.0,

borderRadius: BorderRadius.circular(10),

child: Container(

padding: EdgeInsets.symmetric(vertical: 5.0),

width: 150,

decoration: BoxDecoration(

color: Colors.black,

borderRadius: BorderRadius.circular(10)),

child: Center(

child: Text(

"Add",

style: TextStyle(

color: Colors.white,

fontSize: 22.0,

fontWeight: FontWeight.bold),

),

),

),

),

),

)

],

),

),

),

);

}

}

A screenshot of a phone

Description automatically generated

Conclusion :Thus I learnt to create and use common widgets