

**Name :- Hertika Batra**

**Roll:-5**

**Batch:-A**

### **Experiment:5**

**Aim:- To Apply Navigation ,Routing and gestures in Flutter App**

#### **Theory:-**

In Flutter, navigation, routing, and gestures are like the directions and actions that help users explore and interact with an app easily. Navigation is about moving between different parts of the app, like switching from the homepage to a settings page. Routing is the behind-the-scenes logic that decides which page to show next when a user taps a button or performs an action. Flutter provides tools to manage this process smoothly, so users can navigate through the app without confusion. Gestures, on the other hand, are the touch-based movements users make on the screen, like swiping or tapping. Flutter gives developers ways to recognize these gestures and respond accordingly, making the app feel more interactive and fun to use. So, by using Flutter's navigation, routing, and gesture features, developers can create apps that are easy to explore and enjoyable to interact with.

#### **Code:-**

Recent files:-

```
import 'package:driveclone/controllers/files_screen_controller.dart';  
import 'package:driveclone/utlis.dart';  
import 'package:flutter/material.dart';  
import 'package:get/get_state_manager/src/rx_flutter/rx_getx_widget.dart';  
import 'package:driveclone/widgets/storage_container.dart';
```

```
class RecentFiles extends StatelessWidget {  
  @override
```

```

Widget build(BuildContext context) {
  return Column(
    children: [
      Align(
        alignment: Alignment.centerLeft,
        child: Text(
          "Recent Files",
          style: TextStyle(20, textColor, FontWeight.bold),
        ),
      ),
      const SizedBox(height: 15),
     GetX<FilesScreenController>(
        builder: (FilesScreenController controller) {
          return Container(
            height: 100,
            child: ListView.builder(
              scrollDirection: Axis.horizontal,
              itemCount: controller.recentfileList.length,
              itemBuilder: (context, index) {
                return Padding(
                  padding: const EdgeInsets.only(right: 13.0),
                  child: Container(
                    height: 65,
                    child: Column(
                      crossAxisAlignment: CrossAxisAlignment.start,
                      children: [
                        controller.recentfileList[index].fileType == "image"
                          ? ClipRRect(
                              borderRadius: BorderRadius.circular(18),
                              child: Image(

```

```

        width: 65,
        height: 60,
        image: NetworkImage(
            controller.recentfileList[index].url),
        fit: BoxFit.cover,
    ),
)
: Container(
    width: 65,
    height: 60,
    decoration: BoxDecoration(
        border: Border.all(
            color: Colors.grey, width: 0.15),
        borderRadius: BorderRadius.circular(14)),
    child: Center(
        child: Image(
            width: 42,
            height: 42,
            image: AssetImage(
                'assets/${controller.recentfileList[index].fileExtension}.png'),
        ),
    ),
),
const SizedBox(height: 5),
Text(
    controller.recentfileList[index].name,
    style: textStyle(13, textColor, FontWeight.w500),
    overflow: TextOverflow.ellipsis,
)
],

```

```

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

```

Folder Section:-

```

import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:driveclone/screens/display_files_screen.dart';
import 'package:driveclone/utils.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import 'package:driveclone/controllers/files_screen_controller.dart';
// import 'package:driveclone/controllers/files_controller.dart';
import 'package:driveclone/widgets/storage_container.dart';

class FoldersSection extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return GetX<FilesScreenController>(
      builder: (FilesScreenController folderController) {
        if (folderController != null && folderController.foldersList != null) {

```

```

return GridView.builder(
  shrinkWrap: true,
  physics: const NeverScrollableScrollPhysics(),
  itemCount: folderController.foldersList.length,
  gridDelegate: const SliverGridDelegateWithFixedCrossAxisCount(
    crossAxisCount: 2,
    crossAxisSpacing: 20,
    mainAxisSpacing: 20,
  ),
  itemBuilder: (context, index) {
    return InkWell(
      onTap: () {
        if (FirebaseAuth.instance.currentUser != null) {
          Get.to(DisplayFilesScreen(
            folderController.foldersList[index].name, "folder"
          ));
        } else {
          // Handle case when user is not authenticated
          // You can show a snackbar or navigate to a login screen
          ScaffoldMessenger.of(context).showSnackBar(
            SnackBar(content: Text('Please sign in to access folders'))
          );
        }
      },
      child: Container(
        decoration: BoxDecoration(
          borderRadius: BorderRadius.circular(10),
          boxShadow: [
            BoxShadow(
              color: Colors.grey.withOpacity(0.00001),

```

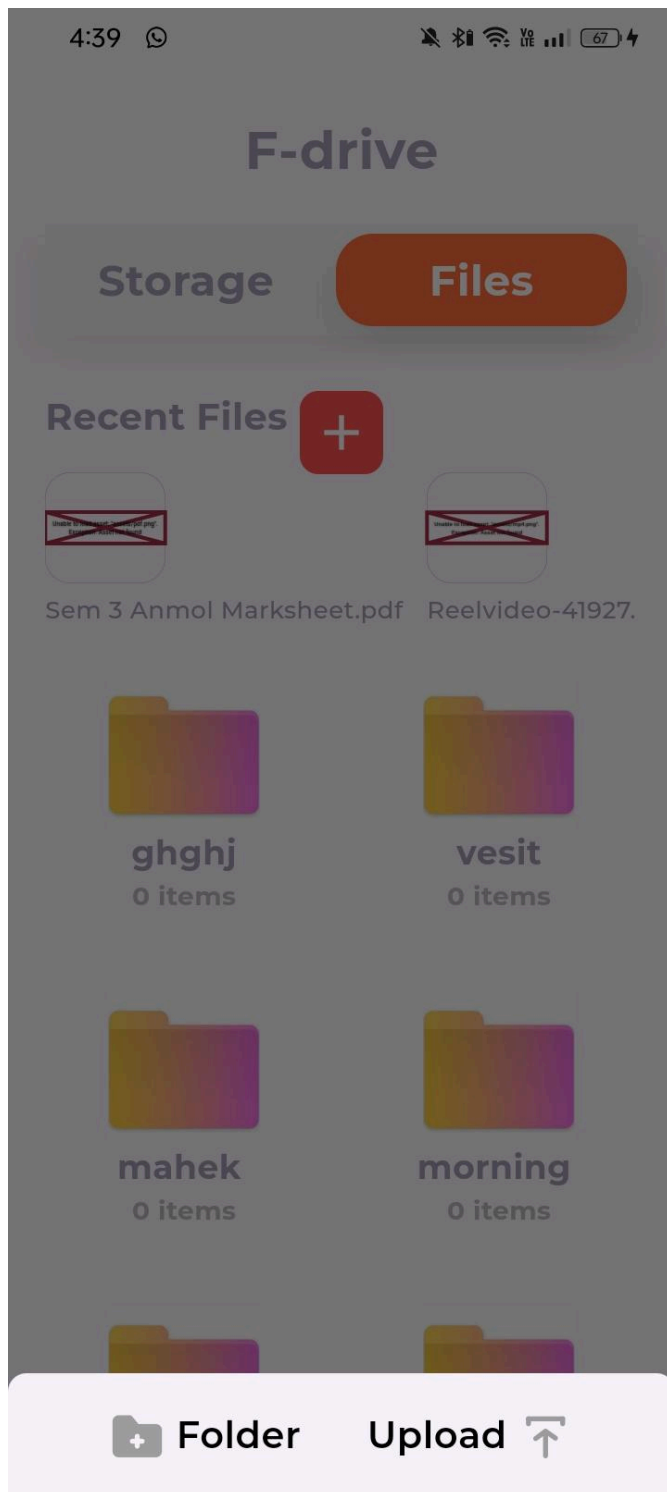
```

        offset: const Offset(10, 10),
        blurRadius: 5,
      )
    ],
  ),
  child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      Image.asset(
        'assets/folder.jpg',
        width: 82,
        height: 82,
        fit: BoxFit.cover,
      ),
      Text(
        folderController.foldersList[index].name,
        style: textStyle(18, textColor, FontWeight.bold),
      ),
      StreamBuilder<QuerySnapshot>(
        stream: FirebaseFirestore.instance
          .collection('users')
          .doc(FirebaseAuth.instance.currentUser!.uid)
          .collection('files')
          .where('folder', isEqualTo: folderController.foldersList[index].name)
          .snapshots(),
        builder: (context, snapshot) {
          if (snapshot.connectionState == ConnectionState.waiting) {
            return CircularProgressIndicator();
          }
          if (snapshot.hasError) {

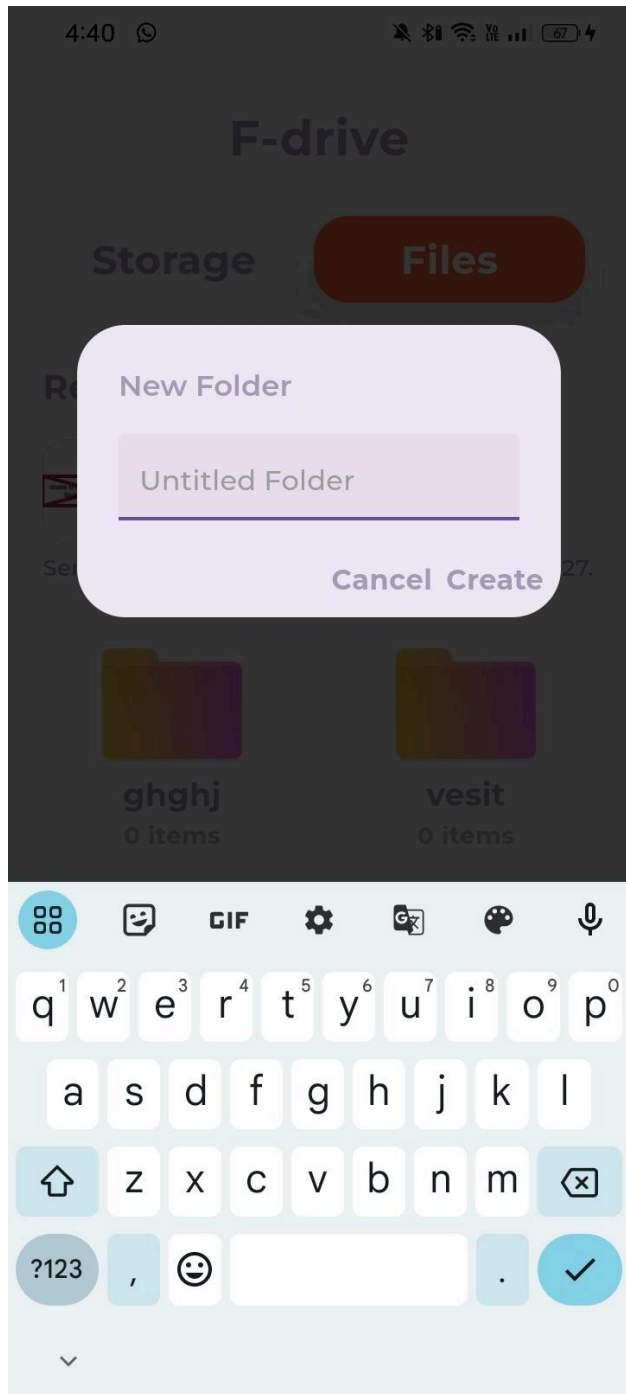
```

```
        return Text('Error: ${snapshot.error}');
    }
    return Text(
        "${snapshot.data!.docs.length} items",
        style: textStyle(14, Colors.grey[400]!, FontWeight.bold),
    );
},
)
],
),
),
);
},
);
} else {
    return Center(
        child: CircularProgressIndicator(),
    );
}
},
);
}
}
```

Output:-







**Conclusion:-** In this experiment I learnt about navigation and routing in flutter and successfully implemented it in my project.

