Name: Atharva Chavan Div: D15A Roll no: 10 Batch 'A'

MAD PWA Lab Exp 5

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

Theory:

In the realm of Flutter app development, the effective application of navigation, routing, and gestures is pivotal for creating an intuitive and engaging user experience. This theoretical exploration delves into the foundational concepts and principles guiding the implementation of navigation, routing, and gestures in Flutter applications.

Navigation and Routing

Navigation is the process of moving between different screens or sections within an app. Flutter employs a structured routing system to facilitate seamless transitions between these screens. This system is driven by the following principles:

Navigator.push()

The Navigator.push() method is employed to navigate from one screen to another by adding a new page to the navigation stack. This method is integral to the dynamic flow of a Flutter application.

Principles:

1. Screen Transition:

Invoking Navigator.push() triggers a transition to a new screen, pushing it onto the navigation stack.

This method is typically used when moving from one screen to another, such as transitioning from a home screen to a details screen.

2. Named Routes:

Navigator.pushNamed() is a variation of this method used when navigating to a screen defined by a named route.

Named routes provide a clear and maintainable way to organize and navigate between different screens in the app.

3. Arguments:

The method allows the passing of arguments to the destination screen, enabling dynamic content rendering.

Arguments facilitate the customization of the new screen based on the context of the navigation.

Navigator.pop()

Conversely, the Navigator.pop() method is employed to remove the current screen from the navigation stack, returning to the previous screen. It plays a vital role in controlling the flow of the application.

Principles:

1. Screen Removal:

When Navigator.pop() is called, the current screen is popped off the navigation stack, reverting to the previous screen.

This method is typically used in scenarios where a user completes a task on one screen and returns to the previous screen.

2. Data Passing:

Information can be passed back to the previous screen using the Navigator.pop() method.

This allows for a seamless exchange of data between screens, enhancing the overall user experience.

3. Return Values:

Developers can retrieve values returned from the popped screen, enabling dynamic updates or actions based on the user's interactions.

Code:

```
//home.dart

import 'package:flutter/material.dart';

import 'package:jio_saavn_app/screens/library.dart';

import 'package:jio_saavn_app/screens/search.dart';
```

```
import 'navigation_bar.dart'; // Import the custom navigation bar

class HomeScreen extends StatefulWidget {
  @override
  _HomeScreenState createState() => _HomeScreenState();
}

class _HomeScreenState extends State<HomeScreen> {
  int _selectedIndex = 0;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.black,
      body: SingleChildScrollView(
        child: Padding(
          padding: const EdgeInsets.all(12.0),
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              Center(
                child: Align(
                  alignment: Alignment.topLeft,
                  child: Text(
                    'Music',
                    style: TextStyle(
                      fontSize: 30,
                      fontWeight: FontWeight.bold,
                      color: Color.fromARGB(255, 255, 255, 255),
                    ),
                  ),
```

```
  ),
),
SizedBox(height: 10),
SizedBox(height: 20),
Text(
  'Recommended Songs for you:',
  style: TextStyle(
    fontSize: 20,
    fontWeight: FontWeight.bold,
    color: Color.fromARGB(255, 255, 255, 255),
  ),
),
SizedBox(height: 25),
_buildImageRow([
  'Alone',
  'Kesariya',
  'We own it',
], [
  'assets/song1.png',
  'assets/song2.png',
  'assets/song3.png',
]),
SizedBox(height: 20),
Text(
  'Top artists for you:',
  style: TextStyle(
    fontSize: 20,
    fontWeight: FontWeight.bold,
    color: Color.fromARGB(255, 255, 255, 255),
  ),
),
```

```
SizedBox(height: 25),
_buildImageRow([
  'Eminem',
  'Arjit Singh',
  'Selena Gomez',
], [
  'assets/eminem.png',
  'assets/arjit.png',
  'assets/selena.png',
]),
SizedBox(height: 20),
Text(
  'New Releases:',
  style: TextStyle(
    fontSize: 20,
    fontWeight: FontWeight.bold,
    color: Color.fromARGB(255, 255, 255, 255),
  ),
),
SizedBox(height: 25),
_buildImageRow([
  'Alone',
  'Kesariya',
  'We own it',
  'We own it',
], [
  'assets/song1.png',
  'assets/song2.png',
  'assets/song3.png',
  'assets/song3.png',
]),
```

```
        SizedBox(height: 20),
        Text(
          'Jio Saavn picks:',
          style: TextStyle(
            fontSize: 20,
            fontWeight: FontWeight.bold,
            color: Color.fromARGB(255, 255, 255, 255),
          ),
        ),
        SizedBox(height: 25),
        _buildImageRow([
          'Alone',
          'Kesariya',
          'We own it',
        ], [
          'assets/song1.png',
          'assets/song2.png',
          'assets/song3.png',
        ]),
      ],
    ),
  ),
),
bottomNavigationBar: CustomBottomNavigationBar(
  selectedIndex: _selectedIndex,
  onItemTapped: (index) {
    setState(() {
      _selectedIndex = index;
    });
    if (_selectedIndex == 1) {
      // Navigate to SearchScreen when the Search tab is tapped
```

```
          Navigator.push(
            context,
            MaterialPageRoute(builder: (context) => SearchScreen()),
          );
        } else if (_selectedIndex == 2) {
          // Navigate to MyLibrary when the Library tab is tapped
          Navigator.push(
            context,
            MaterialPageRoute(builder: (context) => MyLibrary()),
          );
        }
      },
    ),
  );
}

Color _getColor(int index) {
  return _selectedIndex == index
      ? Colors.white
      : const Color.fromARGB(255, 128, 128, 128);
}

Widget _buildSuggestedItem(
    BuildContext context, String name, String imagePath) {
  return GestureDetector(
    onTap: () {
      // Navigate to SongDetailsScreen when the image is tapped
      //  Navigator.push(
      //    context,
      //    MaterialPageRoute(
      //      builder: (context) => SongDetailsScreen(songTitle: name, songImagePath: imagePath),
```

```
      //    ),
      //  );
    },
    child: Container(
      width: 120,
      margin: EdgeInsets.only(right: 5),
      child: Column(
        children: [
          Image.asset(
            imagePath,
            height: 100,
            width: 100,
            fit: BoxFit.cover,
          ),
          SizedBox(height: 5),
          Text(
            name,
            style: TextStyle(color: Colors.white),
          ),
        ],
      ),
    ),
  );
}

 Widget _buildImageRow(List<String> names, List<String> imagePaths) {
   return SizedBox(
     height: 150,
     child: ListView.builder(
       scrollDirection: Axis.horizontal,
       itemCount: names.length,
```

```
      itemBuilder: (BuildContext context, int index) {
        return _buildSuggestedItem(
          context,
          names[index],
          imagePaths[index],
        );
      },
    ),
  );
 }
}

//search.dart
import 'package:flutter/material.dart';
import 'package:jio_saavn_app/screens/library.dart';
import 'home.dart';
import 'navigation_bar.dart'; // Import your custom navigation bar

class SearchScreen extends StatefulWidget {
 @override
 _SearchScreenState createState() => _SearchScreenState();
}

class _SearchScreenState extends State<SearchScreen> {
 int _selectedIndex = 1; // Set the default index to Search

 @override
 Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: Colors.black, // Set background color here
```

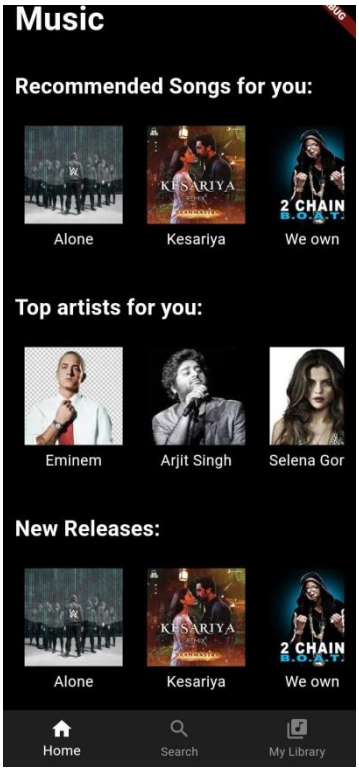
```
body: SingleChildScrollView(
 child: Padding(
  padding: const EdgeInsets.all(16.0),
  child: Column(
   crossAxisAlignment: CrossAxisAlignment.start,
   children: [
    // Add your search bar here
    _buildSearchBar(),

    // Add the top trending songs title
    Text(
     'Trending',
     style: TextStyle(
      fontSize: 24,
      fontWeight: FontWeight.bold,
      color: Color.fromARGB(255, 255, 255, 255),
     ),
    ),

    // Add the three images with their song names
    _buildSuggestedItem('Song 4', 'assets/song4.png'),
    _buildSuggestedItem('Song 5', 'assets/song5.png'),
    _buildSuggestedItem('Song 6', 'assets/song6.png'),
   ],
  ),
 ),
),
bottomNavigationBar: CustomBottomNavigationBar(
 selectedIndex: _selectedIndex,
 onItemTapped: (index) {
  setState(() {
```

```
        _selectedIndex = index;
       });
       if (_selectedIndex == 0) {
        // Navigate to HomeScreen when the Home tab is tapped
        Navigator.push(
         context,
         MaterialPageRoute(builder: (context) => HomeScreen()),
        );
       } else if (_selectedIndex == 2) {
        // Navigate to MyLibrary when the Library tab is tapped
        Navigator.push(
         context,
         MaterialPageRoute(builder: (context) => MyLibrary()),
        );
       }
      },
      currentIndex: _selectedIndex,
    ),
  );
}

Color _getColor(int index) {
  return _selectedIndex == index
      ? Colors.white
      : const Color.fromARGB(255, 128, 128, 128);
}

Widget _buildSearchBar() {
  return Container(
    margin: EdgeInsets.symmetric(vertical: 16),
    padding: EdgeInsets.symmetric(horizontal: 16),
```

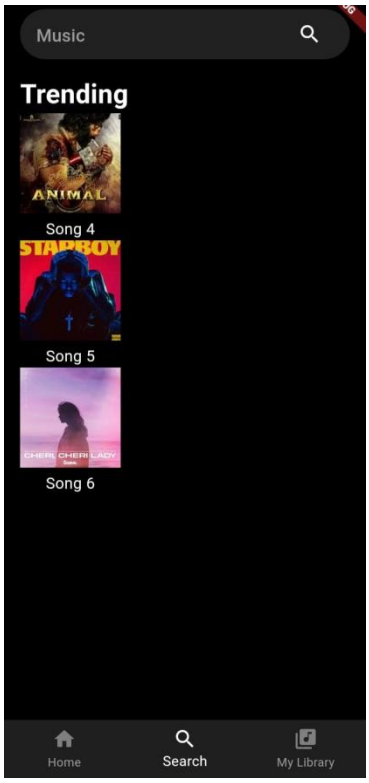
```
    decoration: BoxDecoration(
      color: Colors.grey[900],
      borderRadius: BorderRadius.circular(30),
    ),
    child: TextField(
      style: TextStyle(color: Colors.white),
      decoration: InputDecoration(
        hintText: 'Music',
        hintStyle: TextStyle(color: Colors.grey),
        suffixIcon: Icon(Icons.search, color: Colors.white),
        border: InputBorder.none,
      ),
    ),
  );
}

Widget _buildSuggestedItem(String name, String imagePath) {
  return Column(
    children: [
      Image.asset(
        imagePath,
        height: 100, // Set the desired height
        width: 100, // Set the desired width
        fit: BoxFit.cover, // Adjust the image size
      ),
      SizedBox(height: 5),
      Text(
        name,
        style: TextStyle(color: Colors.white),
      ),
    ],
```

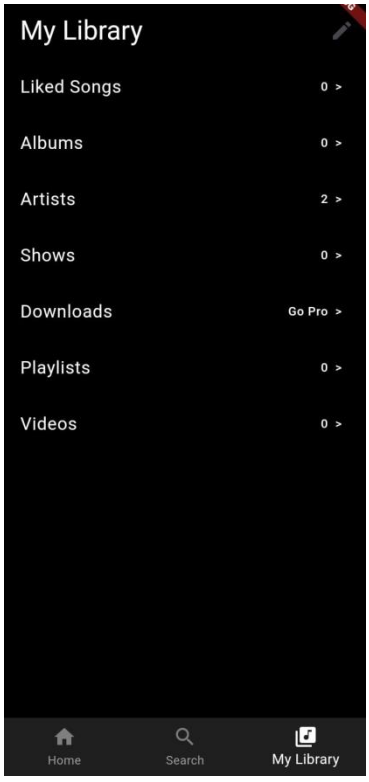
```
    );
  }
}
```

Output:


Music
Recommended Songs for you:
Alone
Kesariya
We own
Top artists for you:
Eminem
Arjit Singh
Selena Gor
New Releases:
Alone
Kesariya
We own
Home
Search
My Library

Name: Atharva Chavan Div: D15A Roll no: 10 Batch 'A'





Conclusion:

In conclusion, Navigator.push() and Navigator.pop() are foundational methods in Flutter navigation, facilitating the seamless movement between screens. Navigator.push() initiates

the addition of a new screen to the stack, allowing for dynamic navigation, while Navigator.pop() removes the current screen, providing a mechanism for controlled navigation and data exchange between screens.