

MPL EXPERIMENT : 5
D15A-30
Jai navani
(A page with navigation and routing)

Home_page UI:

Theory:

The HomePage is a stateful widget (StatefulWidget) that manages the bottom navigation and switches between different pages like Home, Chat, Sell, My Ads, and Account.

Key Components:

- `_currentIndex`: Tracks which tab is currently selected.
- `_onItemTapped(int index)`: Updates `_currentIndex` to switch tabs.
- `_pages` List: Contains all five pages, and the selected page is displayed using `_currentIndex`.
- `BottomNavigationBar`: Provides a tab bar at the bottom for easy navigation.

Widgets & Properties (Home Page)

- `StatefulWidget (HomePage)`: Since navigation changes dynamically, `HomePage` is a `StatefulWidget`. `_HomePageState` manages the selected tab and updates the UI when a tab is clicked.
- `Scaffold`: Provides the main structure of the app, including `AppBar`, `body`, and `BottomNavigationBar`.
- `AppBar`: `title: Text([...]_currentIndex)` → Displays the current tab title dynamically. `backgroundColor: Colors.teal` → Matches the app theme.
- `BottomNavigationBar`: Allows navigation between Home, Chat, Sell, My Ads, and Account. `currentIndex` tracks the selected tab.

onTap: _onItemTapped updates _currentIndex, reloading the UI.

selectedItemColor: Colors.teal highlights the active tab.

- HomeContentPage (Main Home Screen)

Includes Search Bar, Categories Grid, and Recommendations List.

- Search Bar (TextField):

hintText: "Search 'Mobiles'" → Placeholder text for searching.

prefixIcon: Icon(Icons.search, color: Colors.teal) → Search icon inside the field.

border: OutlineInputBorder(borderRadius: BorderRadius.circular(30.0)) → Stylish rounded edges.

- Category Grid (GridView.count)

Displays categories like Cars, Properties, Mobiles, etc. in a 4-column grid.

GestureDetector detects taps and updates selectedCategory.

Uses CategoryItem to show icons with labels.

- CategoryItem Widget

CircleAvatar(radius: 24.0, backgroundColor: Colors.teal[100]) → Creates a circular button with an icon.

Icon(icon, color: Colors.teal, size: 30) → Ensures icon visibility.

Text(label, style: TextStyle(fontSize: 12.0, color: Colors.teal)) → Displays category name below the icon.

- Recommendations List (ListView.builder)

Dynamically displays filtered product recommendations based on selectedCategory.

ListView.builder creates cards for each product.

Uses RecommendationItem to structure product details.

- RecommendationItem Widget

Displays product image, title, price, and location inside a Card.

leading: Image.asset(imageUrl, width: 50, height: 50, fit: BoxFit.cover) → Loads the product image.

- title: Text(title), subtitle: Text('\$price\n\$location') → Shows product details.
- trailing: Icon(Icons.favorite_border, color: Colors.teal) → Favorite button for liking items.

Code:

```
import 'package:flutter/material.dart';
import 'chat_page.dart';
import 'sell_page.dart';
import 'my_ads_page.dart';
import 'account_page.dart';

class HomePage extends StatefulWidget {
  @override
  _HomePageState createState() => _HomePageState();
}

class _HomePageState extends State<HomePage> {
  int _currentIndex = 0;

  final List<Widget> _pages = [
    HomeContentPage(), // Main Home Page
    ChatPage(),        // Chat Page
    SellPage(),         // Sell Page
    MyAdsPage(),        // My Ads Page
    AccountPage(),      // Account Page
  ];

  void _onItemTapped(int index) {
    setState(() {
      _currentIndex = index;
    });
  }
}
```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text([
        'Home', 'Chat', 'Sell', 'My Ads', 'Account'
      ][_currentIndex]),
      backgroundColor: Colors.teal,
    ),
    body: _pages[_currentIndex],
    bottomNavigationBar: BottomNavigationBar(
      currentIndex: _currentIndex,
      onTap: _onItemTapped,
      selectedItemColor: Colors.teal,
      unselectedItemColor: Colors.grey,
      items: [
        BottomNavigationBarItem(icon: Icon(Icons.home), label: 'Home'),
        BottomNavigationBarItem(icon: Icon(Icons.chat), label: 'Chat'),
        BottomNavigationBarItem(icon: Icon(Icons.add_circle, size: 36.0), label: 'Sell'),
        BottomNavigationBarItem(icon: Icon(Icons.post_add), label: 'My Ads'),
        BottomNavigationBarItem(icon: Icon(Icons.account_circle), label: 'Account'),
      ],
    ),
  );
}

```

```

class HomeContentPage extends StatefulWidget {
  @override
  _HomeContentPageState createState() => _HomeContentPageState();
}

```

```

class _HomeContentPageState extends State<HomeContentPage> {
  String selectedCategory = "All";
  final Map<String, List<String>> categoryItems = {
    "All": ["Car 1", "Car 2", "Car 3", "Car 4", "Car 5"],
    "Cars": ["Car 1", "Car 2", "Car 3"],
    "Properties": ["House 1", "House 2"],
    "Mobiles": ["Phone 1", "Phone 2"],
    "Jobs": ["Job 1", "Job 2"],
    "Bikes": ["Bike 1", "Bike 2"],
    "Electronics": ["TV 1", "Laptop 2"],
    "Commercial": ["Truck 1"],
  };
}

```

```
"Furniture": ["Sofa 1"],  
};
```

```
final Map<String, IconData> categoryIcons = {  
  "Cars": Icons.directions_car,  
  "Properties": Icons.home,  
  "Mobiles": Icons.smartphone,  
  "Jobs": Icons.work,  
  "Bikes": Icons.motorcycle,  
  "Electronics": Icons.tv,  
  "Commercial": Icons.local_shipping,  
  "Furniture": Icons.chair,  
};
```

```
@override  
Widget build(BuildContext context) {  
  return SingleChildScrollView(  
    child: Padding(  
      padding: const EdgeInsets.all(8.0),  
      child: Column(  
        crossAxisAlignment: CrossAxisAlignment.start,  
        children: [  
          // Search Bar  
          Container(  
            margin: EdgeInsets.symmetric(vertical: 8.0),  
            child: TextField(  
              decoration: InputDecoration(  
                hintText: 'Search "Mobiles"',  
                prefixIcon: Icon(Icons.search, color: Colors.teal),  
                filled: true,  
                fillColor: Colors.grey[200],  
                border: OutlineInputBorder(  
                  borderRadius: BorderRadius.circular(30.0),  
                  borderSide: BorderSide.none,  
                ),  
              ),  
            ),  
          ),  
          SizedBox(height: 16.0),  
          // Categories  
          GridView.count(  
            crossAxisCount: 4,  
            shrinkWrap: true,  
            physics: NeverScrollableScrollPhysics(),
```

```

children: categoryItems.keys.map((category) {
  return GestureDetector(
    onTap: () {
      setState(() {
        selectedCategory = category;
      });
    },
    child: CategoryItem(
      icon: categoryIcons[category] ?? Icons.category, // Default icon if not found
      label: category,
    ),
  );
}).toList(),
),
SizedBox(height: 16.0),
// Fresh Recommendations Header
Text(
  'Fresh Recommendations',
  style: TextStyle(fontSize: 18.0, fontWeight: FontWeight.bold, color: Colors.teal),
),
SizedBox(height: 8.0),
// Recommendations List
ListView.builder(
  itemCount: categoryItems[selectedCategory]!.length,
  shrinkWrap: true,
  physics: NeverScrollableScrollPhysics(),
  itemBuilder: (context, index) {
    return RecommendationItem(
      imageUrl: 'assets/images/car.jpg', // Local image
      title: categoryItems[selectedCategory]![index],
      price: '₹${(index + 1) * 10000}',
      location: 'Location ${index + 1}',
    );
  },
),
],
),
),
);
}
}

```

```

// Category Item Widget
class CategoryItem extends StatelessWidget {

```

```
final IconData icon;  
final String label;
```

```
CategoryItem({required this.icon, required this.label});
```

```
@override  
Widget build(BuildContext context) {  
  return Column(  
    mainAxisAlignment: MainAxisAlignment.min,  
    children: [  
      CircleAvatar(  
        radius: 24.0,  
        backgroundColor: Colors.teal[100],  
        child: Icon(icon, color: Colors.teal, size: 30), // Ensuring icon visibility  
      ),  
      SizedBox(height: 4.0),  
      Text(label, style: TextStyle(fontSize: 12.0, color: Colors.teal)),  
    ],  
  );  
}
```

```
// Recommendation Item Widget  
class RecommendationItem extends StatelessWidget {  
  final String imageUrl;  
  final String title;  
  final String price;  
  final String location;
```

```
  RecommendationItem({  
    required this.imageUrl,  
    required this.title,  
    required this.price,  
    required this.location,  
  });
```

```
@override  
Widget build(BuildContext context) {  
  return Card(  
    margin: EdgeInsets.symmetric(vertical: 8.0),  
    child: ListTile(  
      leading: Image.asset(imageUrl, width: 50, height: 50, fit: BoxFit.cover),  
      title: Text(title),  
      subtitle: Text('$price\n$location'),
```

```

        isThreeLine: true,
        trailing: Icon(Icons.favorite_border, color: Colors.teal),
      ),
    );
  }
}

```

ROUTED PAGE:

```

import 'package:flutter/material.dart';

class SellPage extends StatefulWidget {
  @override
  _SellPageState createState() => _SellPageState();
}

class _SellPageState extends State<SellPage> {
  final _formKey = GlobalKey<FormState>();
  final _nameController = TextEditingController();
  final _priceController = TextEditingController();

  String? _selectedCategory;

  final List<String> _categories = [
    'Cars',
    'Properties',
    'Mobiles',
    'Jobs',
    'Bikes',
    'Electronics',
    'Commercial',
    'Furniture'
  ];

  void _submitForm() {
    if (_formKey.currentState!.validate() && _selectedCategory != null) {
      // Handle form submission logic
      showDialog(
        context: context,
        builder: (context) {
          return AlertDialog(
            title: Text('Product Listed'),
            content: Text(

```



```

        'Name: ${_nameController.text}\nCategory: $_selectedCategory\nPrice:
$_priceController.text}',
      ),
      actions: [
        TextButton(
          onPressed: () {
            Navigator.of(context).pop();
          },
          child: Text('Close'),
        ),
      ],
    );
  },
);
}
}

```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('Sell Product'),
      backgroundColor: Colors.teal,
    ),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Form(
        key: _formKey,
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            // Name Input
            TextFormField(
              controller: _nameController,
              decoration: InputDecoration(
                labelText: 'Product Name',
                border: OutlineInputBorder(
                  borderRadius: BorderRadius.circular(10.0),
                ),
              ),
            ),
            validator: (value) {
              if (value == null || value.isEmpty) {
                return 'Please enter the product name';
              }
            }
          ],
        ),
      ),
    ),
  );
}

```

```

        return null;
      },
    ),
    SizedBox(height: 16.0),

    // Category Dropdown
    DropdownButtonFormField<String>(
      decoration: InputDecoration(
        labelText: 'Category',
        border: OutlineInputBorder(
          borderRadius: BorderRadius.circular(10.0),
        ),
      ),
    ),
    value: _selectedCategory,
    items: _categories.map((category) {
      return DropdownMenuItem(
        value: category,
        child: Text(category),
      );
    }).toList(),
    onChanged: (value) {
      setState(() {
        _selectedCategory = value;
      });
    },
    validator: (value) {
      if (value == null) {
        return 'Please select a category';
      }
      return null;
    },
  ),
  SizedBox(height: 16.0),

  // Price Input
  TextFormField(
    controller: _priceController,
    keyboardType: TextInputType.number,
    decoration: InputDecoration(
      labelText: 'Price',
      border: OutlineInputBorder(
        borderRadius: BorderRadius.circular(10.0),
      ),
    ),
  ),

```

```

        validator: (value) {
          if (value == null || value.isEmpty) {
            return 'Please enter the price';
          }
          if (double.tryParse(value) == null) {
            return 'Please enter a valid number';
          }
          return null;
        },
      ),
      SizedBox(height: 16.0),

      // Submit Button
      Center(
        child: ElevatedButton(
          onPressed: _submitForm,
          style: ElevatedButton.styleFrom(
            //primary: Colors.teal,
            shape: RoundedRectangleBorder(
              borderRadius: BorderRadius.circular(10.0),
            ),
          ),
        ),
        child: Text('List Product'),
      ),
    ),
  ],
),
),
),
);
}
}

```

OUTPUT:

🔍 Search "Mobiles"



All



Cars



Properties



Mobiles



Jobs



Bikes



Electronics



Commercial



Furniture

Fresh Recommendations



Car 1

₹10000

Location 1



Car 2

₹20000

Location 2



Car 3

₹30000

Location 3



Home



5:18

Sell

Sell Product

Product Name

phone

Category

Mobiles

Price

1500

List Product

+

Sell

500

Product Listed

Name: phone

Category: Mobiles

Price: 1500

Close