# AIM:

# Learning Resources

# State Management in Flutter

# Practical Task:

# Build a shopping cart app with product listings using ListView and GridView.

# Use setState to manage cart items dynamically.

# Topics Covered:

# Stateful Widgets and State Management

# Navigation and Routing

# User Interaction with Lists

# References: <https://www.youtube.com/watch?v=XCmJagNPbi4> <https://pub.dev/packages/flutter_cart> <https://blog.logrocket.com/building-shopping-cart-flutter/>

# THEORY:

**State Management in Flutter**

State management is a crucial aspect of Flutter development, enabling dynamic user interfaces that respond to changes. In Flutter, the state is divided into:

1. **Ephemeral State (UI State):** Managed within a single widget using setState().
2. **App State (Global State):** Shared across multiple widgets using advanced solutions like Provider, Riverpod, or Bloc.

For this task, setState() is used to manage cart items dynamically. It allows immediate UI updates when users add or remove items from the cart.

**Widgets Used in Shopping Cart App**

1. **ListView & GridView:** Used to display product listings in a structured manner.
2. **Stateful Widgets:** Essential for managing dynamic data such as cart items.
3. **Navigation & Routing:** Helps users navigate between product listing and cart details.
4. **User Interaction:** Implementing buttons to add/remove items from the cart.

# CODE:

# main.dart

import 'package:flutter/material.dart';

import 'package:provider/provider.dart';

import 'providers/cart\_provider.dart';

import 'home\_page.dart';

import 'cart\_page.dart';

void main() {

  runApp(const MyApp());

}

class MyApp extends StatelessWidget {

  const MyApp({super.key});

  @override

  Widget build(BuildContext context) {

    return ChangeNotifierProvider(

      create: (\_) => CartProvider(),

      child: Consumer<CartProvider>(

        builder: (context, cart, child) {

          return MaterialApp(

            title: 'ShopCart',

            debugShowCheckedModeBanner: false,

            home: const BottomNavBar(),

          );

        },

      ),

    );

  }

}

class BottomNavBar extends StatefulWidget {

  const BottomNavBar({super.key});

  @override

  \_BottomNavBarState createState() => \_BottomNavBarState();

}

class \_BottomNavBarState extends State<BottomNavBar> {

  int \_selectedIndex = 0;

  static final List<Widget> \_pages = [const HomePage(), const CartPage()];

  void \_onItemTapped(int index) {

    setState(() {

      \_selectedIndex = index;

    });

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      body: \_pages[\_selectedIndex],

      bottomNavigationBar: BottomNavigationBar(

        currentIndex: \_selectedIndex,

        onTap: \_onItemTapped,

        items: const [

          BottomNavigationBarItem(icon: Icon(Icons.home), label: 'Home'),

          BottomNavigationBarItem(

              icon: Icon(Icons.shopping\_cart), label: 'Cart'),

        ],

      ),

    );

  }

}

# home\_page.dart

import 'dart:convert';

import 'package:flutter/material.dart';

import 'package:http/http.dart' as http;

import 'product\_detail.dart';

import 'models/product.dart';

class HomePage extends StatefulWidget {

  const HomePage({super.key});

  @override

  \_HomePageState createState() => \_HomePageState();

}

class \_HomePageState extends State<HomePage> {

  List<Product> \_products = [];

  List<Product> \_filteredProducts = [];

  bool \_isLoading = true;

  @override

  void initState() {

    super.initState();

    fetchProducts();

  }

  Future<void> fetchProducts() async {

    final response =

        await http.get(Uri.parse('https://fakestoreapi.com/products'));

    if (response.statusCode == 200) {

      List<dynamic> data = json.decode(response.body);

      setState(() {

        \_products = data.map((item) => Product.fromJson(item)).toList();

        \_filteredProducts = \_products;

        \_isLoading = false;

      });

    }

  }

  void \_filterProducts(String query) {

    setState(() {

      \_filteredProducts = \_products

          .where((p) => p.title.toLowerCase().contains(query.toLowerCase()))

          .toList();

    });

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      backgroundColor: Colors.white, // Light Background

      appBar: AppBar(

        title: const Text(

          "ShopCart",

          style: TextStyle(

            color: Colors.white,

            fontSize: 24,

            fontWeight: FontWeight.bold,

            letterSpacing: 2,

          ),

        ),

        backgroundColor: const Color.fromARGB(255, 103, 9, 253),

        centerTitle: true,

      ),

      body: Column(

        children: [

          // Search Box

          Padding(

            padding: const EdgeInsets.all(12.0),

            child: TextField(

              onChanged: \_filterProducts,

              style: const TextStyle(color: Colors.black87),

              decoration: InputDecoration(

                filled: true,

                fillColor: Colors.white,

                prefixIcon: const Icon(Icons.search, color: Colors.purple),

                labelText: "Search Products",

                labelStyle: const TextStyle(color: Colors.black54),

                border: OutlineInputBorder(

                  borderRadius: BorderRadius.circular(15),

                ),

                enabledBorder: OutlineInputBorder(

                  borderSide: const BorderSide(color: Colors.purple),

                  borderRadius: BorderRadius.circular(15),

                ),

                focusedBorder: OutlineInputBorder(

                  borderSide:

                      const BorderSide(color: Colors.deepPurple, width: 2),

                  borderRadius: BorderRadius.circular(15),

                ),

              ),

            ),

          ),

          // Product List with Animation

          Expanded(

            child: \_isLoading

                ? const Center(

                    child: CircularProgressIndicator(

                      color: Colors.purple,

                    ),

                  )

                : ListView.builder(

                    itemCount: \_filteredProducts.length,

                    itemBuilder: (context, index) {

                      final product = \_filteredProducts[index];

                      return AnimatedContainer(

                        duration: const Duration(milliseconds: 500),

                        curve: Curves.easeInOut,

                        margin: const EdgeInsets.symmetric(

                            horizontal: 10, vertical: 5),

                        decoration: BoxDecoration(

                          color: Colors.white,

                          borderRadius: BorderRadius.circular(12),

                          boxShadow: [

                            BoxShadow(

                              color: Colors.purple.withOpacity(0.2),

                              blurRadius: 5,

                              spreadRadius: 1,

                            ),

                          ],

                        ),

                        child: ListTile(

                          leading: Hero(

                            tag: "product\_${product.id}",

                            child: Image.network(

                              product.image,

                              width: 60,

                              height: 60,

                              fit: BoxFit.cover,

                            ),

                          ),

                          title: Text(

                            product.title,

                            style: const TextStyle(

                                fontWeight: FontWeight.bold, fontSize: 16),

                          ),

                          subtitle: Text(

                            "\$${product.price}",

                            style: const TextStyle(

                                color: Colors.deepPurple,

                                fontSize: 14,

                                fontWeight: FontWeight.bold),

                          ),

                          trailing: const Icon(Icons.arrow\_forward\_ios,

                              color: Colors.purple),

                          onTap: () => Navigator.push(

                            context,

                            MaterialPageRoute(

                              builder: (\_) =>

                                  ProductDetailPage(product: product),

                            ),

                          ),

                        ),

                      );

                    },

                  ),

          ),

        ],

      ),

    );

  }

}

# product\_detail.dart

import 'package:flutter/material.dart';

import 'package:provider/provider.dart';

import 'models/product.dart';

import 'providers/cart\_provider.dart';

class ProductDetailPage extends StatelessWidget {

  final Product product;

  const ProductDetailPage({super.key, required this.product});

  @override

  Widget build(BuildContext context) {

    final cart = Provider.of<CartProvider>(context);

    return Scaffold(

      backgroundColor: const Color.fromARGB(255, 230, 243, 255),

      appBar: AppBar(

        title: Text(

          product.title,

          style: const TextStyle(fontSize: 18, fontWeight: FontWeight.bold),

        ),

        backgroundColor: const Color.fromARGB(255, 0, 174, 255),

      ),

      body: Padding(

        padding: const EdgeInsets.all(16.0),

        child: Column(

          crossAxisAlignment: CrossAxisAlignment.center,

          children: [

            Hero(

              tag: product.id, // Adds animation between pages

              child: ClipRRect(

                borderRadius: BorderRadius.circular(12),

                child: Image.network(product.image, height: 250),

              ),

            ),

            const SizedBox(height: 16),

            Text(

              product.title,

              textAlign: TextAlign.center,

              style: const TextStyle(

                  fontSize: 22,

                  fontWeight: FontWeight.bold,

                  color: Colors.black),

            ),

            const SizedBox(height: 8),

            Text(

              "\$${product.price}",

              style: const TextStyle(

                  fontSize: 20,

                  fontWeight: FontWeight.bold,

                  color: Colors.green),

            ),

            const SizedBox(height: 12),

            Text(

              product.description,

              textAlign: TextAlign.center,

              style: const TextStyle(fontSize: 16, color: Colors.black87),

            ),

            const SizedBox(height: 20),

            ElevatedButton.icon(

              onPressed: () {

                cart.addToCart(product);

                ScaffoldMessenger.of(context).showSnackBar(

                  SnackBar(

                    content: const Text("Item added to cart!"),

                    behavior: SnackBarBehavior.floating,

                    duration: const Duration(seconds: 2),

                    shape: RoundedRectangleBorder(

                      borderRadius: BorderRadius.circular(8),

                    ),

                    backgroundColor: Colors.green,

                  ),

                );

              },

              icon: const Icon(Icons.shopping\_cart),

              label: const Text("Add to Cart"),

              style: ElevatedButton.styleFrom(

                backgroundColor: Colors.blue,

                padding:

                    const EdgeInsets.symmetric(horizontal: 20, vertical: 12),

                textStyle:

                    const TextStyle(fontSize: 18, fontWeight: FontWeight.bold),

              ),

            ),

          ],

        ),

      ),

    );

  }

}

# cart\_page.dart

import 'package:flutter/material.dart';

import 'providers/cart\_provider.dart';

import 'package:provider/provider.dart';

class CartPage extends StatelessWidget {

  const CartPage({super.key});

  @override

  Widget build(BuildContext context) {

    final cart = Provider.of<CartProvider>(context);

    return Scaffold(

      backgroundColor: Colors.white,

      appBar: AppBar(

        title: const Text(

          "My Cart",

          style: TextStyle(

            color: Colors.white,

            fontSize: 22,

            fontWeight: FontWeight.bold,

          ),

        ),

        backgroundColor: Colors.purple,

        centerTitle: true,

      ),

      body: cart.items.isEmpty

          ? const Center(

              child: Text(

                "Your cart is empty!",

                style: TextStyle(fontSize: 18, fontWeight: FontWeight.w600),

              ),

            )

          : Column(

              children: [

                Expanded(

                  child: ListView.builder(

                    itemCount: cart.items.length,

                    itemBuilder: (context, index) {

                      final product = cart.items[index];

                      return Card(

                        margin: const EdgeInsets.symmetric(

                            horizontal: 10, vertical: 5),

                        shape: RoundedRectangleBorder(

                          borderRadius: BorderRadius.circular(12),

                        ),

                        elevation: 4,

                        child: ListTile(

                          leading: Hero(

                            tag: "cart\_${product.id}",

                            child: Image.network(

                              product.image,

                              width: 60,

                              height: 60,

                              fit: BoxFit.cover,

                            ),

                          ),

                          title: Text(

                            product.title,

                            style: const TextStyle(

                                fontWeight: FontWeight.bold, fontSize: 16),

                          ),

                          subtitle: Text(

                            "\$${product.price}",

                            style: const TextStyle(

                                color: Colors.deepPurple,

                                fontSize: 14,

                                fontWeight: FontWeight.bold),

                          ),

                          trailing: Row(

                            mainAxisSize: MainAxisSize.min,

                            children: [

                              IconButton(

                                icon: const Icon(Icons.remove\_circle,

                                    color: Colors.red),

                                onPressed: () => cart.decreaseQuantity(product),

                              ),

                              Text(

                                "${cart.getQuantity(product)}",

                                style: const TextStyle(

                                    fontSize: 16, fontWeight: FontWeight.bold),

                              ),

                              IconButton(

                                icon: const Icon(Icons.add\_circle,

                                    color: Colors.green),

                                onPressed: () => cart.addToCart(product),

                              ),

                            ],

                          ),

                        ),

                      );

                    },

                  ),

                ),

                // Total Price with Animated Update

                Container(

                  padding:

                      const EdgeInsets.symmetric(horizontal: 15, vertical: 10),

                  color: Colors.grey.shade100,

                  child: Row(

                    mainAxisAlignment: MainAxisAlignment.spaceBetween,

                    children: [

                      const Text(

                        "Total:",

                        style: TextStyle(

                          fontSize: 18,

                          fontWeight: FontWeight.bold,

                        ),

                      ),

                      AnimatedSwitcher(

                        duration: const Duration(milliseconds: 500),

                        transitionBuilder:

                            (Widget child, Animation<double> animation) {

                          return ScaleTransition(

                              scale: animation, child: child);

                        },

                        child: Text(

                          "\$${cart.totalPrice.toStringAsFixed(2)}",

                          key: ValueKey<double>(cart.totalPrice),

                          style: const TextStyle(

                            fontSize: 18,

                            fontWeight: FontWeight.bold,

                            color: Colors.deepPurple,

                          ),

                        ),

                      ),

                    ],

                  ),

                ),

                // "Buy Now" Button with Stylish Design

                Padding(

                  padding: const EdgeInsets.all(12.0),

                  child: ElevatedButton(

                    onPressed: cart.items.isEmpty ? null : () {},

                    style: ElevatedButton.styleFrom(

                      backgroundColor: Colors.deepPurple,

                      padding: const EdgeInsets.symmetric(

                          horizontal: 40, vertical: 12),

                      shape: RoundedRectangleBorder(

                        borderRadius: BorderRadius.circular(10),

                      ),

                      elevation: 5,

                    ),

                    child: const Text(

                      "Buy Now",

                      style: TextStyle(fontSize: 18, color: Colors.white),

                    ),

                  ),

                ),

              ],

            ),

    );

  }

}

# models/product.dart

class Product {

  final int id;

  final String title;

  final double price;

  final String category;

  final String description;

  final String image;

  Product({

    required this.id,

    required this.title,

    required this.price,

    required this.category,

    required this.description,

    required this.image,

  });

  factory Product.fromJson(Map<String, dynamic> json) {

    return Product(

      id: json['id'],

      title: json['title'],

      price: (json['price'] as num).toDouble(),

      category: json['category'],

      description: json['description'],

      image: json['image'],

    );

  }

}

# providers/cart\_provider.dart

import 'package:flutter/material.dart';

import '../models/product.dart';

class CartProvider with ChangeNotifier {

  final List<Product> \_items = [];

  final Map<int, int> \_quantities = {}; // Track product quantities

  List<Product> get items => \_items;

  double get totalPrice {

    return \_items.fold(0,

        (total, product) => total + (product.price \* \_quantities[product.id]!));

  }

  void addToCart(Product product) {

    if (\_quantities.containsKey(product.id)) {

      \_quantities[product.id] = \_quantities[product.id]! + 1;

    } else {

      \_items.add(product);

      \_quantities[product.id] = 1;

    }

    notifyListeners();

  }

  void decreaseQuantity(Product product) {

    if (\_quantities.containsKey(product.id) && \_quantities[product.id]! > 1) {

      \_quantities[product.id] = \_quantities[product.id]! - 1;

    } else {

      \_items.remove(product);

      \_quantities.remove(product.id);

    }

    notifyListeners();

  }

  int getQuantity(Product product) {

    return \_quantities[product.id] ?? 0;

  }

  void removeFromCart(Product product) {

    \_items.remove(product);

    \_quantities.remove(product.id);

    notifyListeners();

  }

}

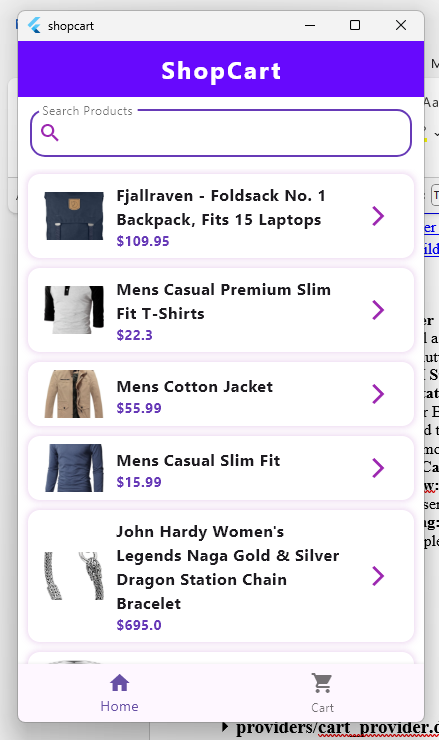
# theme.dart

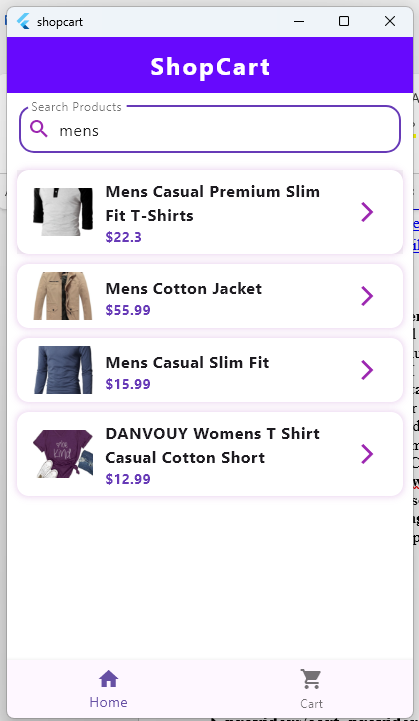
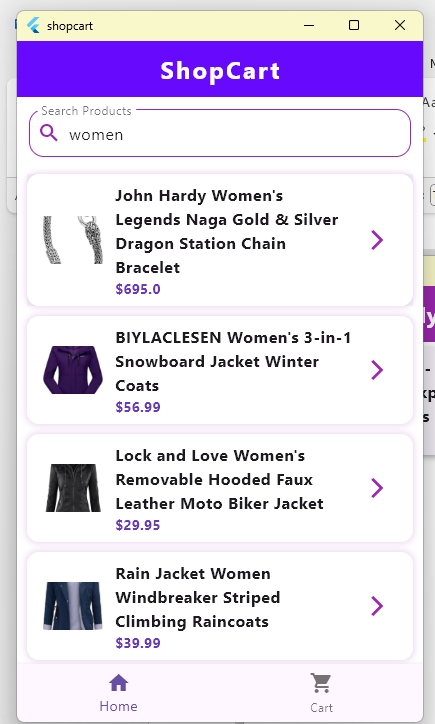
import 'package:flutter/material.dart';

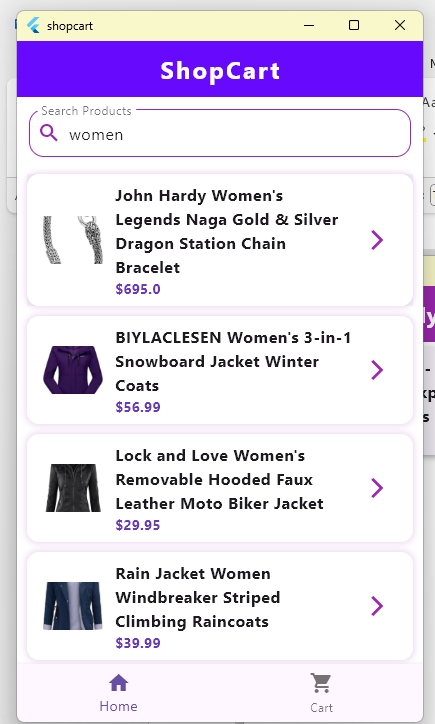
final darkTheme = ThemeData.dark().copyWith(primaryColor: Colors.blue);

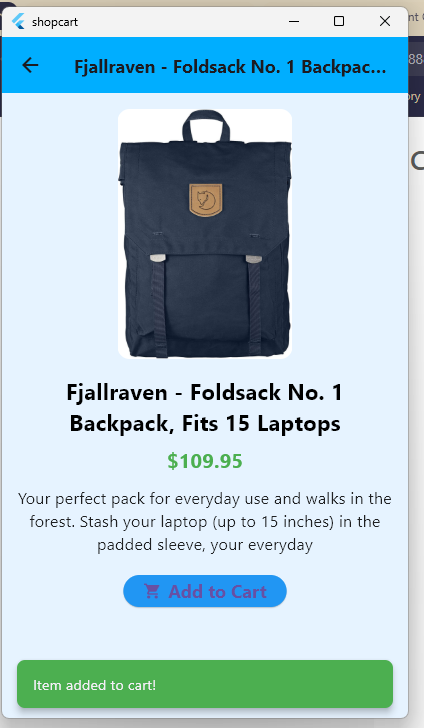
final lightTheme = ThemeData.light();

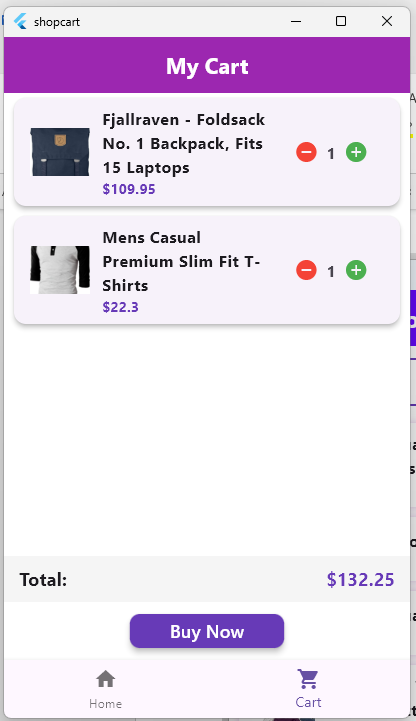
# OUTPUT:

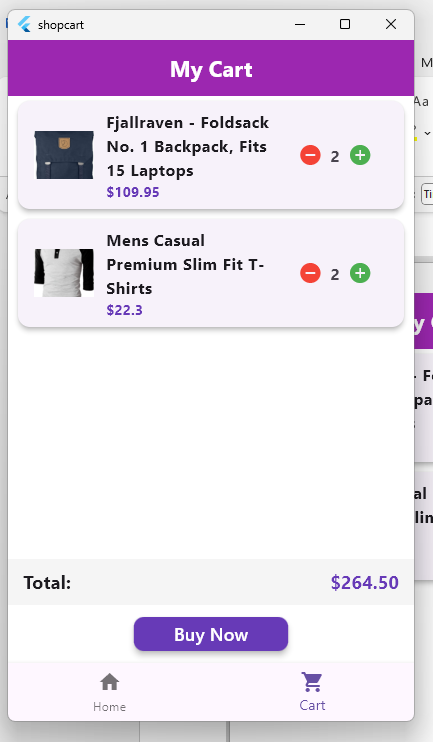


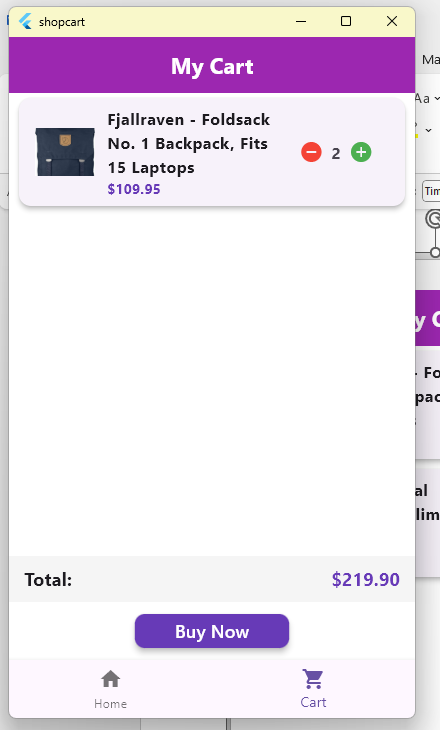












# Latest Applications:

📌 **E-commerce Apps:** Shopping platforms like Amazon or Flipkart use similar cart functionality.  
📌 **Food Ordering Apps:** Users add items to their cart before checkout, similar to Zomato or Swiggy.  
📌 **Booking Systems:** Event or hotel booking apps require cart-like selection for multiple services.  
📌 **Inventory Management:** Managing lists of products in a warehouse application.

# Learning Outcomes:

After completing this task, students will be able to:  
✔ Understand the concept of stateful widgets and state management in Flutter.  
✔ Implement setState() to update UI dynamically.  
✔ Use ListView and GridView to display product listings efficiently.  
✔ Manage navigation between different screens using Flutter’s Navigator.  
✔ Handle user interactions such as adding/removing products from the cart.