Source code:

Code:

Main.Dart

import 'package:flutter/material.dart';

import 'package:matcha/core/app\_theme.dart';

import 'package:matcha/views/location\_page.dart';

import 'package:matcha/views/notification\_page.dart';

import 'package:provider/provider.dart';

import 'package:matcha/providers/loginstate\_provider.dart';

import 'package:matcha/views/home\_page.dart';

import 'package:matcha/views/login\_page.dart';

import 'package:matcha/views/splash\_screen.dart';

import 'package:matcha/views/introPageOne.dart';

import 'package:matcha/views/introPageTwo.dart';

import 'package:matcha/views/introPageThree.dart';

import 'package:matcha/views/details\_page.dart';

import 'package:matcha/views/my\_cart\_page.dart';

import 'package:matcha/database/cart\_database.dart';

void main() async {

  WidgetsFlutterBinding.ensureInitialized();

  await CartDatabase().database; // Initialize the database

  runApp(const MyApp());

}

class MyApp extends StatelessWidget {

  const MyApp({super.key});

  @override

  Widget build(BuildContext context) {

    return ChangeNotifierProvider(

      create: (context) => LoginStateProvider(),

      child: MaterialApp(

        debugShowCheckedModeBanner: false,

        title: 'Matcha',

        theme: AppTheme.lightTheme,

        initialRoute: '/',

        routes: {

          '/': (context) => const SplashScreen(),

          '/loginPage': (context) => const LoginPage(),

          '/homePage': (context) => const HomePage(),

          '/introPageOne': (context) => const IntroPageOne(),

          '/introPageTwo': (context) => const IntroPageTwo(),

          '/introPageThree': (context) => const IntroPageThree(),

          '/detailsPage': (context) => const DetailsPage(),

          '/cart': (context) => const MyCartPage(),

          '/notificationsPage': (context) => const NotificationPage(),

          '/locationPage': (context) => const LocationPage(),

        },

      ),

    );

  }

}

Core Folder:

app\_theme.dart:

import 'package:flutter/material.dart';

import 'app.colors.dart';

class AppTheme {

  static ThemeData get lightTheme {

    return ThemeData(

      primaryColor: AppColors.primary,

      colorScheme: ColorScheme.fromSwatch().copyWith(

        primary: AppColors.primary,

        secondary: AppColors.secondary,

      ),

      scaffoldBackgroundColor: AppColors.background,

      appBarTheme: AppBarTheme(

        backgroundColor: AppColors.primary,

        titleTextStyle: TextStyle(

          color: Colors.white, fontSize: 20, fontWeight: FontWeight.bold),

          iconTheme: IconThemeData(color: Colors.white),

        ),

        floatingActionButtonTheme: FloatingActionButtonThemeData(

          backgroundColor: AppColors.secondary,

        ),

        elevatedButtonTheme: ElevatedButtonThemeData(

          style: ElevatedButton.styleFrom(

            backgroundColor: AppColors.primary,

            foregroundColor: Colors.white,

            textStyle: TextStyle(fontSize: 20, fontWeight: FontWeight.bold)

          ),

        ),

        inputDecorationTheme: InputDecorationTheme(

          border: OutlineInputBorder(), //Default border

          enabledBorder: OutlineInputBorder(

            borderSide: BorderSide(color: AppColors.primary), // Set Border color

          ),

          focusedBorder: OutlineInputBorder(

            borderSide: BorderSide(color: AppColors.secondary, width: 2.0),

          ),

        ),

      );

  }

}

app\_colors.dart:

import 'package:flutter/material.dart';

class AppColors {

  static const Color primary = Color(0xFF1976D2); //Deep Blue

  static const Color secondary = Color(0xFFFF9800); //Orange

  static const Color background = Colors.white;

  static const Color textPrimary = Colors.black87;

  static const Color textSecondary = Colors.black54;

}

constants.dart

class AppConstants {

  static const double pi = 3.141592653589793;

}

Database folder

cart\_database.dart:

import 'package:sqflite/sqflite.dart';

import 'package:path/path.dart';

import 'package:matcha/models/cart\_item.dart';

class CartDatabase {

  static final CartDatabase \_instance = CartDatabase.\_internal();

  factory CartDatabase() => \_instance;

  CartDatabase.\_internal();

  Database? \_database; // Initialize it as nullable

  // Get the database instance

  Future<Database> get database async {

    // Initialize the database if it's not already initialized

    if (\_database == null) {

      \_database = await \_initDatabase();

    }

    return \_database!;

  }

  // Initialize the database

  Future<Database> \_initDatabase() async {

    final path = await getDatabasesPath();

    return await openDatabase(

      join(path, 'cart.db'),

      onCreate: (db, version) {

        return db.execute(

          'CREATE TABLE cart\_items(id INTEGER PRIMARY KEY, title TEXT, imagePath TEXT, price REAL, quantity INTEGER)',

        );

      },

      version: 1,

    );

  }

  // Insert a cart item into the database

  Future<void> insertCartItem(CartItem cartItem) async {

    final db = await database;

    await db.insert(

      'cart\_items',

      cartItem.toMap(),

      conflictAlgorithm: ConflictAlgorithm.replace,

    );

  }

  // Get all cart items

  Future<List<CartItem>> getCartItems() async {

    final db = await database;

    final List<Map<String, dynamic>> maps = await db.query('cart\_items');

    return List.generate(maps.length, (i) {

      return CartItem.fromMap(

          maps[i]); // Use fromMap here instead of creating manually

    });

  }

  // Update a cart item

  Future<void> updateCartItem(CartItem cartItem) async {

    final db = await database;

    await db.update(

      'cart\_items',

      cartItem.toMap(),

      where: 'id = ?',

      whereArgs: [cartItem.id],

    );

  }

  // Remove a cart item

  Future<void> deleteCartItem(int id) async {

    final db = await database;

    await db.delete(

      'cart\_items',

      where: 'id = ?',

      whereArgs: [id],

    );

  }

  // Clear all cart items

  Future<void> clearCart() async {

    final db = await database;

    await db.delete('cart\_items');

  }

}

Models folder

cart\_item.dart

class CartItem {

  final int id;

  final String title;

  final String imagePath;

  final double price;

  final int quantity;

  CartItem({

    required this.id,

    required this.title,

    required this.imagePath,

    required this.price,

    required this.quantity,

  });

  // Convert CartItem to a map for inserting into the database

  Map<String, dynamic> toMap() {

    return {

      'id': id,

      'title': title,

      'imagePath': imagePath,

      'price': price,

      'quantity': quantity,

    };

  }

  // Create CartItem from a map (when fetching from the database)

  static CartItem fromMap(Map<String, dynamic> map) {

    return CartItem(

      id: map['id'],

      title: map['title'],

      imagePath: map['imagePath'],

      price: map['price'],

      quantity: map['quantity'],

    );

  }

}

cart\_model.dart:

import 'cart\_item.dart';

class CartModel {

  static final CartModel \_instance = CartModel.\_internal();

  factory CartModel() => \_instance;

  CartModel.\_internal();

  final List<CartItem> \_cartItems = [];

  List<CartItem> get cartItems => \_cartItems;

  void addItem(CartItem item) {

    \_cartItems.add(item);

  }

  void removeItem(CartItem item) {

    \_cartItems.remove(item);

  }

  void clear() {

    \_cartItems.clear();

  }

  double get totalPrice {

    return \_cartItems.fold(

        0, (sum, item) => sum + (item.price \* item.quantity));

  }

}

task\_arguments.dart:

class TaskArguments {

  final int selectedTaskId;

  TaskArguments({required this.selectedTaskId});

}

user\_model.dart:

class UserModel {

  String username;

  String password;

  bool isLoggedIn;

  String errorMessage;

  UserModel({

    this.username = ' ',

    this.password = ' ',

    this.isLoggedIn = false,

    this.errorMessage = ' ',

  });

}

Providers Folder

loginstate\_provider/dart:

import 'package:flutter/material.dart';

import 'package:matcha/models/user\_model.dart';

class LoginStateProvider extends ChangeNotifier {

  final UserModel \_user = UserModel(); //user the model to store user related

  UserModel get user => \_user; // getter to access the user model data

  //method to login - this will called form the login page

  void login(String username, String password) {

    //values passed from the login page

    if (username == 'user' && password == 'password') {

      // in project should implement data base \_ cant be hard code like this

      \_user.isLoggedIn = true;

      \_user.errorMessage = '';

    } else {

      \_user.isLoggedIn = false;

      \_user.errorMessage = 'Invalid username or password';

    }

    notifyListeners(); // to notify the UI to update (who is listener? all the pages)

  }

}

Views Folder

splash\_screen.dart:

import 'dart:async';

import 'package:flutter/material.dart';

import 'package:matcha/core/app.colors.dart';

import 'dart:math';

// Splash Screen

class SplashScreen extends StatefulWidget {

  const SplashScreen({super.key});

  @override

  SplashScreenState createState() => SplashScreenState();

}

class SplashScreenState extends State<SplashScreen>

    with SingleTickerProviderStateMixin {

  late AnimationController controller;

  @override

  void initState() {

    super.initState();

    controller = AnimationController(

      duration: const Duration(seconds: 2), // Rotation speed

      vsync: this,

    )..repeat();

    Timer(Duration(seconds: 3), () {

      Navigator.pushNamed(context, '/introPageOne');

    });

  }

  @override

  void dispose() {

    controller.dispose();

    super.dispose();

  }

  @override

  Widget build (BuildContext context) {

    return Scaffold(

      backgroundColor: AppColors.background, //Background color

      body: Center(

        child: AnimatedBuilder(

          animation: controller,

          builder: (context, child) {

            return Transform.rotate(

              angle: controller.value \* 2 \* pi, //Full rotation - AppConstants.pi from constants or dart:math to pi

              child: child,

            );

          },

          child: Image.asset('assets/icons/matcha\_logo.png', width: 150), //Rotating logo

        ),

      ),

    );

  }

}

introPageOne.dart:

import 'package:flutter/material.dart';

class IntroPageOne extends StatelessWidget {

  const IntroPageOne({super.key});

  @override

  Widget build(BuildContext context) {

return Scaffold(

      backgroundColor: Colors.grey.shade100,

      body: SafeArea(

        child: Column(

          children: [

            // Image section

            ClipRRect(

              borderRadius: BorderRadius.vertical(

                bottom: Radius.circular(40),

              ),

              child: Image.asset(

                'assets/images/board1.png',

                width: double.infinity,

                height: 600,

                fit: BoxFit.cover,

              ),

            ),

            // Content section

            Expanded(

              child: Padding(

                padding: const EdgeInsets.symmetric(horizontal: 24.0),

                child: Column(

                  mainAxisAlignment: MainAxisAlignment.spaceBetween,

                  children: [

                    Column(

                      children: [

                        const SizedBox(height: 24),

                        Text(

                          'The Amazon of Matcha',

                          style: TextStyle(

                            fontSize: 20,

                            fontWeight: FontWeight.bold,

                            color: Colors.green.shade900,

                          ),

                        ),

                        const SizedBox(height: 10),

                        Text(

                          'Sells premium matcha, accessories, and creative drink kits for users love high-quality, niche products',

                          textAlign: TextAlign.center,

                          style: TextStyle(

                            fontSize: 14,

                            color: Colors.grey.shade700,

                          ),

                        ),

                      ],

                    ),

                    // Dots and nav buttons

                    Row(

                      mainAxisAlignment: MainAxisAlignment.spaceBetween,

                      children: [

                        TextButton(

                          onPressed: () {

                            Navigator.pushNamed(context, "/loginPage");

                          },

                          child: Text(

                            'Skip',

                            style: TextStyle(

                              color: Colors.grey.shade700,

                            ),

                          ),

                        ),

                        Row(

                          children: [

                            \_buildDot(isActive: true),

                            \_buildDot(),

                            \_buildDot(),

                          ],

                        ),

                        IconButton(

                          onPressed: () {

                            Navigator.pushNamed(context, "/introPageTwo");

                          },

                          icon: Icon(

                            Icons.arrow\_forward\_ios,

                            color: Colors.grey.shade700,

                          ),

                        ),

                      ],

                    ),

                  ],

                ),

              ),

            ),

          ],

        ),

      ),

    );

  }

  Widget \_buildDot({bool isActive = false}) {

    return Container(

      margin: const EdgeInsets.symmetric(horizontal: 4),

      width: 10,

      height: 10,

      decoration: BoxDecoration(

        color: isActive ? Colors.green.shade700 : Colors.grey.shade400,

        shape: BoxShape.circle,

      ),

    );

  }

}

introPageTwo.dart:

import 'package:flutter/material.dart';

class IntroPageTwo extends StatelessWidget {

  const IntroPageTwo({super.key});

  @override

  Widget build(BuildContext context) {

  return Scaffold(

      backgroundColor: Colors.grey.shade100,

      body: SafeArea(

        child: Column(

          children: [

            // Image section

            ClipRRect(

              borderRadius: BorderRadius.vertical(

                bottom: Radius.circular(40),

              ),

              child: Image.asset(

                'assets/images/board2.png',

                width: double.infinity,

                height: 600,

                fit: BoxFit.cover,

              ),

            ),

            // Content section

            Expanded(

              child: Padding(

                padding: const EdgeInsets.symmetric(horizontal: 24.0),

                child: Column(

                  mainAxisAlignment: MainAxisAlignment.spaceBetween,

                  children: [

                    Column(

                      children: [

                        const SizedBox(height: 24),

                        Text(

                          'Matcha Match',

                          style: TextStyle(

                            fontSize: 20,

                            fontWeight: FontWeight.bold,

                            color: Colors.green.shade900,

                          ),

                        ),

                        const SizedBox(height: 10),

                        Text(

                          "Instead of swiping for friends, we're giving users a matcha hub-a place to share recipes, discover new drinks, and connect in a more organic way",

                          textAlign: TextAlign.center,

                          style: TextStyle(

                            fontSize: 14,

                            color: Colors.grey.shade700,

                          ),

                        ),

                      ],

                    ),

                    // Dots and nav buttons

                    Row(

                      mainAxisAlignment: MainAxisAlignment.spaceBetween,

                      children: [

                        TextButton(

                          onPressed: () {

                            Navigator.pushNamed(context, "/loginPage");

                          },

                          child: Text(

                            'Skip',

                            style: TextStyle(

                              color: Colors.grey.shade700,

                            ),

                          ),

                        ),

                        Row(

                          children: [

                            \_buildDot(),

                            \_buildDot(isActive: true),

                            \_buildDot(),

                          ],

                        ),

                        IconButton(

                          onPressed: () {

                            Navigator.pushNamed(context, "/introPageThree");

                          },

                          icon: Icon(

                            Icons.arrow\_forward\_ios,

                            color: Colors.grey.shade700,

                          ),

                        ),

                      ],

                    ),

                  ],

                ),

              ),

            ),

          ],

        ),

      ),

    );

  }

  Widget \_buildDot({bool isActive = false}) {

    return Container(

      margin: const EdgeInsets.symmetric(horizontal: 4),

      width: 10,

      height: 10,

      decoration: BoxDecoration(

        color: isActive ? Colors.green.shade700 : Colors.grey.shade400,

        shape: BoxShape.circle,

      ),

    );

  }

}

introPageThree.dart:

import 'package:flutter/material.dart';

class IntroPageThree extends StatelessWidget {

  const IntroPageThree({super.key});

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      backgroundColor: Colors.grey.shade100,

      body: SafeArea(

        child: Column(

          children: [

            // Image section

            ClipRRect(

              borderRadius: BorderRadius.vertical(

                bottom: Radius.circular(40),

              ),

              child: Image.asset(

                'assets/images/board3.png',

                width: double.infinity,

                height: 600,

                fit: BoxFit.cover,

              ),

            ),

            // Content section

            Expanded(

              child: Padding(

                padding: const EdgeInsets.symmetric(horizontal: 24.0),

                child: Column(

                  mainAxisAlignment: MainAxisAlignment.spaceBetween,

                  children: [

                    Column(

                      children: [

                        const SizedBox(height: 24),

                        Text(

                          'Matcha Experiences',

                          style: TextStyle(

                            fontSize: 20,

                            fontWeight: FontWeight.bold,

                            color: Colors.green.shade900,

                          ),

                        ),

                        const SizedBox(height: 10),

                        Text(

                          "Imagine matcha tasting events, exclusive cafe pop-ips, and local matcha socials where users can meet in a way that's actually natural and fun",

                          textAlign: TextAlign.center,

                          style: TextStyle(

                            fontSize: 14,

                            color: Colors.grey.shade700,

                          ),

                        ),

                      ],

                    ),

                    // Dots and nav buttons

                    Row(

                      mainAxisAlignment: MainAxisAlignment.spaceBetween,

                      children: [

                        SizedBox(width: 48), // placeholder to balance layout

                        Row(

                          children: [

                            \_buildDot(),

                            \_buildDot(),

                            \_buildDot(isActive: true),

                          ],

                        ),

                        IconButton(

                          onPressed: () {

                            Navigator.pushNamed(context, "/loginPage");

                          },

                          icon: Icon(

                            Icons.arrow\_forward\_ios,

                            color: Colors.grey.shade700,

                          ),

                        ),

                      ],

                    ),

                  ],

                ),

              ),

            ),

          ],

        ),

      ),

    );

  }

  Widget \_buildDot({bool isActive = false}) {

    return Container(

      margin: const EdgeInsets.symmetric(horizontal: 4),

      width: 10,

      height: 10,

      decoration: BoxDecoration(

        color: isActive ? Colors.green.shade700 : Colors.grey.shade400,

        shape: BoxShape.circle,

      ),

    );

  }

}

login\_page.dart:

import 'package:flutter/material.dart';

import 'package:provider/provider.dart';

import 'package:matcha/providers/loginstate\_provider.dart';

class LoginPage extends StatefulWidget {

  const LoginPage({super.key});

  @override

  LoginPageState createState() {

    return LoginPageState();

  }

}

class LoginPageState extends State<LoginPage> {

  final usernameController = TextEditingController();

  final passwordController = TextEditingController();

  @override

  void dispose() {

    usernameController.dispose();

    passwordController.dispose();

    super.dispose();

  }

  @override

  Widget build(BuildContext context) {

    final loginState = Provider.of<LoginStateProvider>(context);

    return Scaffold(

      body: Container(

        width: double.infinity,

        height: double.infinity,

        decoration: const BoxDecoration(

          image: DecorationImage(

            image: AssetImage('assets/images/login.png'), // background image

            fit: BoxFit.cover,

          ),

        ),

        child: SafeArea(

          child: Padding(

            padding: const EdgeInsets.symmetric(horizontal: 24),

            child: Column(

              crossAxisAlignment: CrossAxisAlignment.start,

              mainAxisAlignment: MainAxisAlignment.center,

              children: [

                const Text(

                  "Hello Again!",

                  style: TextStyle(

                    fontSize: 28,

                    fontWeight: FontWeight.bold,

                    color: Colors.black87,

                  ),

                ),

                const SizedBox(height: 4),

                const Text(

                  "Welcome Back",

                  style: TextStyle(

                    fontSize: 16,

                    color: Colors.black54,

                  ),

                ),

                const SizedBox(height: 40),

                TextField(

                  controller: usernameController,

                  decoration: InputDecoration(

                    hintText: "Email Address",

                    filled: true,

                    fillColor: Colors.white,

                    prefixIcon: Icon(Icons.email\_outlined),

                    contentPadding: EdgeInsets.symmetric(vertical: 20),

                    border: OutlineInputBorder(

                      borderRadius: BorderRadius.circular(30),

                      borderSide: BorderSide.none,

                    ),

                  ),

                  onChanged: (value) => loginState.user.username = value,

                ),

                const SizedBox(height: 20),

                TextField(

                  controller: passwordController,

                  obscureText: true,

                  decoration: InputDecoration(

                    hintText: "Password",

                    filled: true,

                    fillColor: Colors.white,

                    prefixIcon: Icon(Icons.lock\_outline),

                    contentPadding: EdgeInsets.symmetric(vertical: 20),

                    border: OutlineInputBorder(

                      borderRadius: BorderRadius.circular(30),

                      borderSide: BorderSide.none,

                    ),

                  ),

                  onChanged: (value) => loginState.user.password = value,

                ),

                const SizedBox(height: 30),

                SizedBox(

                  width: double.infinity,

                  child: ElevatedButton(

                    style: ElevatedButton.styleFrom(

                      backgroundColor: Color(0xFF9AA742), // matcha green

                      shape: RoundedRectangleBorder(

                        borderRadius: BorderRadius.circular(30),

                      ),

                      padding: const EdgeInsets.symmetric(vertical: 16),

                    ),

                    onPressed: () {

                      loginState.login(

                        usernameController.text,

                        passwordController.text,

                      );

                    },

                    child: const Text(

                      "Login",

                      style: TextStyle(fontSize: 16),

                    ),

                  ),

                ),

                const SizedBox(height: 16),

                Center(

                  child: TextButton(

                    onPressed: () {

                      // Forgot password logic (optional)

                    },

                    child: const Text(

                      "Forgot Password",

                      style: TextStyle(color: Colors.black54),

                    ),

                  ),

                ),

                Consumer<LoginStateProvider>(

                  builder: (context, loginState, child) {

                    if (loginState.user.errorMessage.isEmpty &&

                        loginState.user.isLoggedIn) {

                      Future.microtask(() {

                        Navigator.pushNamed(context, '/locationPage');

                      });

                      return Container();

                    } else {

                      return loginState.user.errorMessage.isNotEmpty

                          ? Text(

                              loginState.user.errorMessage,

                              style: TextStyle(color: Colors.red),

                            )

                          : Container();

                    }

                  },

                ),

              ],

            ),

          ),

        ),

      ),

    );

  }

}

home\_page.dart:

// ignore\_for\_file: unused\_local\_variable

import 'package:flutter/material.dart';

import 'package:matcha/widgets/bottom\_nav.dart';

import 'package:provider/provider.dart';

import 'package:matcha/providers/loginstate\_provider.dart';

class HomePage extends StatefulWidget {

  const HomePage({super.key});

  @override

  State<HomePage> createState() => \_HomePageState();

}

class \_HomePageState extends State<HomePage> {

  int \_selectedIndex = 0;

  void \_onTabChanged(int index) {

    setState(() {

      \_selectedIndex = index;

    });

  }

  @override

  Widget build(BuildContext context) {

    final loginState = Provider.of<LoginStateProvider>(context);

    return Scaffold(

      backgroundColor: Colors.grey.shade100,

      bottomNavigationBar: CustomBottomNavBar(

        currentIndex: \_selectedIndex,

        onTabChanged: \_onTabChanged,

      ),

      body: SafeArea(

        child: SingleChildScrollView(

          padding: const EdgeInsets.all(16),

          child: Column(

            crossAxisAlignment: CrossAxisAlignment.start,

            children: [

              Row(

                mainAxisAlignment: MainAxisAlignment.spaceBetween,

                children: [

                  Row(

                    children: [

                      const Icon(Icons.location\_on, color: Color(0xFFB5C888)),

                      const SizedBox(width: 5),

                      Text("Jess, Canada",

                          style: TextStyle(fontWeight: FontWeight.w500)),

                    ],

                  ),

                  const CircleAvatar(

                    backgroundImage: AssetImage('assets/images/board3.png'),

                    radius: 20,

                  )

                ],

              ),

              const SizedBox(height: 20),

              const Text(

                "How are you today ?",

                style: TextStyle(fontSize: 22, fontWeight: FontWeight.w600),

              ),

              const SizedBox(height: 14),

              Container(

                decoration: BoxDecoration(

                  color: Colors.grey.shade100,

                  borderRadius: BorderRadius.circular(12),

                  border: Border.all(color: Colors.grey.shade300),

                ),

                child: TextField(

                  decoration: InputDecoration(

                    contentPadding: const EdgeInsets.symmetric(

                        horizontal: 16, vertical: 14),

                    hintText: 'Search here...',

                    suffixIcon: Icon(Icons.search, color: Color(0xFFB5C888)),

                    border: InputBorder.none,

                  ),

                ),

              ),

              const SizedBox(height: 20),

              Row(

                mainAxisAlignment: MainAxisAlignment.spaceBetween,

                children: const [

                  Text("Category",

                      style: TextStyle(fontWeight: FontWeight.bold)),

                  Text("See all", style: TextStyle(color: Color(0xFFB5C888))),

                ],

              ),

              const SizedBox(height: 14),

              Row(

                mainAxisAlignment: MainAxisAlignment.spaceAround,

                children: [

                  \_buildCategoryChip("All", true),

                  \_buildCategoryChip("Matcha", false),

                  \_buildCategoryChip("Tool Kit", false, icon: Icons.build),

                  \_buildCategoryChip("Gift", false, icon: Icons.card\_giftcard),

                ],

              ),

              const SizedBox(height: 24),

              GridView.count(

                crossAxisCount: 2,

                shrinkWrap: true,

                physics: const NeverScrollableScrollPhysics(),

                childAspectRatio: 0.7,

                crossAxisSpacing: 16,

                mainAxisSpacing: 16,

                children: [

                  \_buildProductCard(context, "blend\n93 Matcha",

                      "assets/images/image1.png", "\$134.00", "\$188.00"),

                  \_buildProductCard(context, "bamboo\nwhisk",

                      "assets/images/image2.png", "\$84.00", "\$98.00"),

                  \_buildProductCard(context, "matcha kit",

                      "assets/images/image3.png", "\$150.00", "\$170.00"),

                  \_buildProductCard(context, "green gold",

                      "assets/images/image4.png", "\$120.00", "\$150.00"),

                ],

              ),

            ],

          ),

        ),

      ),

    );

  }

  Widget \_buildCategoryChip(String label, bool isSelected, {IconData? icon}) {

    return Chip(

      padding: const EdgeInsets.symmetric(horizontal: 12, vertical: 4),

      label: Row(

        children: [

          if (icon != null)

            Icon(icon,

                size: 16, color: isSelected ? Colors.white : Colors.black),

          if (icon != null) const SizedBox(width: 4),

          Text(label,

              style: TextStyle(

                  color: isSelected ? Colors.white : Colors.black,

                  fontWeight: FontWeight.w500)),

        ],

      ),

      backgroundColor:

          isSelected ? const Color(0xFFB5C888) : Colors.grey.shade200,

    );

  }

  Widget \_buildProductCard(BuildContext context, String title, String imagePath,

      String price, String oldPrice) {

    return TextButton(

      style: TextButton.styleFrom(padding: EdgeInsets.zero),

      onPressed: () {

        Navigator.pushNamed(

          context,

          '/detailsPage',

          arguments: {

            'title': title,

            'imagePath': imagePath,

            'price': price,

            'oldPrice': oldPrice,

            'description':

                "This is a high-quality product from Matcha. Enjoy the authentic taste!",

          },

        );

      },

      child: Container(

        decoration: BoxDecoration(

          borderRadius: BorderRadius.circular(16),

          image: DecorationImage(

            image: AssetImage(imagePath),

            fit: BoxFit.cover,

          ),

        ),

        child: Container(

          padding: const EdgeInsets.all(12),

          decoration: BoxDecoration(

            borderRadius: BorderRadius.circular(16),

            gradient: LinearGradient(

              begin: Alignment.bottomCenter,

              end: Alignment.topCenter,

              colors: [

                Colors.black.withOpacity(0.6),

                Colors.transparent,

              ],

            ),

          ),

          child: Column(

            mainAxisAlignment: MainAxisAlignment.end,

            crossAxisAlignment: CrossAxisAlignment.start,

            children: [

              Text(

                title,

                style: const TextStyle(

                  color: Colors.white,

                  fontSize: 16,

                  fontWeight: FontWeight.bold,

                ),

              ),

              const SizedBox(height: 4),

              Row(

                children: [

                  Text(

                    price,

                    style: const TextStyle(

                      color: Colors.white,

                      fontWeight: FontWeight.bold,

                    ),

                  ),

                  const SizedBox(width: 8),

                  Text(

                    oldPrice,

                    style: TextStyle(

                      color: Colors.white.withOpacity(0.7),

                      decoration: TextDecoration.lineThrough,

                    ),

                  ),

                ],

              )

            ],

          ),

        ),

      ),

    );

  }

}

details\_page.dart:

import 'package:flutter/material.dart';

import 'package:matcha/models/cart\_item.dart';

import 'package:matcha/database/cart\_database.dart';

class DetailsPage extends StatefulWidget {

  const DetailsPage({super.key});

  @override

  State<DetailsPage> createState() => \_DetailsPageState();

}

class \_DetailsPageState extends State<DetailsPage> {

  int quantity = 2;

  @override

  Widget build(BuildContext context) {

    final args =

        ModalRoute.of(context)?.settings.arguments as Map<String, dynamic>;

    final String title = args['title'] ?? '';

    final String imagePath = args['imagePath'] ?? '';

    final String price = args['price'] ?? '';

    final String description = args['description'] ??

        "A feisty, gorgeous matcha that's ready to slap you with its attitude. Plenty of umami and natural sweetness.";

    final double unitPrice =

        double.tryParse(price.replaceAll(RegExp(r'[^0-9.]'), '')) ?? 0.0;

    final double totalPrice = unitPrice \* quantity;

    return Scaffold(

      backgroundColor: const Color(0xFFEDEDED),

      body: SafeArea(

        child: Column(

          children: [

            // Top image section with rounded bottom

            ClipRRect(

              borderRadius: const BorderRadius.only(

                bottomLeft: Radius.circular(40),

                bottomRight: Radius.circular(40),

              ),

              child: Stack(

                children: [

                  Image.asset(

                    imagePath,

                    height: 280,

                    width: double.infinity,

                    fit: BoxFit.cover,

                  ),

                  Positioned(

                    top: 16,

                    left: 16,

                    child: IconButton(

                      icon: const Icon(Icons.arrow\_back\_ios\_new\_rounded),

                      color: Colors.white,

                      onPressed: () => Navigator.pop(context),

                    ),

                  ),

                  const Positioned(

                    top: 24,

                    right: 24,

                    child: Text(

                      "Detail",

                      style: TextStyle(

                        color: Colors.white,

                        fontSize: 16,

                        fontWeight: FontWeight.w600,

                      ),

                    ),

                  ),

                ],

              ),

            ),

            // Bottom content section

            Expanded(

              child: Container(

                width: double.infinity,

                padding:

                    const EdgeInsets.symmetric(horizontal: 24, vertical: 20),

                decoration: const BoxDecoration(

                  color: Colors.white,

                  borderRadius: BorderRadius.vertical(top: Radius.circular(40)),

                ),

                child: SingleChildScrollView(

                  child: Column(

                    crossAxisAlignment: CrossAxisAlignment.start,

                    children: [

                      Text(title,

                          style: const TextStyle(

                            fontSize: 22,

                            fontWeight: FontWeight.bold,

                          )),

                      const SizedBox(height: 4),

                      const Text(

                        "buoyant, creamy, lush",

                        style: TextStyle(color: Colors.grey),

                      ),

                      const SizedBox(height: 12),

                      Text(

                        description,

                        style: const TextStyle(fontSize: 14),

                        maxLines: 3,

                        overflow: TextOverflow.ellipsis,

                      ),

                      const Text(

                        "... Learn more",

                        style: TextStyle(color: Colors.blueAccent),

                      ),

                      const SizedBox(height: 16),

                      Row(

                        children: const [

                          Icon(Icons.star, color: Colors.amber, size: 18),

                          SizedBox(width: 4),

                          Text("4.5"),

                          SizedBox(width: 16),

                          Text("\$2.05 / serving"),

                          SizedBox(width: 16),

                          Icon(Icons.location\_on\_outlined, size: 18),

                          SizedBox(width: 4),

                          Text("Tawara"),

                        ],

                      ),

                      const SizedBox(height: 20),

                      Row(

                        mainAxisAlignment: MainAxisAlignment.spaceBetween,

                        children: [

                          Column(

                            crossAxisAlignment: CrossAxisAlignment.start,

                            children: [

                              const Text(

                                "Total amount",

                                style: TextStyle(fontSize: 14),

                              ),

                              Text(

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

                                style: const TextStyle(

                                    fontSize: 20, fontWeight: FontWeight.bold),

                              ),

                            ],

                          ),

                          Row(

                            children: [

                              IconButton(

                                onPressed: () {

                                  setState(() {

                                    if (quantity > 1) quantity--;

                                  });

                                },

                                icon: const Icon(Icons.remove\_circle\_outline),

                              ),

                              Text(

                                "$quantity",

                                style: const TextStyle(fontSize: 18),

                              ),

                              IconButton(

                                onPressed: () {

                                  setState(() => quantity++);

                                },

                                icon: const Icon(Icons.add\_circle\_outline),

                              ),

                            ],

                          )

                        ],

                      ),

                      const SizedBox(height: 20),

                      SizedBox(

                        width: double.infinity,

                        child: ElevatedButton(

                          style: ElevatedButton.styleFrom(

                            backgroundColor: const Color(0xFFB5D46D),

                            shape: RoundedRectangleBorder(

                              borderRadius: BorderRadius.circular(30),

                            ),

                            padding: const EdgeInsets.symmetric(vertical: 16),

                          ),

                          onPressed: () async {

                            final cartItem = CartItem(

                              id: DateTime.now().millisecondsSinceEpoch,

                              title: title,

                              imagePath: imagePath,

                              price: unitPrice,

                              quantity: quantity,

                            );

                            await CartDatabase().insertCartItem(cartItem);

                            Navigator.pushNamed(context, '/cart');

                          },

                          child: const Text(

                            "Add to cart",

                            style: TextStyle(

                                fontSize: 16, fontWeight: FontWeight.bold),

                          ),

                        ),

                      ),

                    ],

                  ),

                ),

              ),

            ),

          ],

        ),

      ),

    );

  }

}

location\_page.dart:

import 'package:flutter/material.dart';

import 'package:google\_maps\_flutter/google\_maps\_flutter.dart';

import 'package:geolocator/geolocator.dart';

import 'package:url\_launcher/url\_launcher.dart';

import '../widgets/drawer\_page\_scaffold.dart';

class LocationPage extends StatefulWidget {

  const LocationPage({super.key});

  @override

  State<LocationPage> createState() => \_LocationPageState();

}

class \_LocationPageState extends State<LocationPage> {

  GoogleMapController? mapController;

  LatLng? currentLocation;

  final Set<Marker> \_markers = {};

  final TextEditingController \_searchController = TextEditingController();

  @override

  void initState() {

    super.initState();

    \_determinePosition();

  }

  Future<void> \_determinePosition() async {

    bool serviceEnabled;

    LocationPermission permission;

    serviceEnabled = await Geolocator.isLocationServiceEnabled();

    if (!serviceEnabled) return;

    permission = await Geolocator.checkPermission();

    if (permission == LocationPermission.denied) {

      permission = await Geolocator.requestPermission();

      if (permission == LocationPermission.denied) return;

    }

    if (permission == LocationPermission.deniedForever) return;

    final position = await Geolocator.getCurrentPosition();

    setState(() {

      currentLocation = LatLng(position.latitude, position.longitude);

      \_markers.add(Marker(

        markerId: const MarkerId('current\_location'),

        position: currentLocation!,

        infoWindow: const InfoWindow(title: 'You are here'),

      ));

    });

    if (mapController != null) {

      mapController!.animateCamera(CameraUpdate.newLatLngZoom(currentLocation!, 15));

    }

  }

  void \_openMatchaSearch() async {

    if (currentLocation == null) return;

    final url = Uri.parse(

        'https://www.google.com/maps/search/matcha/@${currentLocation!.latitude},${currentLocation!.longitude},15z');

    if (await canLaunchUrl(url)) {

      await launchUrl(url);

    }

  }

  Widget \_buildMiniButton(IconData icon, String label, VoidCallback onPressed) {

    return SizedBox(

      width: 150,

      height: 40,

      child: ElevatedButton.icon(

        onPressed: onPressed,

        icon: Icon(icon, size: 16, color: Colors.white),

        label: Text(

          label,

          style: const TextStyle(fontSize: 12),

        ),

        style: ElevatedButton.styleFrom(

          backgroundColor: const Color(0xFF8DB255),

          padding: const EdgeInsets.symmetric(horizontal: 8),

          shape: RoundedRectangleBorder(

            borderRadius: BorderRadius.circular(20),

          ),

        ),

      ),

    );

  }

  @override

  Widget build(BuildContext context) {

    return DrawerPageScaffold(

      title: "Matcha Map",

      body: Column(

        children: [

          if (currentLocation == null)

            const Padding(

              padding: EdgeInsets.all(16.0),

              child: CircularProgressIndicator(),

            ),

          if (currentLocation != null)

            Expanded(

              child: GoogleMap(

                initialCameraPosition: CameraPosition(

                  target: currentLocation!,

                  zoom: 15,

                ),

                onMapCreated: (controller) => mapController = controller,

                markers: \_markers,

                myLocationEnabled: true,

              ),

            ),

          Padding(

            padding: const EdgeInsets.symmetric(horizontal: 16.0, vertical: 8.0),

            child: TextField(

              controller: \_searchController,

              decoration: InputDecoration(

                hintText: 'Search location or venue...',

                prefixIcon: const Icon(Icons.search),

                filled: true,

                fillColor: Colors.white,

                contentPadding: const EdgeInsets.symmetric(vertical: 10),

                border: OutlineInputBorder(

                  borderRadius: BorderRadius.circular(12.0),

                  borderSide: BorderSide.none,

                ),

              ),

            ),

          ),

          Padding(

            padding: const EdgeInsets.symmetric(horizontal: 16.0, vertical: 4.0),

            child: Wrap(

              spacing: 8,

              runSpacing: 8,

              alignment: WrapAlignment.center,

              children: [

                \_buildMiniButton(Icons.local\_cafe, 'Matcha Near Me', \_openMatchaSearch),

                \_buildMiniButton(Icons.home, 'Home Page', () {

                  Navigator.pushNamed(context, '/homePage');

                }),

                \_buildMiniButton(Icons.favorite, 'Matching Mate', () {

                  // Navigate or show matching mates

                }),

                \_buildMiniButton(Icons.event, 'Events', () {

                  // Navigate or show events

                }),

              ],

            ),

          ),

          const NotificationSection(),

        ],

      ),

    );

  }

}

class NotificationSection extends StatelessWidget {

  const NotificationSection({super.key});

  @override

  Widget build(BuildContext context) {

    return Card(

      margin: const EdgeInsets.symmetric(horizontal: 16, vertical: 8),

      child: const ListTile(

        leading: Icon(Icons.location\_on, color: Colors.green),

        title: Text("Location services are active"),

        subtitle: Text("Using your location to find nearby matcha spots"),

        trailing: Icon(Icons.check\_circle, color: Colors.green),

      ),

    );

  }

}

my\_cart\_page.dart:

import 'package:flutter/material.dart';

import 'package:matcha/database/cart\_database.dart';

import 'package:matcha/models/cart\_item.dart';

import 'package:matcha/widgets/bottom\_nav.dart';

class MyCartPage extends StatefulWidget {

  const MyCartPage({super.key});

  @override

  \_MyCartPageState createState() => \_MyCartPageState();

}

class \_MyCartPageState extends State<MyCartPage> {

  late Future<List<CartItem>> \_cartItems;

  @override

  void initState() {

    super.initState();

    \_cartItems = CartDatabase().getCartItems();

  }

  void \_refreshCartItems() {

    setState(() {

      \_cartItems = CartDatabase().getCartItems();

    });

  }

  void \_updateQuantity(CartItem item, int newQuantity) async {

    if (newQuantity < 1) {

      await CartDatabase().deleteCartItem(item.id);

    } else {

      await CartDatabase().updateCartItem(

        CartItem(

          id: item.id,

          title: item.title,

          imagePath: item.imagePath,

          price: item.price,

          quantity: newQuantity,

        ),

      );

    }

    \_refreshCartItems();

  }

  double \_calculateTotal(List<CartItem> items) {

    return items.fold(0, (sum, item) => sum + item.price \* item.quantity);

  }

  void \_onTabChanged(int index) {

    // Handled by CustomBottomNavBar

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      backgroundColor: const Color(0xFFF2F2F2),

      appBar: AppBar(

        title: const Text(

          'My Cart',

          style: TextStyle(color: Colors.black, fontWeight: FontWeight.bold),

        ),

        centerTitle: true,

        backgroundColor: Colors.transparent,

        elevation: 0,

        iconTheme: const IconThemeData(color: Colors.black),

        leading: IconButton(

          icon: const Icon(Icons.arrow\_back),

          onPressed: () {

            Navigator.pushReplacementNamed(context, '/homePage');

          },

        ),

      ),

      body: SafeArea(

        child: FutureBuilder<List<CartItem>>(

          future: \_cartItems,

          builder: (context, snapshot) {

            if (snapshot.connectionState == ConnectionState.waiting) {

              return const Center(child: CircularProgressIndicator());

            }

            final items = snapshot.data ?? [];

            if (items.isEmpty) {

              return const Center(

                child: Text("Your cart is empty."),

              );

            }

            double total = \_calculateTotal(items);

            return Column(

              children: [

                Expanded(

                  child: ListView.builder(

                    padding: const EdgeInsets.only(bottom: 16),

                    itemCount: items.length,

                    itemBuilder: (context, index) {

                      final item = items[index];

                      return Card(

                        margin: const EdgeInsets.symmetric(

                            horizontal: 16, vertical: 8),

                        shape: RoundedRectangleBorder(

                            borderRadius: BorderRadius.circular(16)),

                        elevation: 2,

                        child: Padding(

                          padding: const EdgeInsets.all(12.0),

                          child: Row(

                            crossAxisAlignment: CrossAxisAlignment.start,

                            children: [

                              ClipRRect(

                                borderRadius: BorderRadius.circular(12),

                                child: Image.asset(

                                  item.imagePath,

                                  width: 80,

                                  height: 80,

                                  fit: BoxFit.cover,

                                ),

                              ),

                              const SizedBox(width: 12),

                              Expanded(

                                child: Column(

                                  crossAxisAlignment: CrossAxisAlignment.start,

                                  children: [

                                    Text(

                                      item.title,

                                      style: const TextStyle(

                                        fontSize: 16,

                                        fontWeight: FontWeight.bold,

                                      ),

                                    ),

                                    const SizedBox(height: 4),

                                    // Text(

                                    //   "Size: ${item.size}",

                                    //   style: const TextStyle(

                                    //     color: Colors.grey,

                                    //     fontSize: 14,

                                    //   ),

                                    // ),

                                    const SizedBox(height: 8),

                                    Row(

                                      mainAxisAlignment:

                                          MainAxisAlignment.spaceBetween,

                                      children: [

                                        Text(

                                          "\$${(item.price \* item.quantity).toStringAsFixed(2)}",

                                          style: const TextStyle(

                                            fontWeight: FontWeight.bold,

                                            fontSize: 16,

                                          ),

                                        ),

                                        Row(

                                          children: [

                                            IconButton(

                                              icon: const Icon(

                                                  Icons.remove\_circle\_outline),

                                              onPressed: () => \_updateQuantity(

                                                  item, item.quantity - 1),

                                            ),

                                            Text('${item.quantity}'),

                                            IconButton(

                                              icon: const Icon(

                                                  Icons.add\_circle\_outline),

                                              onPressed: () => \_updateQuantity(

                                                  item, item.quantity + 1),

                                            ),

                                          ],

                                        ),

                                      ],

                                    ),

                                  ],

                                ),

                              ),

                            ],

                          ),

                        ),

                      );

                    },

                  ),

                ),

                Container(

                  padding:

                      const EdgeInsets.symmetric(horizontal: 16, vertical: 12),

                  decoration: const BoxDecoration(

                    color: Colors.white,

                    boxShadow: [

                      BoxShadow(

                        color: Colors.black12,

                        blurRadius: 6,

                        offset: Offset(0, -2),

                      ),

                    ],

                  ),

                  child: Column(

                    children: [

                      Row(

                        mainAxisAlignment: MainAxisAlignment.spaceBetween,

                        children: [

                          const Text(

                            "Total:",

                            style: TextStyle(

                              fontSize: 18,

                              fontWeight: FontWeight.bold,

                            ),

                          ),

                          Text(

                            "\$${total.toStringAsFixed(2)}",

                            style: const TextStyle(

                              fontSize: 18,

                              fontWeight: FontWeight.bold,

                              color: Colors.green,

                            ),

                          ),

                        ],

                      ),

                      const SizedBox(height: 12),

                      SizedBox(

                        width: double.infinity,

                        height: 50,

                        child: ElevatedButton(

                          onPressed: () {

                            // Implement order functionality here

                            ScaffoldMessenger.of(context).showSnackBar(

                              const SnackBar(content: Text("Order placed!")),

                            );

                          },

                          style: ElevatedButton.styleFrom(

                            backgroundColor: Colors.green,

                            shape: RoundedRectangleBorder(

                              borderRadius: BorderRadius.circular(12),

                            ),

                          ),

                          child: const Text(

                            "Order Now",

                            style: TextStyle(fontSize: 16),

                          ),

                        ),

                      ),

                    ],

                  ),

                ),

              ],

            );

          },

        ),

      ),

      bottomNavigationBar: CustomBottomNavBar(

        currentIndex: 2,

        onTabChanged: \_onTabChanged,

      ),

    );

  }

}

notification\_page.dart

import 'package:flutter/material.dart';

class NotificationPage extends StatelessWidget {

  const NotificationPage({super.key});

  @override

  Widget build(BuildContext context) {

    return WillPopScope(

      onWillPop: () async {

        Navigator.pushReplacementNamed(context, '/homePage');

        return false;

      },

      child: Scaffold(

        backgroundColor:

            const Color.fromARGB(255, 203, 215, 207), // soft matcha background

        appBar: AppBar(

          backgroundColor:

              const Color.fromARGB(255, 119, 155, 133), // matcha pastel green

          elevation: 0,

          leading: IconButton(

            icon: const Icon(Icons.arrow\_back, color: Colors.white),

            onPressed: () {

              Navigator.pushReplacementNamed(context, '/homePage');

            },

          ),

          title: const Text(

            "Notifications",

            style: TextStyle(

              fontWeight: FontWeight.bold,

              color: Colors.white,

            ),

          ),

        ),

        body: ListView(

          padding: const EdgeInsets.all(16),

          children: [

            \_SectionHeader(title: "Today"),

            const SizedBox(height: 12),

            const \_NotificationTile(

              icon: Icons.local\_shipping,

              title: "Order Shipped",

              subtitle:

                  "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",

              time: "1h",

            ),

            const \_NotificationTile(

              icon: Icons.flash\_on,

              title: "Flash Sale Alert",

              subtitle:

                  "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",

              time: "1h",

            ),

            const \_NotificationTile(

              icon: Icons.star\_border,

              title: "Product Review Request",

              subtitle:

                  "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",

              time: "1h",

            ),

            const SizedBox(height: 24),

            \_SectionHeader(title: "Yesterday"),

            const SizedBox(height: 12),

            const \_NotificationTile(

              icon: Icons.local\_shipping\_outlined,

              title: "Order Shipped",

              subtitle:

                  "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",

              time: "1d",

            ),

            const \_NotificationTile(

              icon: Icons.account\_balance\_wallet\_outlined,

              title: "New Paypal Added",

              subtitle:

                  "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",

              time: "1d",

            ),

          ],

        ),

      ),

    );

  }

}

class \_SectionHeader extends StatelessWidget {

  final String title;

  const \_SectionHeader({required this.title});

  @override

  Widget build(BuildContext context) {

    return Row(

      mainAxisAlignment: MainAxisAlignment.spaceBetween,

      children: [

        Text(

          title,

          style: const TextStyle(

            fontSize: 16,

            fontWeight: FontWeight.bold,

            color: Colors.black87,

          ),

        ),

        Text(

          "Mark all as read",

          style: TextStyle(

            fontSize: 14,

            color: Colors.green.shade700,

            fontWeight: FontWeight.w500,

          ),

        )

      ],

    );

  }

}

class \_NotificationTile extends StatelessWidget {

  final IconData icon;

  final String title;

  final String subtitle;

  final String time;

  const \_NotificationTile({

    required this.icon,

    required this.title,

    required this.subtitle,

    required this.time,

  });

  @override

  Widget build(BuildContext context) {

    return Card(

      margin: const EdgeInsets.symmetric(vertical: 6),

      shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(16)),

      elevation: 2,

      child: Padding(

        padding: const EdgeInsets.all(16),

        child: Row(

          children: [

            Container(

              decoration: BoxDecoration(

                color: const Color(0xFFF0FFF4),

                borderRadius: BorderRadius.circular(12),

              ),

              padding: const EdgeInsets.all(8),

              child: Icon(icon, size: 28, color: const Color(0xFF4CAF50)),

            ),

            const SizedBox(width: 16),

            Expanded(

              child: Column(

                crossAxisAlignment: CrossAxisAlignment.start,

                children: [

                  Text(title,

                      style: const TextStyle(

                        fontSize: 15,

                        fontWeight: FontWeight.w600,

                      )),

                  const SizedBox(height: 4),

                  Text(

                    subtitle,

                    style: TextStyle(

                      fontSize: 13,

                      color: Colors.grey.shade600,

                    ),

                  ),

                ],

              ),

            ),

            const SizedBox(width: 8),

            Text(

              time,

              style: TextStyle(

                fontSize: 12,

                color: Colors.grey.shade500,

              ),

            )

          ],

        ),

      ),

    );

  }

}

Widgets Folder

bottom\_nav.dart:

import 'package:flutter/material.dart';

class CustomBottomNavBar extends StatelessWidget {

  final int currentIndex;

  final Function(int) onTabChanged;

  const CustomBottomNavBar({

    super.key,

    required this.currentIndex,

    required this.onTabChanged,

  });

  @override

  Widget build(BuildContext context) {

    return BottomNavigationBar(

      type: BottomNavigationBarType.fixed,

      currentIndex: currentIndex,

      onTap: (index) => onNavItemTapped(context, index),

      items: const [

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

        BottomNavigationBarItem(icon: Icon(Icons.search), label: "Nearby"),

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

        BottomNavigationBarItem(

            icon: Icon(Icons.notifications), label: "Notifications"),

      ],

      selectedItemColor: Colors.green,

      unselectedItemColor: Colors.grey,

    );

  }

  void onNavItemTapped(BuildContext context, int index) {

    // If the index is already selected, do nothing

    if (currentIndex == index) return;

    // Change the parent widget's state for navigation

    onTabChanged(index);

    // Perform the actual navigation

    switch (index) {

      case 0:

        Navigator.pushReplacementNamed(context, "/homePage");

        break;

      case 1:

        Navigator.pushReplacementNamed(context, "/locationPage");

        break;

      case 2:

        Navigator.pushReplacementNamed(context, "/cart");

        break;

      case 3:

        Navigator.pushReplacementNamed(context, "/notificationsPage");

        break;

    }

  }

}

custom\_drawer.dart:

import 'package:flutter/material.dart';

import 'package:provider/provider.dart';

import 'package:matcha/providers/loginstate\_provider.dart';

class CustomDrawer extends StatelessWidget {

  const CustomDrawer({super.key});

  @override

  Widget build(BuildContext context) {

    final loginState = Provider.of<LoginStateProvider>(context);

    return Drawer(

      child: Column(

        children: [

          UserAccountsDrawerHeader(

            accountName: Text(

              loginState.user.username,

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

            ),

            accountEmail: Text("Welcome to Matcha"),

            currentAccountPicture: CircleAvatar(

              backgroundColor: Colors.white,

              child: Icon(Icons.person, size: 40, color: Colors.blue),

            ),

          ),

          Expanded(

            child: ListView(

              children: [

                \_buildDrawerItem(Icons.home, "Home", context, "/homePage", false),

              ],

            ),

          ),

          Divider(),

          \_buildDrawerItem(Icons.logout, "Logout", context, "/loginPage", true),

        ],

      ),

    );

  }

  Widget \_buildDrawerItem(IconData icon, String title, BuildContext context, String route, isLogout) {

    final loginState = Provider.of<LoginStateProvider>(context);

    return ListTile(

      leading: Icon(icon),

      title: Text(title),

      onTap: () {

        Navigator.pop(context);

        if (isLogout) {

          loginState.user.isLoggedIn=false;

          //Clear the navigation stack and go to the login page

          Navigator.pushNamedAndRemoveUntil(

            context,

            "/loginPage",

            (route) => false, //Removes all previous routes

          );

        } else {

          Navigator.pushNamed(context, route);

        }

      },

    );

  }

}

drawer\_page\_scaffold.dart:

import 'package:flutter/material.dart';

class DrawerPageScaffold extends StatelessWidget {

  final Widget body;

  final String? title;

  const DrawerPageScaffold({

    super.key,

    required this.body,

    this.title,

  });

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: Text(title ?? 'Matcha App'),

      ),

      drawer: Drawer(

        child: ListView(

          padding: EdgeInsets.zero,

          children: [

            const DrawerHeader(

              decoration: BoxDecoration(

                color: Colors.green,

              ),

              child: Text(

                'Matcha Menu',

                style: TextStyle(

                  color: Colors.white,

                  fontSize: 24,

                ),

              ),

            ),

            ListTile(

              leading: Icon(Icons.map),

              title: Text('Map'),

              onTap: () {

                Navigator.pushNamed(context, '/location');

              },

            ),

            ListTile(

              leading: Icon(Icons.local\_cafe),

              title: Text('Find Matcha'),

              onTap: () {

                Navigator.pop(context);

              },

            ),

          ],

        ),

      ),

      body: body,

    );

  }

}

Screenshots:

When Loading the terminal App:

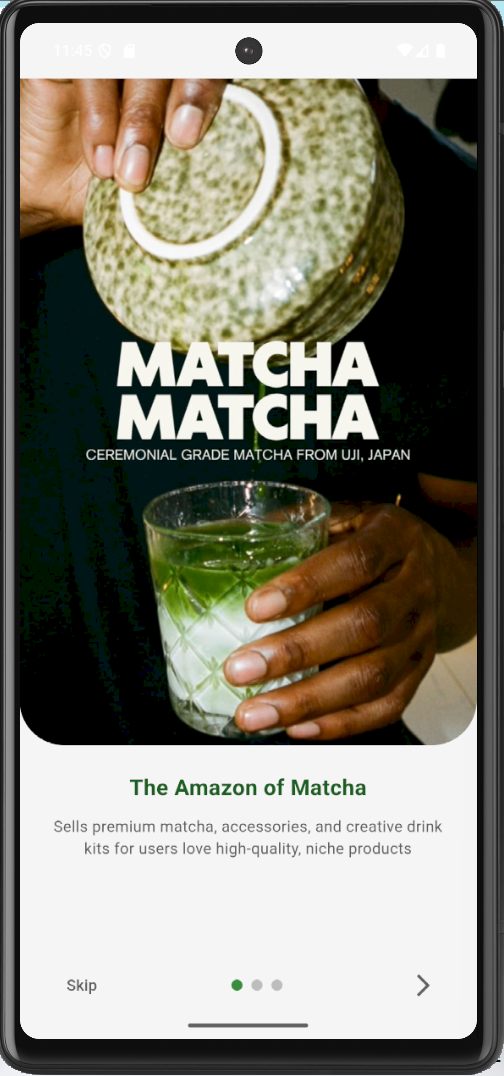
A screen shot of a phone

AI-generated content may be incorrect.

Splash Screen:



Intro Screen One:



Intro Screen Two:

A screenshot of a cellphone

AI-generated content may be incorrect.

Intro Screen Three:

A person holding two glasses of green liquid

AI-generated content may be incorrect.

Login Screen:

A screenshot of a phone

AI-generated content may be incorrect.

Home Screen:

A screenshot of a cell phone

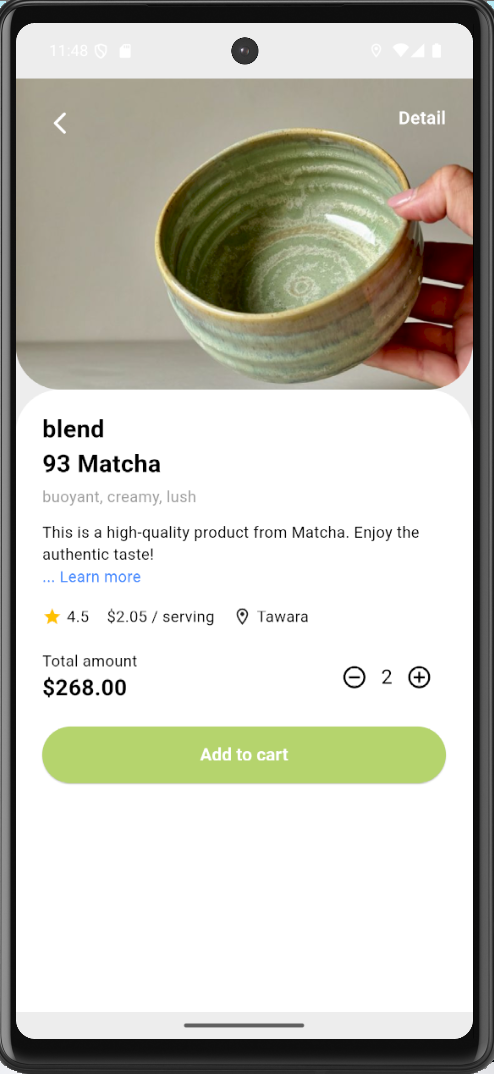
AI-generated content may be incorrect.

Detail Screen (My Cart):

A screenshot of a phone

AI-generated content may be incorrect.

Details Item One:



Details Item Two:

A screenshot of a phone

AI-generated content may be incorrect.

Details Item Three:

A screenshot of a phone

AI-generated content may be incorrect.

Details Item Four:

A screenshot of a phone

AI-generated content may be incorrect.

Notifications Screen:

A screenshot of a phone

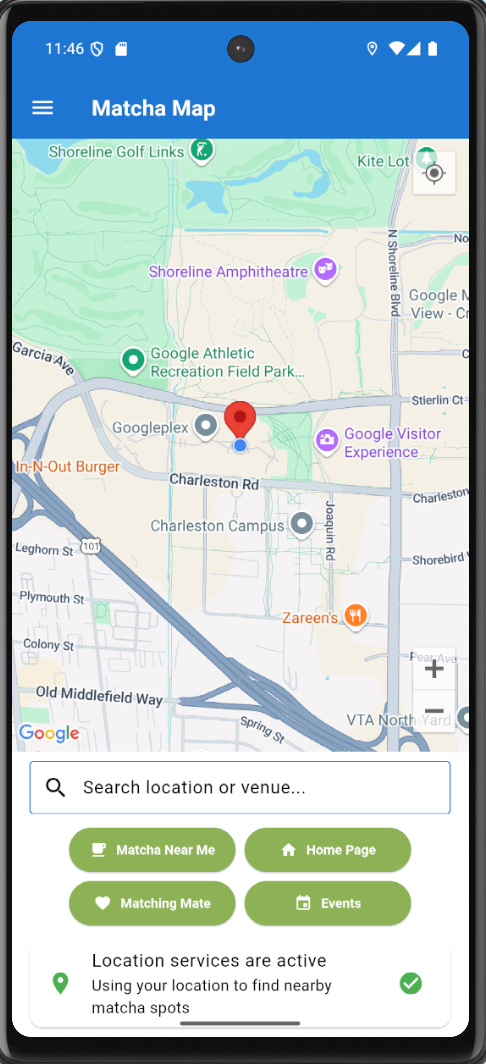
AI-generated content may be incorrect.

Ask for Location:

A screenshot of a phone

AI-generated content may be incorrect.

Map Screen:



Map Detail:

A screenshot of a phone

AI-generated content may be incorrect.