

Project Report

On

CAMPUS BITE

Submitted in partial fulfillment of the requirements for the award of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

(Artificial Intelligence & Machine Learning)

by

Ms. R.ASHRITHA - 22WH1A6626

Ms. M.SAATVIKA- 22WH1A6628

Ms. M.NITHYASRI- 22WH1A6630

Ms. B. RISHITA REDDY-22WH1A6631

Under the esteemed guidance of

Ms. S Annapoorna

Assistant Professor,CSE(AI&ML)



BVRITHYDERABAD College of Engineering for Women

(UGC Autonomous Institution |Approved by AICTE |Affiliated to JNTUH)

(NAAC Accredited –A Grade |NBA Accredited B.Tech.(EEE, ECE,CSE and IT)

Bachupally, Hyderabad – 500090

2024-25

Department of Computer Science & Engineering
(Artificial Intelligence & Machine Learning)
BVRIT HYDERABAD COLLEGE OF ENGINEERING FOR WOMENS
(Approved by AICTE, New Delhi and Affiliated to JNTUH, Hyderabad)
Accredited by NBA and NAAC with A Grade
Bachupally, Hyderabad – 500090

2024-25



CERTIFICATE

This is to certify that the major project entitled “**Campus Bite using Flutter**” is a bonafide work carried out by **Ms.R Ashritha (22wh1a6626), Ms.M Saatvika (22wh1a6628), Ms.M Nithyasri (22wh1a6630), Ms.B Rishita (22wh1a6631)** in partial fulfillment for the award of B.Tech degree in **Computer Science & Engineering (AI&ML), BVRIT HYDERABAD College of Engineering for Women, Bachupally, Hyderabad**, affiliated to Jawaharlal Nehru Technological University Hyderabad, Hyderabad under my guidance and supervision. The results embodied in the project work have not been submitted to any other University or Institute for the award of any degree or diploma.

Supervisor
Ms.S Annapoorna
Assistant Professor
Dept of CSE(AI&ML)

Head of the Department
Dr.B Lakshmi Praveena
HOD&Professor
Dept of CSE(AI&ML)

External Examiner

DECLARATION

We here by declare that the work presented in this project entitled “**Habit tracker using Flutter**” submitted towards completion of Project work in IV Year of B.Tech of CSE(AI&ML) at **BVRIT HYDERABAD College of Engineering for Women**, Hyderabad is an authentic record of our original work carried out under the guidance of **Ms. S Annapoorna, Assistant Professor, Department of CSE(AI&ML)**.

Sign with Date:

R.Ashritha

(22wh1a6626)

Sign with Date:

M. Saatvika

(22wh1a6628)

Sign with Date:

M. NithyaSri

(22wh1a6630)

Sign with Date:

B. Rishita Reddy

(22wh1a6631)

ACKNOWLEDGEMENT

We would like to express our sincere thanks to **Dr. K V N Sunitha, Principal, BVRIT HYDERABAD College of Engineering for Women**, for her support by providing the working facilities in the college.

Our sincere thanks and gratitude to **Dr B Lakshmi Praveena, Head of the Department, Department of CSE(AI&ML), BVRIT HYDERABAD College of Engineering for Women**, for all timely support and valuable suggestions during the period of our project.

We are extremely thankful to our Internal Guide, **Ms. S Annapoorna, Assistant Professor, CSE(AI&ML), BVRIT HYDERABAD College of Engineering for Women**, for her constant guidance and encouragement throughout the project.

Finally, we would like to thank our Major Project Coordinator, all Faculty and Staff of CSE(AI&ML) department who helped us directly or indirectly. Last but not least, we wish to acknowledge our **Parents and Friends** for giving moral strength and constant encouragement.

R Ashritha (22wh1a6626)

M Saatvika (22wh1a6628)

M NithyaSri (22wh1a6630)

B Rishita Reddy (22wh1a6631)

ABSTRACT

In the modern fast-paced world, developing and maintaining healthy habits is crucial for personal growth and productivity. A habit tracker application provides users with a convenient way to monitor, manage, and sustain their daily routines. This project explores the development of a **Food ordering app for the college premises** using **Flutter**, a versatile open-source UI software development kit by Google.

The proposed habit tracker is a cross-platform application capable of running seamlessly on both Android and iOS devices. The application leverages Flutter's robust framework to deliver a visually appealing and highly responsive user interface.

The app architecture employs **MVVM (Model-View-ViewModel)** principles, ensuring a clean and scalable codebase. State management is implemented using **Provider**, **Riverpod**, or **Bloc** (depending on requirements), enhancing maintainability. The application also includes responsive design principles to optimize the user experience across various screen sizes.

The proposed solution emphasizes accessibility, usability, and scalability, aiming to help users achieve their personal goals effectively. Future iterations can incorporate advanced features like habit sharing, social leaderboards, and AI-driven habit recommendations, further enhancing user engagement.

This project demonstrates the power of Flutter in creating dynamic, functional, and user-centric mobile applications while addressing a universal need for self-improvement.

PROBLEM STATEMENT

In college cafeterias and food stalls, students often face challenges such as long waiting times during peak hours, limited break periods, and inefficient order management. This leads to delays in receiving food, reduced time for relaxation, and increased stress for both students and stall owners. There is a need for a streamlined system that allows students to pre-order food, make payments in advance, and generate a token for efficient pickup, thereby saving time and enhancing the overall dining experience while ensuring seamless operations for stall owners.

FILE STRUCTURE

UI_Flutter

- |— .dart_tool
- |— .idea
- |— .vscode
- |— android
- |— assets
 - | |— images
 - | |— data.json
 - | |— data1.json
- |— build
- |— images
- |— ios
- |— lib
 - | |— components
 - | |— pages
 - | | |— cart_manager.dart
 - | | |— cart_page1.dart
 - | | |— menu_page.dart
 - | | |— payment.dart
 - | |— screens
 - | | |— home.dart
 - | | |— signin_screen.dart
 - | | |— signup_screen.dart
 - | |— theme
 - | | |— theme.dart
 - | |— utils
 - | |— widgets
 - | | |— custom_scaffold.dart
 - | | |— search_widget.dart
 - | | |— welcome_button.dart
 - | |— main.dart
- |— linux

- |— macos
- |— test
- |— web
- |— windows
- |— .gitattributes
- |— .gitignore
- |— .metadata
- |— analysis_options.yaml
- |— login_signup.iml
- |— pubspec.lock
- |— pubspec.yaml
- └— README.md

SOURCE CODE

Main.dart

```
import 'package:flutter/material.dart';
import 'package:login_signup/screens/welcome_screen.dart';
import 'package:login_signup/theme/theme.dart';

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'Campus Bite',
      theme: lightMode,
      home: const WelcomeScreen(),
    );
  }
}
```

welcome_screen.dart

```
import 'package:flutter/material.dart';
import 'package:login_signup/screens/signin_screen.dart';
import 'package:login_signup/screens/signup_screen.dart';
import 'package:login_signup/widgets/custom_scaffold.dart';
import 'package:login_signup/widgets/welcome_button.dart';
```

```
class WelcomeScreen extends StatelessWidget {
  const WelcomeScreen({super.key});
```

```
  @override
```

```
  Widget build(BuildContext context) {
```

```
    return CustomScaffold(
```

```
      drawer: Drawer(
```

```
        child: ListView(
```

```
          children: <Widget>[
```

```
            ListTile(
```

```
              title: Text('Item 1'),
```

```
              onTap: () {},
```

```
            ),
```

```
            ListTile(
```

```
              title: Text('Item 2'),
```

```
              onTap: () {},
```

```
            ),
```

```
            ListTile(
```

```
              title: Text('Item 3'),
```

```
              onTap: () {},
```

```
            ),
```

```
          ],
```

```
        ),
```

```
      ),
```

```
      child: Column(
```

```
        children: [
```

```
          Flexible(
```

```
            flex: 5,
```

```
            child: Container(
```

```
              padding: const EdgeInsets.symmetric(
```

```
                vertical: 0,
```

```
                horizontal: 40.0,
```

```
              ),
```

```
              child: Center(
```

```
                child: RichText(
```

```
textAlign: TextAlign.center,
text: const TextSpan(
  children: [
    TextSpan(
      text: 'Welcome Back!\n',
      style: TextStyle(
        fontSize: 45.0,
        fontWeight: FontWeight.w600,
        color: Color.fromARGB(255, 21, 21, 21),
      ),
    ),
    TextSpan(
      text: '\nEnter your personal details',
      style: TextStyle(
        fontSize: 20.0,
        height: 1.5,
        color: Color.fromARGB(255, 21, 21, 21),
      ),
    ),
  ],
),
),
),
),
const Flexible(
  flex: 1,
  child: Align(
    alignment: Alignment.bottomRight,
    child: Row(
      children: [
        Expanded(
          child: WelcomeButton(
            buttonText: 'Sign in',
            onTap: SignInScreen(),
            color: Color.fromARGB(255, 242, 241, 241),
            textColor: Color.fromARGB(255, 233, 178, 95),
          ),
        ),
        Expanded(
          child: WelcomeButton(
            buttonText: 'Sign up',
            onTap: SignUpScreen(),
            color: Color.fromARGB(255, 227, 173, 92),
            textColor: Color.fromARGB(255, 244, 242, 239),
```

```

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

```

signup_screen.dart

```

import 'package:flutter/material.dart';
import 'package:icons_plus/icons_plus.dart';
import 'package:login_signup/screens/signin_screen.dart';
import 'package:login_signup/theme/theme.dart';
import 'package:login_signup/widgets/custom_scaffold.dart';

class SignUpScreen extends StatefulWidget {
  const SignUpScreen({super.key});

  @override
  State<SignUpScreen> createState() => _SignUpScreenState();
}

class _SignUpScreenState extends State<SignUpScreen> {
  final _formSignupKey = GlobalKey<FormState>();
  bool agreePersonalData = true;

  @override
  Widget build(BuildContext context) {
    return CustomScaffold(
      drawer: Drawer(
        child: ListView(
          children: <Widget>[
            ListTile(
              title: Text('Item 1'),
              onTap: () {},
            ),
            ListTile(
              title: Text('Item 2'),
              onTap: () {},
            ),
          ],
        ),
      ),
    );
  }
}

```

```

    ),
    ListTile(
      title: Text('Item 3'),
      onTap: () {},
    ),
  ],
),
),
child: Column(
  children: [
    const Expanded(
      flex: 1,
      child: SizedBox(
        height: 10,
      ),
    ),
    Expanded(
      flex: 7,
      child: Container(
        padding: const EdgeInsets.fromLTRB(25.0, 50.0, 25.0, 20.0),
        decoration: const BoxDecoration(
          color: Colors.white,
          borderRadius: BorderRadius.only(
            topLeft: Radius.circular(40.0),
            topRight: Radius.circular(40.0),
          ),
        ),
      ),
      child: SingleChildScrollView(
        // Get started form
        child: Form(
          key: _formSignupKey,
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.center,
            children: [
              // Get started text
              Text(
                'Get Started',
                style: TextStyle(
                  fontSize: 30.0,
                  fontWeight: FontWeight.w900,
                  color: lightColorScheme.primary,
                ),
              ),
              const SizedBox(
                height: 40.0,

```

```

    ),
    // Full name
    TextFormField(
      validator: (value) {
        if (value == null || value.isEmpty) {
          return 'Please enter Full name';
        }
        return null;
      },
      decoration: InputDecoration(
        label: const Text('Full Name'),
        hintText: 'Enter Full Name',
        hintStyle: const TextStyle(
          color: Colors.black26,
        ),
        border: OutlineInputBorder(
          borderRadius: BorderRadius.circular(10),
        ),
      ),
    ),
    const SizedBox(
      height: 25.0,
    ),
    // Email
    TextFormField(
      validator: (value) {
        if (value == null || value.isEmpty) {
          return 'Please enter Email';
        }
        // Regex for email validation
        String pattern =
          r'^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$';
        if (!RegExp(pattern).hasMatch(value)) {
          return 'Please enter a valid Email';
        }
        return null;
      },
      decoration: InputDecoration(
        label: const Text('Email'),
        hintText: 'Enter Email',
        hintStyle: const TextStyle(
          color: Colors.black26,
        ),
        border: OutlineInputBorder(
          borderRadius: BorderRadius.circular(10),

```

```

        ),
    ),
),
const SizedBox(
  height: 25.0,
),
// Password
TextFormField(
  obscureText: true,
  obscuringCharacter: '*',
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter Password';
    }
    // Regex for password validation
    String pattern =
      r'^(?=.*[A-Z])(?=.*[!@#\$%&*~])(?=.*\d){8,}$';
    if (!RegExp(pattern).hasMatch(value)) {
      return 'Password must contain at least 1 uppercase, 1
number, and be 8+ characters long';
    }
    return null;
  },
  decoration: InputDecoration(
    label: const Text('Password'),
    hintText: 'Enter Password',
    hintStyle: const TextStyle(
      color: Colors.black26,
    ),
    border: OutlineInputBorder(
      borderRadius: BorderRadius.circular(10),
    ),
  ),
),
const SizedBox(
  height: 25.0,
),
// I agree to the processing
Row(
  children: [
    Checkbox(
      value: agreePersonalData,
      onChanged: (bool? value) {
        setState(() {
          agreePersonalData = value!;

```

```

        ));
    },
    activeColor: lightColorScheme.primary,
),
const Text(
  'I agree to the processing of ',
  style: TextStyle(
    color: Colors.black45,
  ),
),
Text(
  'Personal data',
  style: TextStyle(
    fontWeight: FontWeight.bold,
    color: lightColorScheme.primary,
  ),
),
],
),
const SizedBox(
  height: 25.0,
),
// Signup button
SizedBox(
  width: double.infinity,
  child: ElevatedButton(
    onPressed: () {
      if (_formSignupKey.currentState!.validate() &&
        agreePersonalData) {
        Navigator.push(
          context,
          MaterialPageRoute(
            builder: (context) => const SignInScreen(),
          );
      } else if (!agreePersonalData) {
        ScaffoldMessenger.of(context).showSnackBar(
          const SnackBar(
            content: Text(
              'Please agree to the processing of personal data'),
          );
      }
    },
    child: const Text('Sign up'),
  ),
),

```



```

const SizedBox(
  height: 30.0,
),
// Sign up divider
Row(
  mainAxisAlignment: MainAxisAlignment.center,
  children: [
    Expanded(
      child: Divider(
        thickness: 0.7,
        color: Colors.grey.withOpacity(0.5),
      ),
    ),
    const Padding(
      padding: EdgeInsets.symmetric(
        vertical: 0,
        horizontal: 10,
      ),
      child: Text(
        'Sign up with',
        style: TextStyle(
          color: Colors.black45,
        ),
      ),
    ),
    Expanded(
      child: Divider(
        thickness: 0.7,
        color: Colors.grey.withOpacity(0.5),
      ),
    ),
  ],
),
const SizedBox(
  height: 30.0,
),
// Sign up social media logo
Row(
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,
  children: [
    Logo(Logos.facebook_f),
    Logo(Logos.twitter),
    Logo(Logos.google),
    Logo(Logos.apple),
  ],

```

```

    ),
    const SizedBox(
      height: 25.0,
    ),
    // Already have an account
    Row(
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        const Text(
          'Already have an account? ',
          style: TextStyle(
            color: Colors.black45,
          ),
        ),
        GestureDetector(
          onTap: () {
            Navigator.push(
              context,
              MaterialPageRoute(
                builder: (e) => const SignInScreen(),
              ),
            );
          },
          child: Text(
            'Sign in',
            style: TextStyle(
              fontWeight: FontWeight.bold,
              color: lightColorScheme.primary,
            ),
          ),
        ),
      ],
    ),
    const SizedBox(
      height: 20.0,
    ),
  ],
),
),
),
),
);
}
}

```

menu_page.dart

```
import 'dart:convert';
import 'package:flutter/material.dart';
import 'package:login_signup/pages/cart_manager.dart';
import 'package:login_signup/widgets/custom_scaffold.dart';
import 'package:login_signup/widgets/search_widget.dart';
import 'cart_page1.dart';

class MenuPage extends StatefulWidget {
  final String restaurantName;

  const MenuPage({super.key, required this.restaurantName});

  @override
  _MenuPageState createState() => _MenuPageState();
}

class _MenuPageState extends State<MenuPage> {
  List<Map<String, dynamic>> _menuItems = [];
  List<Map<String, dynamic>> _filteredMenuItems = [];

  Future<List<Map<String, dynamic>>> loadMenuItems(BuildContext context) async {
    try {
      String dataString = await DefaultAssetBundle.of(context).loadString('assets/data1.json');
      List<dynamic> jsonList = jsonDecode(dataString);

      Map<String, dynamic>? restaurantData = jsonList.firstWhere(
        (element) => element['restName'] == widget.restaurantName,
        orElse: () => {},
      );

      if (restaurantData!.isEmpty) {
        throw Exception('Restaurant not found: ${widget.restaurantName}');
      }

      return List<Map<String, dynamic>>.from(restaurantData['menuItems']);
    } catch (e) {
      throw Exception('Error loading menu: $e');
    }
  }

  void addToCart(Map<String, dynamic> item) {
    CartManager().addItem(item, widget.restaurantName);
  }
}
```

```

ScaffoldMessenger.of(context).showSnackBar(
  SnackBar(
    content: Text('${item['Name']} added to cart!'),
    backgroundColor: Colors.green.shade700,
    duration: const Duration(seconds: 2),
  ),
);
}

void _searchMenuItems(String query) {
  final filteredItems = _menuItems.where((item) {
    return item['Name'].toLowerCase().contains(query.toLowerCase());
  }).toList();

  setState(() {
    _filteredMenuItems = filteredItems;
  });
}

@override
Widget build(BuildContext context) {
  return CustomScaffold(
    title: widget.restaurantName,
    drawer: Drawer(
      child: ListView(
        padding: EdgeInsets.zero,
        children: [
          DrawerHeader(
            decoration: BoxDecoration(
              color: Colors.orange.shade700,
            ),
            child: const Text(
              'Menu Options',
              style: TextStyle(
                color: Colors.white,
                fontSize: 24,
              ),
            ),
          ),
          ListTile(
            title: const Text('Item 1'),
            onTap: () {
              // Handle navigation or other actions
            },
          ),
        ],
      ),
    ),
  );
}

```

```

        ListTile(
          title: const Text('Item 2'),
          onTap: () {
            // Handle navigation or other actions
          },
        ),
      ],
    ),
  ),
  child: Stack(
    children: [
      Column(
        children: [
          const Padding(
            padding: EdgeInsets.all(8.0),
            child: SearchWidget(
              // onChanged: _searchMenuItems, // Uncomment if connected to SearchWidget
            ),
          ),
          Expanded(
            child: FutureBuilder<List<Map<String, dynamic>>>(
              future: loadMenuItems(context),
              builder: (context, snapshot) {
                if (snapshot.connectionState == ConnectionState.waiting) {
                  return const Center(child: CircularProgressIndicator());
                } else if (snapshot.hasError) {
                  return const Center(child: Text('Error loading menu'));
                } else if (!snapshot.hasData || snapshot.data!.isEmpty) {
                  return const Center(child: Text('No menu items available'));
                }

                if (_menuItems.isEmpty) {
                  _menuItems = snapshot.data!;
                  _filteredMenuItems = _menuItems;
                }

                return GridView.builder(
                  gridDelegate: const SliverGridDelegateWithFixedCrossAxisCount(
                    crossAxisCount: 2,
                    mainAxisSpacing: 12,
                    crossAxisSpacing: 12,
                    childAspectRatio: 0.8,
                  ),
                  itemCount: _filteredMenuItems.length,
                  itemBuilder: (context, index) {

```

```

return Card(
  elevation: 8,
  shape: RoundedRectangleBorder(
    borderRadius: BorderRadius.circular(15),
  ),
  child: Stack(
    alignment: Alignment.bottomCenter,
    children: [
      ClipRRect(
        borderRadius: BorderRadius.circular(15),
        child: Image.asset(
          _filteredMenuItems[index]['Image'],
          fit: BoxFit.cover,
          width: double.infinity,
          height: double.infinity,
        ),
      ),
      Container(
        padding: const EdgeInsets.all(8),
        color: Colors.black.withOpacity(0.6),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.min,
          children: [
            Text(
              _filteredMenuItems[index]['Name'],
              style: const TextStyle(
                fontSize: 16,
                fontWeight: FontWeight.bold,
                color: Colors.white,
              ),
            ),
            const SizedBox(height: 4),
            Text(
              '₹${_filteredMenuItems[index]['price']}',
              style: const TextStyle(
                fontSize: 14,
                color: Colors.white,
              ),
            ),
            const SizedBox(height: 4),
            ElevatedButton(
              style: ElevatedButton.styleFrom(
                backgroundColor: Colors.orange.shade700,
              ),
              onPressed: () =>

```

```

        addToCart(_filteredMenuItems[index]),
        child: const Text('Add to Cart'),
      ),
    ],
  ),
),
],
),
);
},
);
},
),
],
),
Positioned(
  bottom: 20,
  right: 20,
  child: FloatingActionButton(
    backgroundColor: Colors.orange.shade700,
    onPressed: () {
      Navigator.push(
        context,
        MaterialPageRoute(
          builder: (context) => CartPage(
            cartItems: CartManager().cartItems,
          ),
        ),
      );
    },
    child: const Icon(Icons.shopping_cart),
  ),
),
],
),
);
}
}

```

cart_manager.dart

```
class CartManager {
  static final CartManager _instance = CartManager._internal();

  // Singleton pattern
  factory CartManager() => _instance;

  CartManager._internal();

  final List<Map<String, dynamic>> _cartItems = [];

  List<Map<String, dynamic>> get cartItems => List.unmodifiable(_cartItems);

  void addItem(Map<String, dynamic> item, String restaurantName) {
    var existingItem = _cartItems.firstWhere(
      (cartItem) => cartItem['Name'] == item['Name'] && cartItem['restaurantName'] ==
restaurantName,
      orElse: () => {},
    );

    if (existingItem.isEmpty) {
      _cartItems.add({
        'Name': item['Name'],
        'Image': item['Image'],
        'price': item['price'],
        'quantity': 1,
        'restaurantName': restaurantName,
      });
    } else {
      existingItem['quantity']++;
    }
  }

  void removeItem(String name, String restaurantName) {
    _cartItems.removeWhere((item) => item['Name'] == name && item['restaurantName'] ==
restaurantName);
  }

  void clearCart() {
    _cartItems.clear();
  }

  Map<String, List<Map<String, dynamic>>> groupItemsByRestaurant() {
    Map<String, List<Map<String, dynamic>>> groupedItems = {};
  }
}
```



```

    for (var item in _cartItems) {
      if (groupedItems.containsKey(item['restaurantName'])) {
        groupedItems[item['restaurantName']]!.add(item);
      } else {
        groupedItems[item['restaurantName']] = [item];
      }
    }
    return groupedItems;
  }
}

```

search_widget.dart

```

import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';

class SearchWidget extends StatelessWidget {
  const SearchWidget({super.key});

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.symmetric(
        vertical: 10,
        horizontal: 15,
      ),
      child: Container(
        width: double.infinity,
        height: 50,
        decoration: BoxDecoration(
          color: Colors.white,
          borderRadius: BorderRadius.circular(20),
          boxShadow: [
            BoxShadow(
              color: Colors.grey.withOpacity(0.5),
              spreadRadius: 2,
              blurRadius: 10,
              offset: const Offset(0,3),
            ),
          ],
        ),
        child: Padding(
          padding: const EdgeInsets.symmetric(

```

```

        horizontal: 10,
      ),
      child: Row(children: [
        const Icon(
          CupertinoIcons.search,
          color: Colors.orange
        ),
        SizedBox(
          height: 50,
          width: 300,
          child: Padding(padding: const EdgeInsets.symmetric(
            horizontal: 15,
          ),
            child: TextFormField(
              decoration: const InputDecoration(
                hintText: "What would you like to have?",
                border: InputBorder.none,
              ),
            ),
          ),
        ),
        const Spacer(),
        const Icon(Icons.filter_list),
      ],
    ),
  ),
);
}}

```

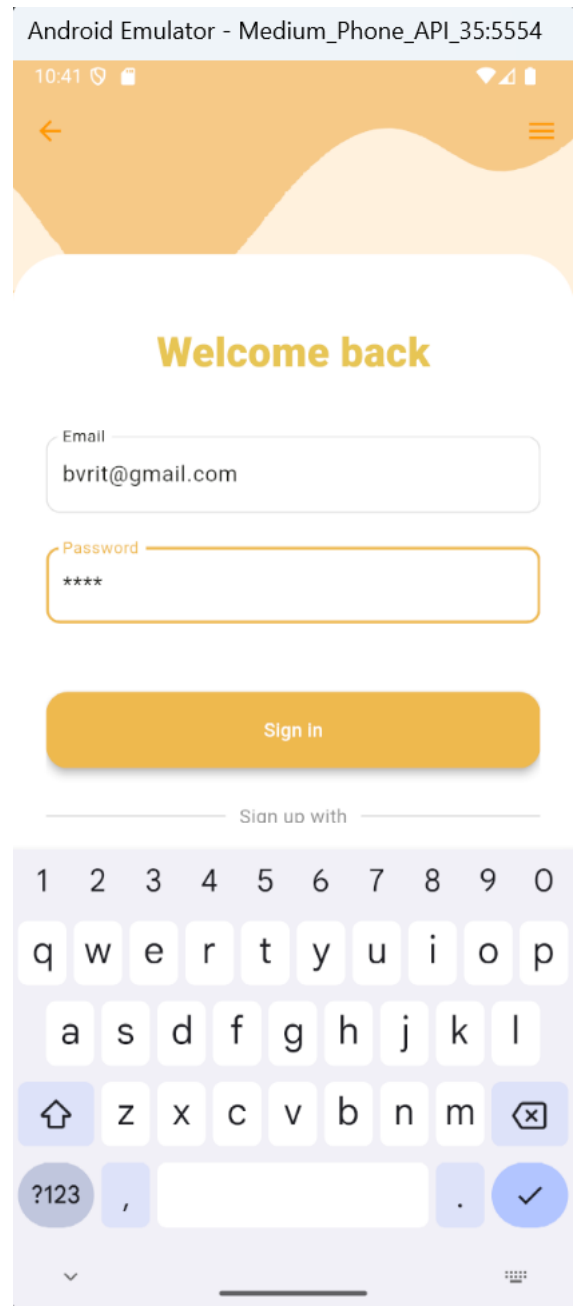
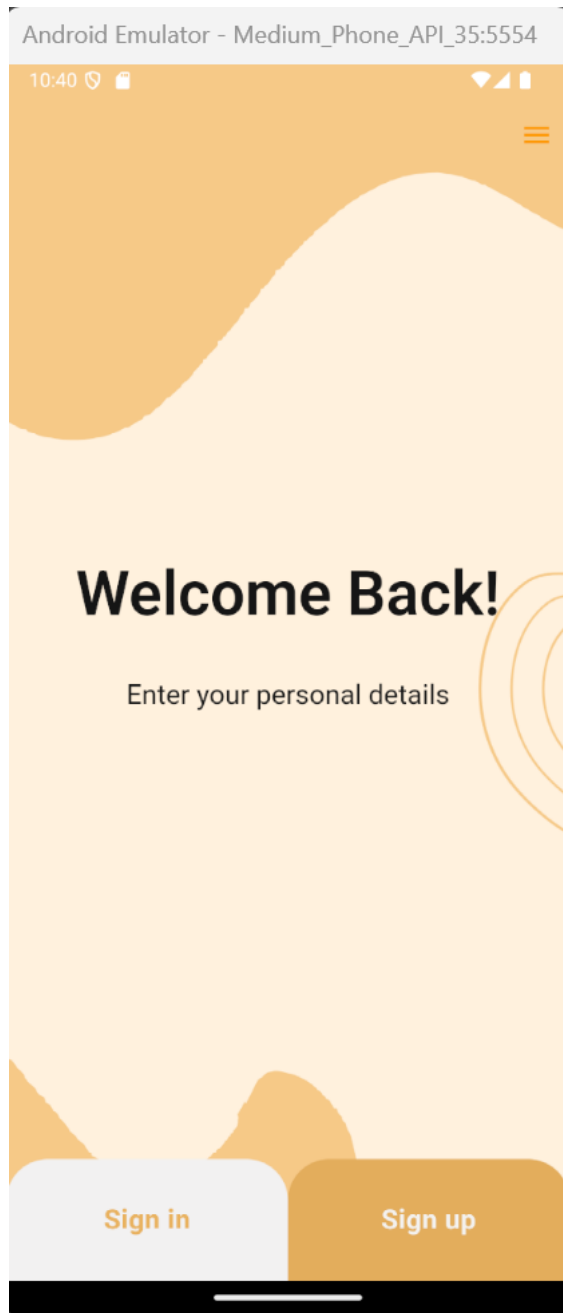
welcome_button.dart

```
import 'package:flutter/material.dart';

class WelcomeButton extends StatelessWidget {
  const WelcomeButton(
    {super.key, this.buttonText, this.onTap, this.color, this.textColor});
  final String? buttonText;
  final Widget? onTap;
  final Color? color;
  final Color? textColor;

  @override
  Widget build(BuildContext context) {
    return GestureDetector(
      onTap: () {
        Navigator.push(
          context,
          MaterialPageRoute(
            builder: (e) => onTap!,
          ),
        );
      },
      child: Container(
        padding: const EdgeInsets.all(30.0),
        decoration: BoxDecoration(
          color: color!,
          borderRadius: const BorderRadius.only(
            topLeft: Radius.circular(30),
            topRight: Radius.circular(30),
          ),
        ),
        child: Text(
          buttonText!,
          textAlign: TextAlign.center,
          style: TextStyle(
            fontSize: 20.0,
            fontWeight: FontWeight.bold,
            color: textColor!,
          ),
        ),
      ),
    );
  }
}
```

OUTPUT



10:43

Home

CampusBite



Fast Food



Tiffins



Juice



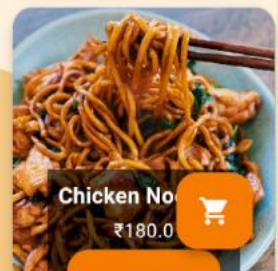
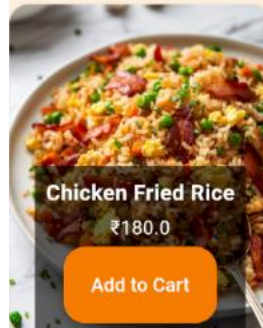
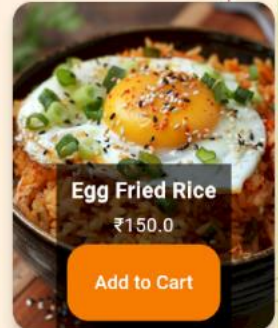
PaniPuri & Chats

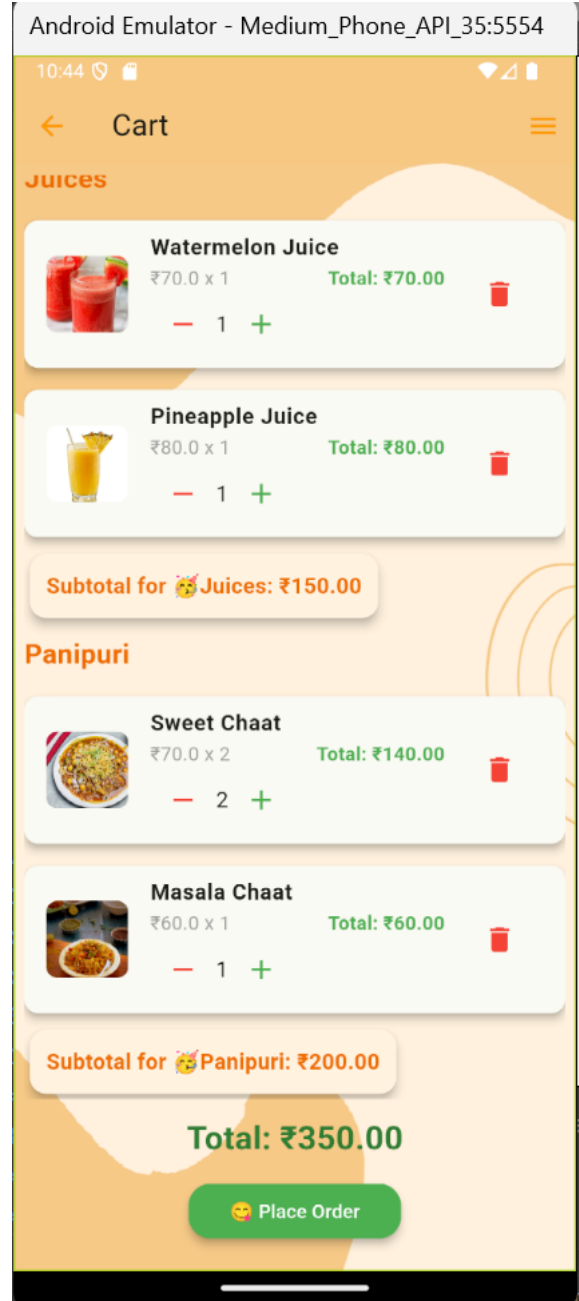
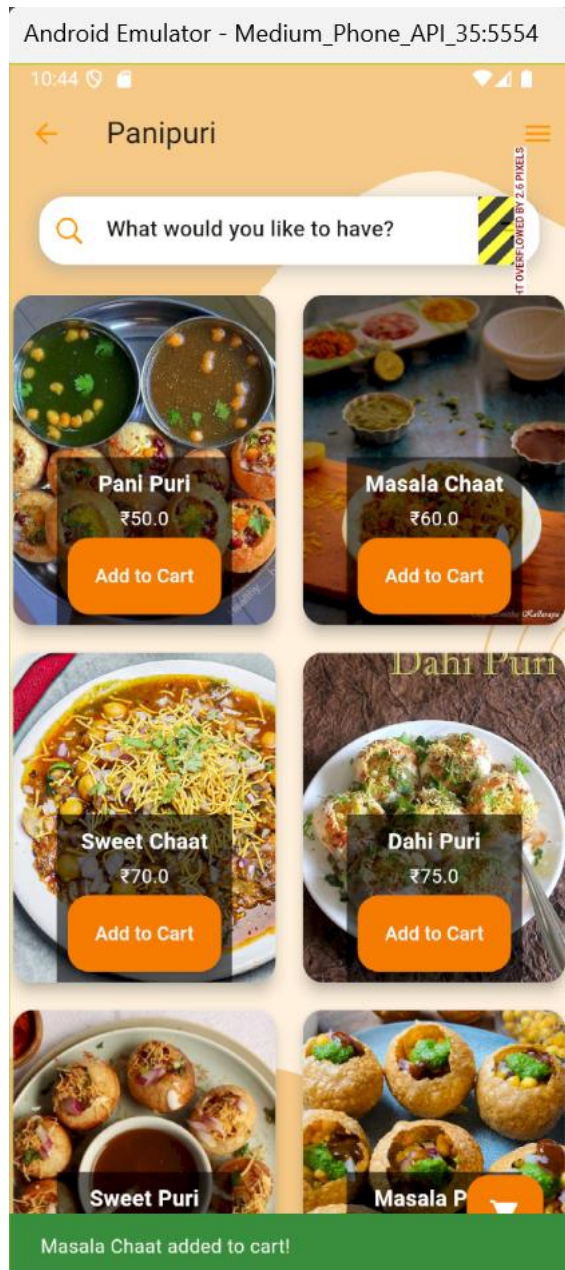
10:43

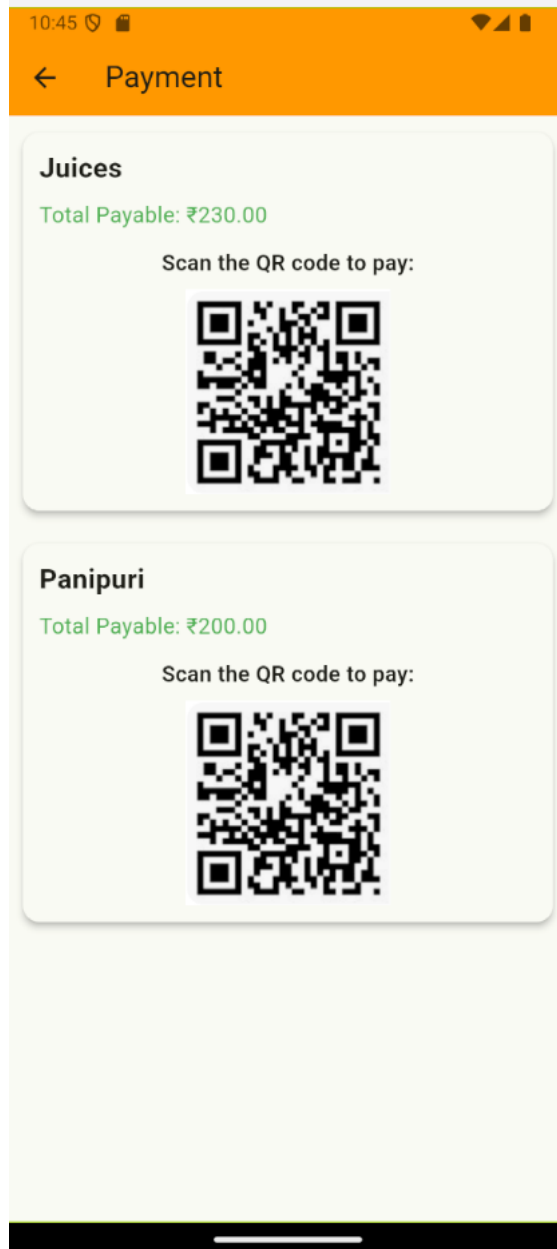
Fastfood



What would you like to have?







GITHUB Link:

<https://github.com/31Rishita/Campus-Bites>