

Mobile Application

Group 5

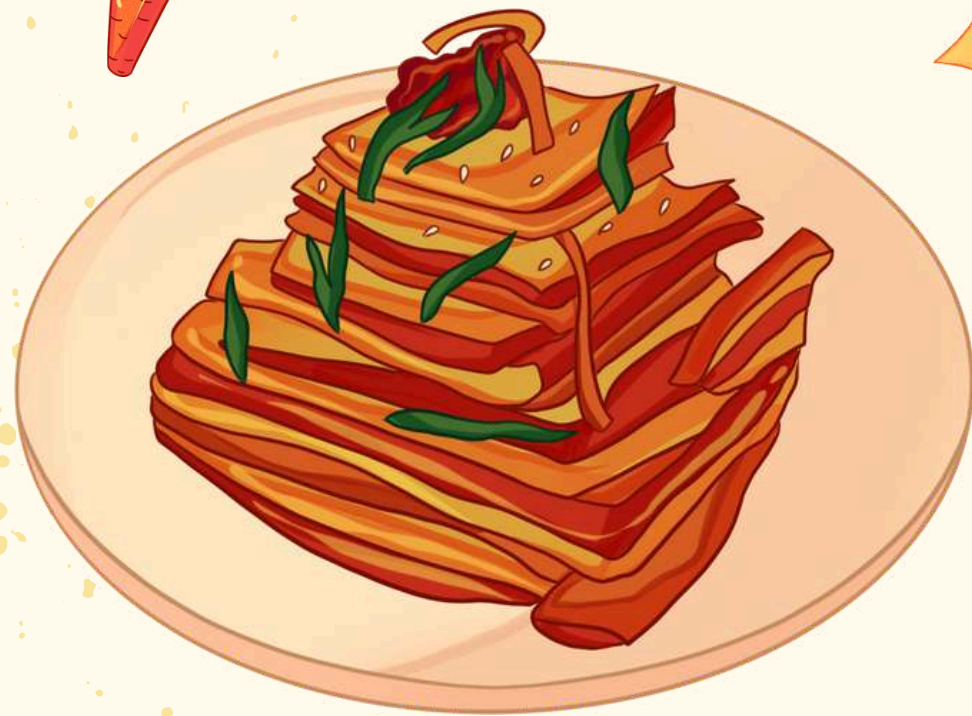
Present By:

Caleb Lau Lik Hung SWE2209841

Yong Zhi Shen SWE2209847

Xu Jingxiang SWE2209543

Li linyi DMT2209210



Project Overview



What is our app?

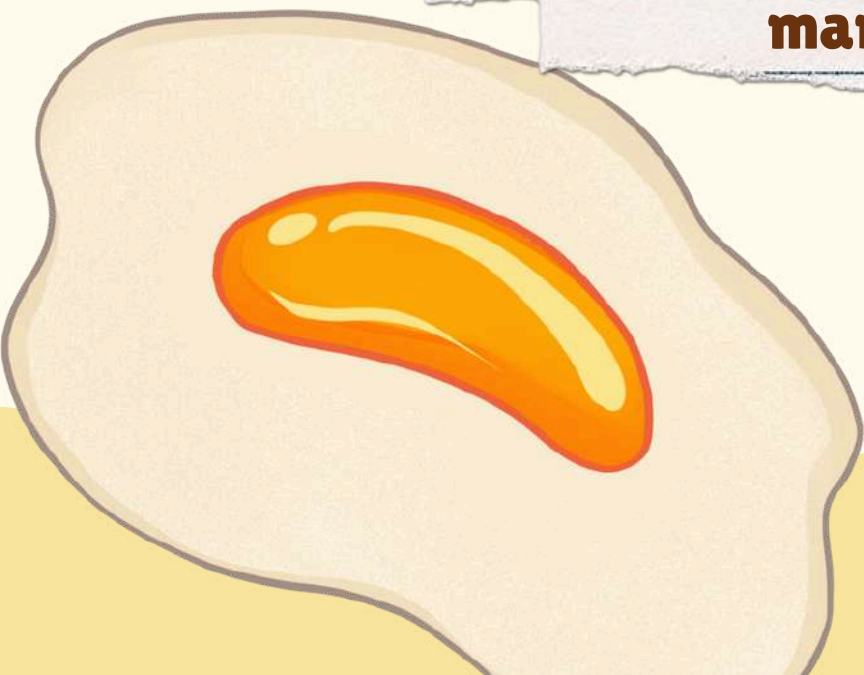
- **Application Name :Food Lover**
- **a food ordering mobile app**

What users can do?

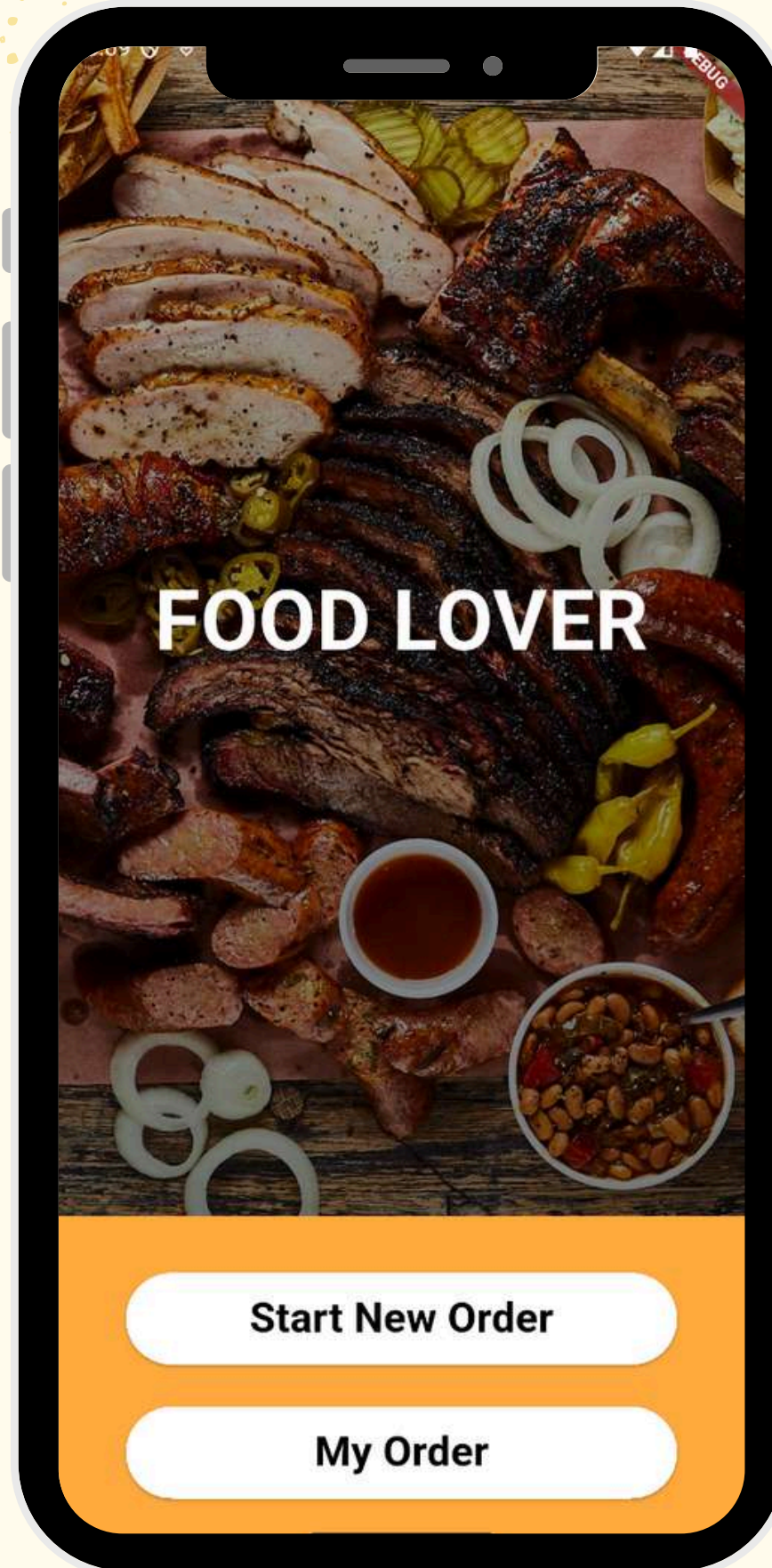
- **browse menu, order food, manage cart, get discounts, etc.**

The overall goal

- **To provide a seamless, efficient, and enjoyable food ordering experience for all users.**



Start Page



Start Page:

Attractive background, app name, and two main actions:

- Start New Order (login/register)
- My Order (view cart)

Technical Highlights:

- Used Stack and Column widgets to layer background and content.
- Navigation implemented with `Navigator.push` to Login/Register or Cart.



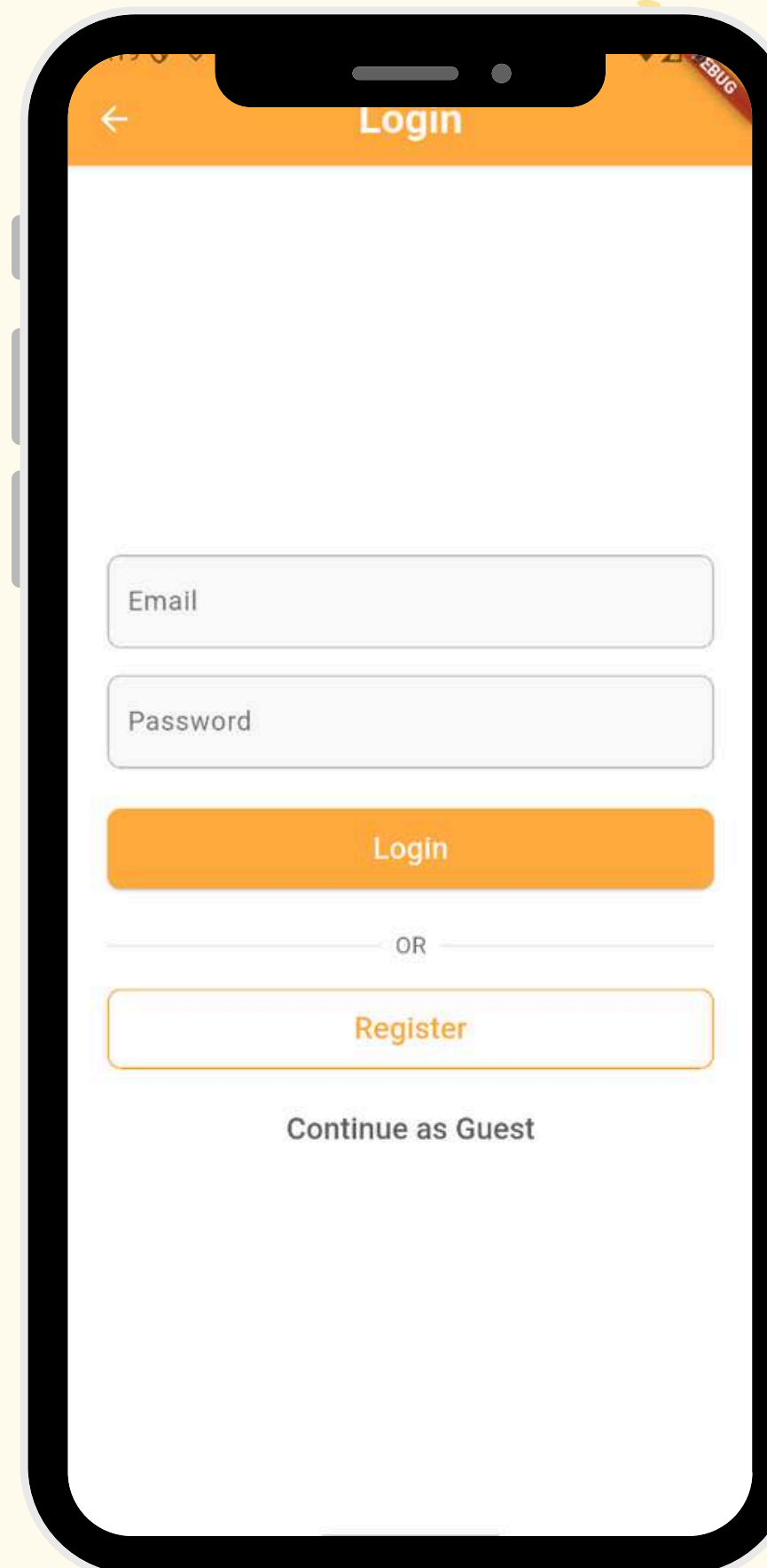
Login & Register Page

Login/Register Page:

- Login or registration
- Option to continue as guest.

Technical Highlights:

- Authentication handled with Firebase Auth (login/register).
- Error and success feedback via Snackbar.
- Firestore integration to store new user data on registration.



← Login

Email

Password

Login

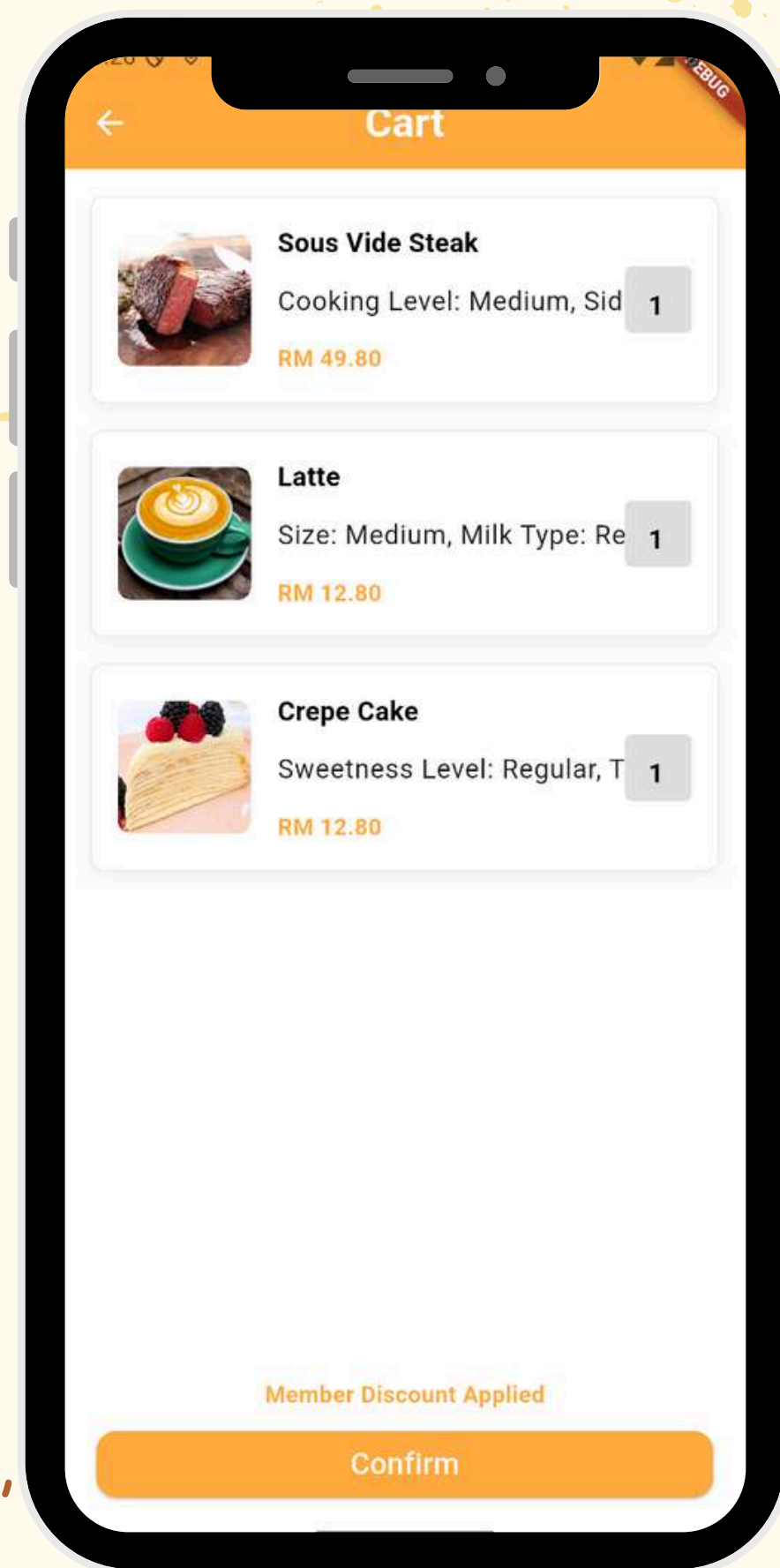
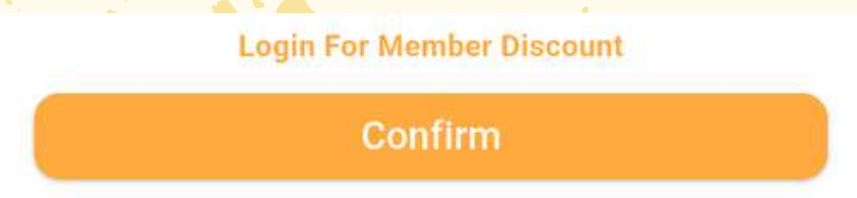
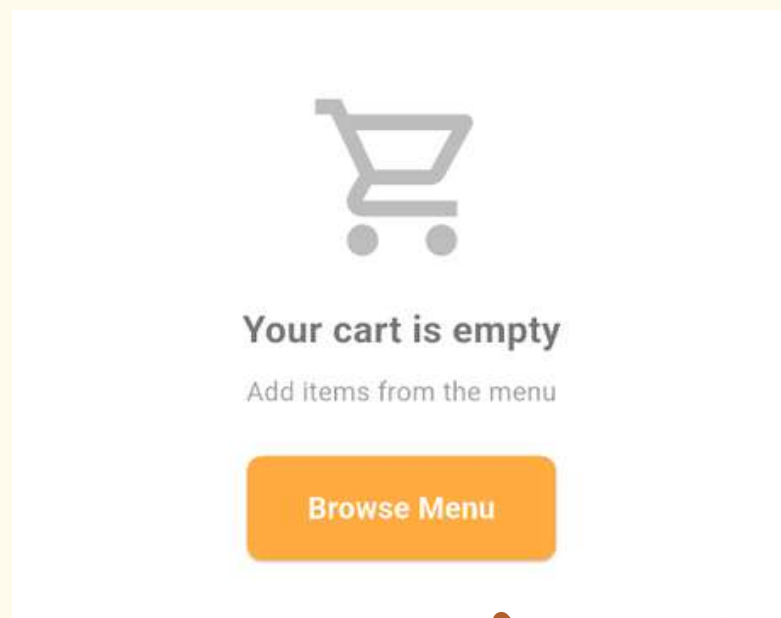
OR

Register

Continue as Guest



Cart Page



Cart Page:

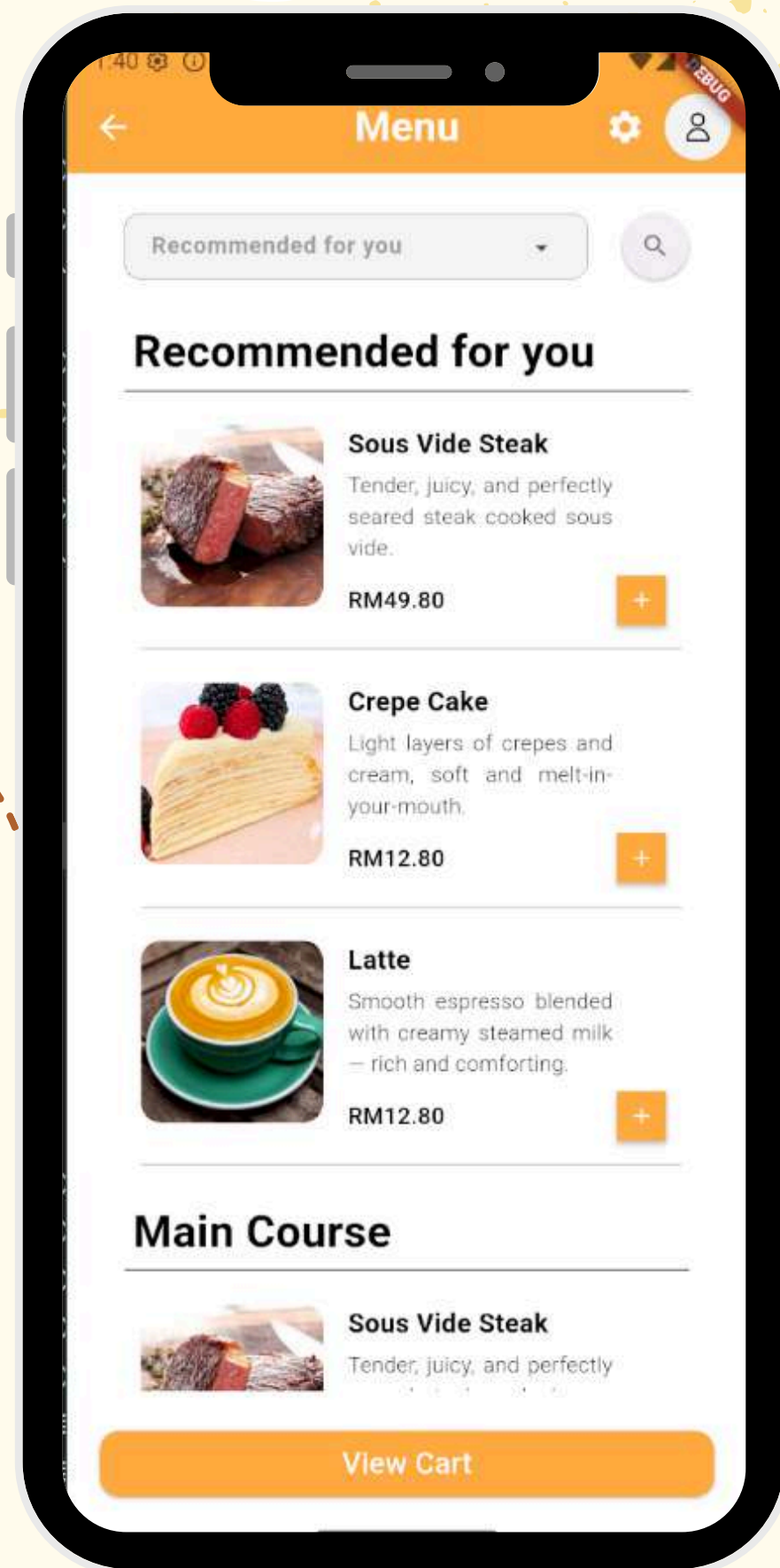
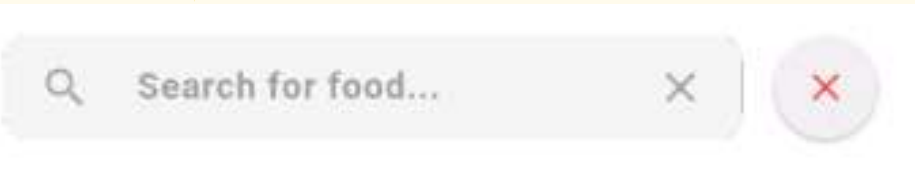
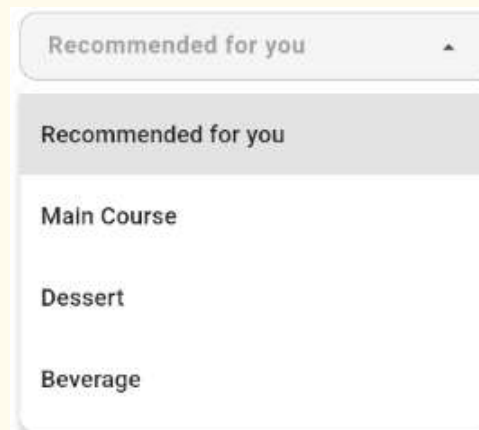
- **View, edit, and remove items.**
- **Add remarks and adjust quantity.**
- **See member discount status and total price.**

Technical Highlights:

- **Used ListView.builder for dynamic cart item display.**
- **Real-time updates to cart using singleton CartManager.**
- **Member discount message shown based on login status (DiscountManager).**
- **Navigation to payment page with cart data passed as arguments.**



Menu Page



Menu Page:

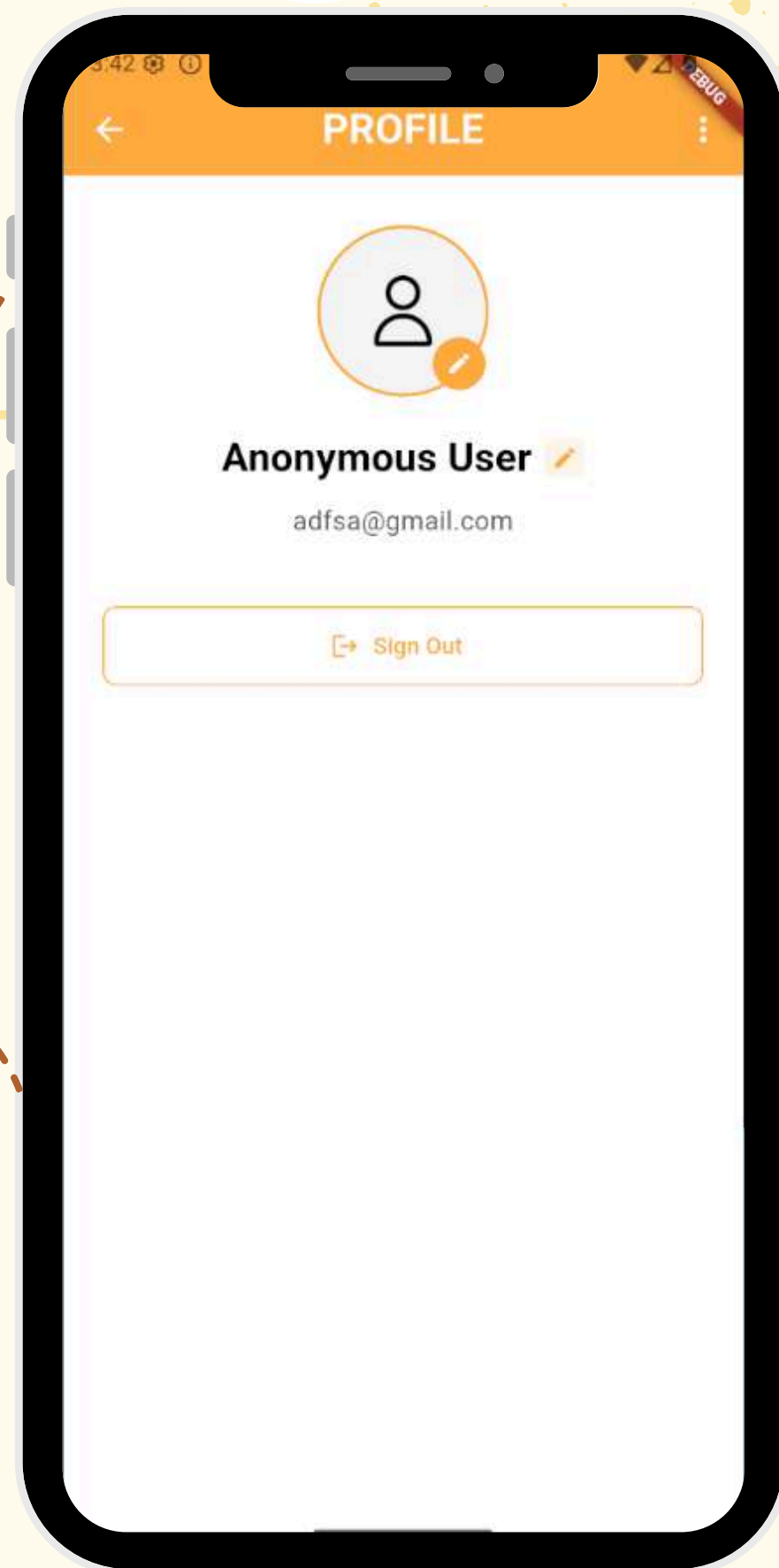
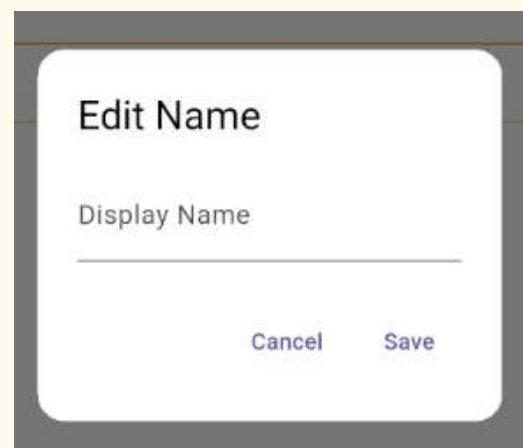
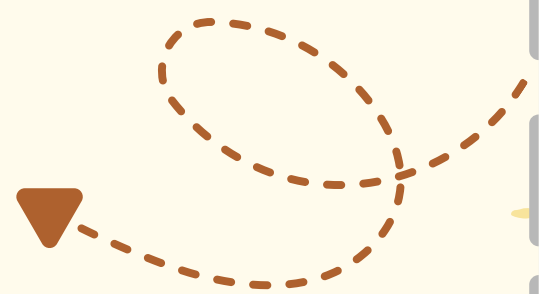
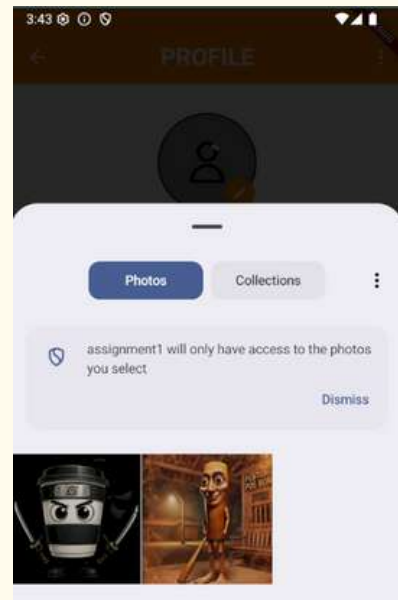
- Access to food detail page while pressing the add button.
- Add Profile Icon Button while logged in.
- Scroll to the section after pressing the options inside the dropdown menu.
- Enable user to search the food.

Technical Highlights:

- Used `ListView.builder` for dynamic menu item display.
- Use `StreamBuilder<User?>` to check user login with `if (user != null)` to show the profile icon button.
- Use `GlobalKey` and `BuildContext?` to scroll to section



Profile Page



Profile Page:

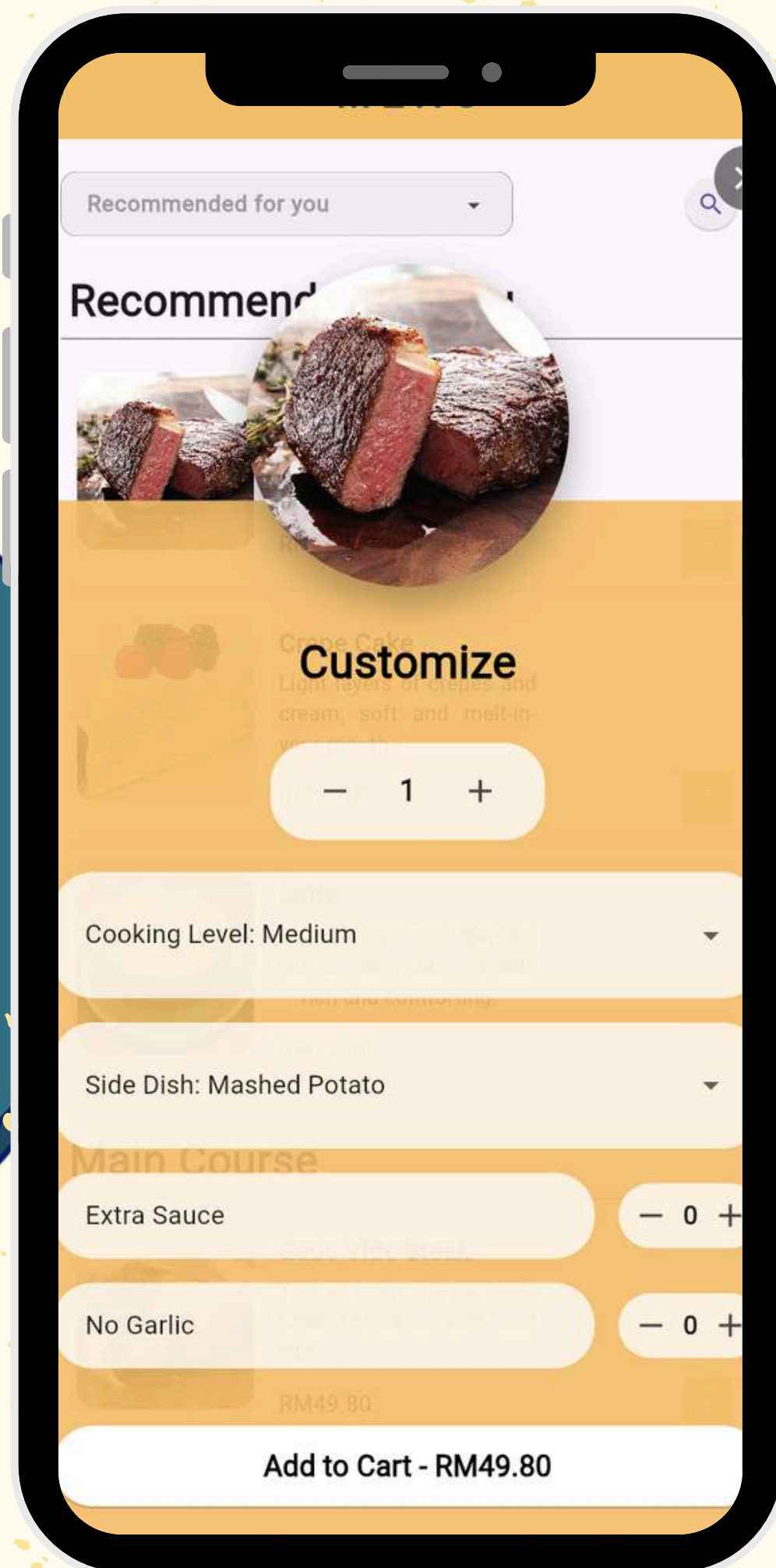
- Allow user to sign out.
- Allow user to edit his profile picture and username.
- Allow user to choose image from his phone gallery.

Technical Highlights:

- Used Cloudinary for storing user profile image.
- Used Firebase to read and update username and email.
- Used `FirebaseAuth.instance.signOut()` to sign out user.
- Used Image picker to choose image.
- Used Snack bar to show error messages.



Food detail Page



Display Food Details
Food Customization Options
Quantity Selection
Add to Cart



Technical Highlights :

Hero Animation Transition

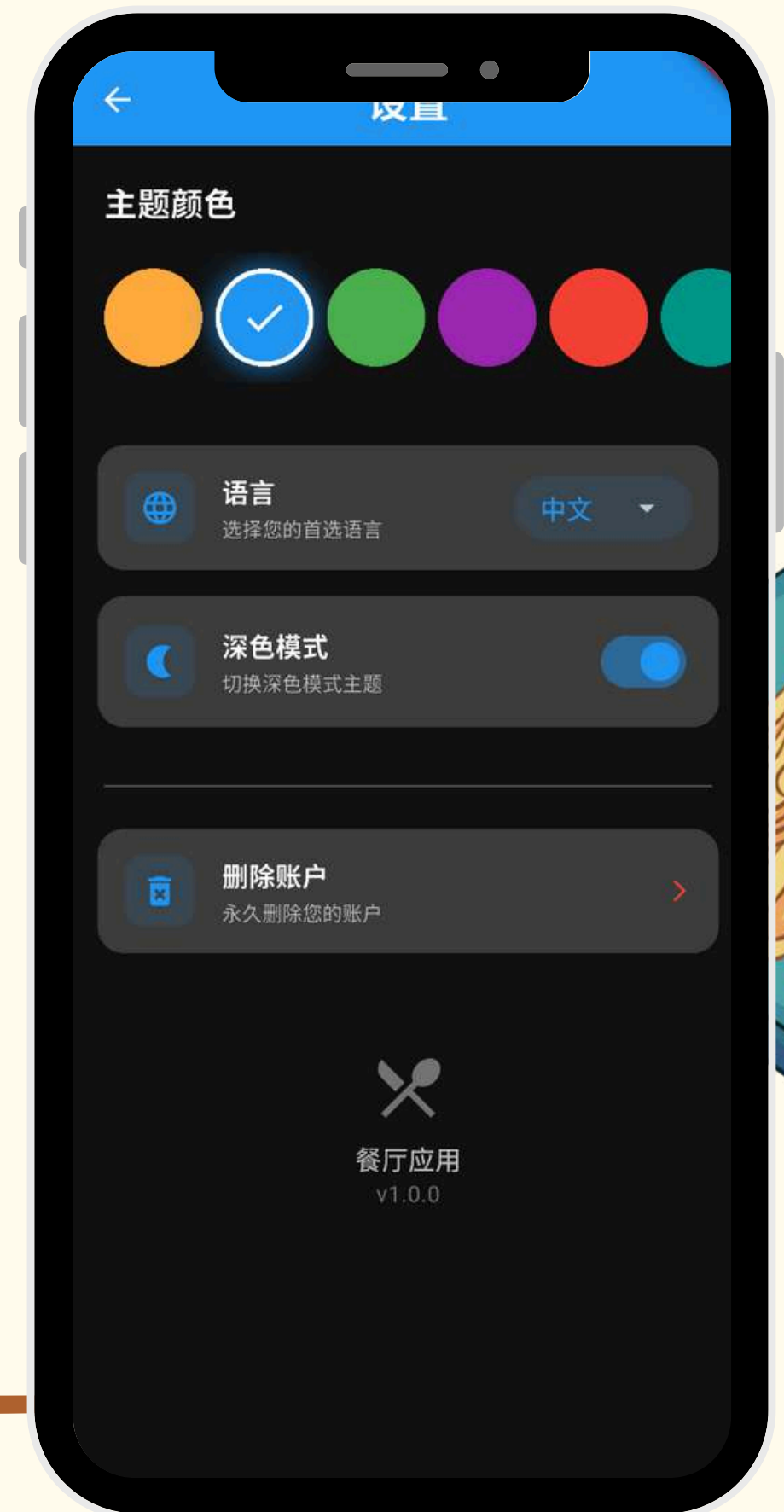
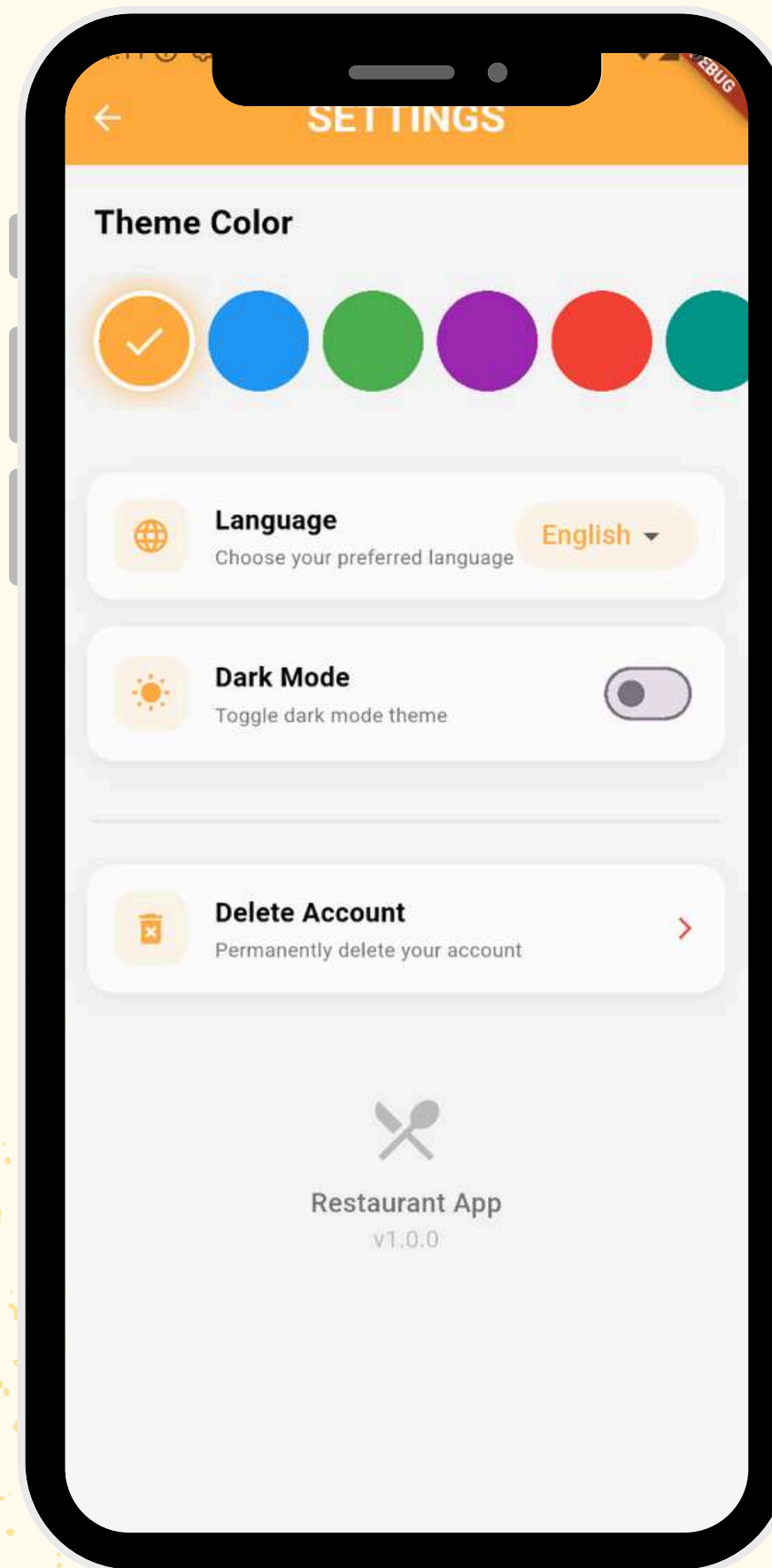
Multi-layer Animation Orchestration : Three **AnimationControllers** control main animation, menu slide-in, and options fade-in sequentially

Tween<double> animates border radius from 12 to 100 (rectangle to circle transformation)

Settings Page

Settings Options

- **Theme Colors:** Interactive color selection wheel
- **Language:** Dropdown (English/Chinese)
- **Dark Mode:** Toggle switch
- **Delete Account:** Red warning option (for logged-in users)





Settings Page

Global Theme System – Real-time color switching

```
// Set theme color
Future<void> setThemeColor(Color color) async {
  _themeColor = color;
  notifyListeners();

  final prefs = await SharedPreferences.getInstance();
  final colorIndex = availableColors.indexOf(color);
  await prefs.setInt(_themeColorKey, colorIndex);
}
```

Firebase Account Management – Secure account deletion

```
if (shouldDelete == true) {
  try {
    final User? currentUser = FirebaseAuth.in
    if (currentUser != null) {
      await currentUser.delete();
    }
  }
}
```

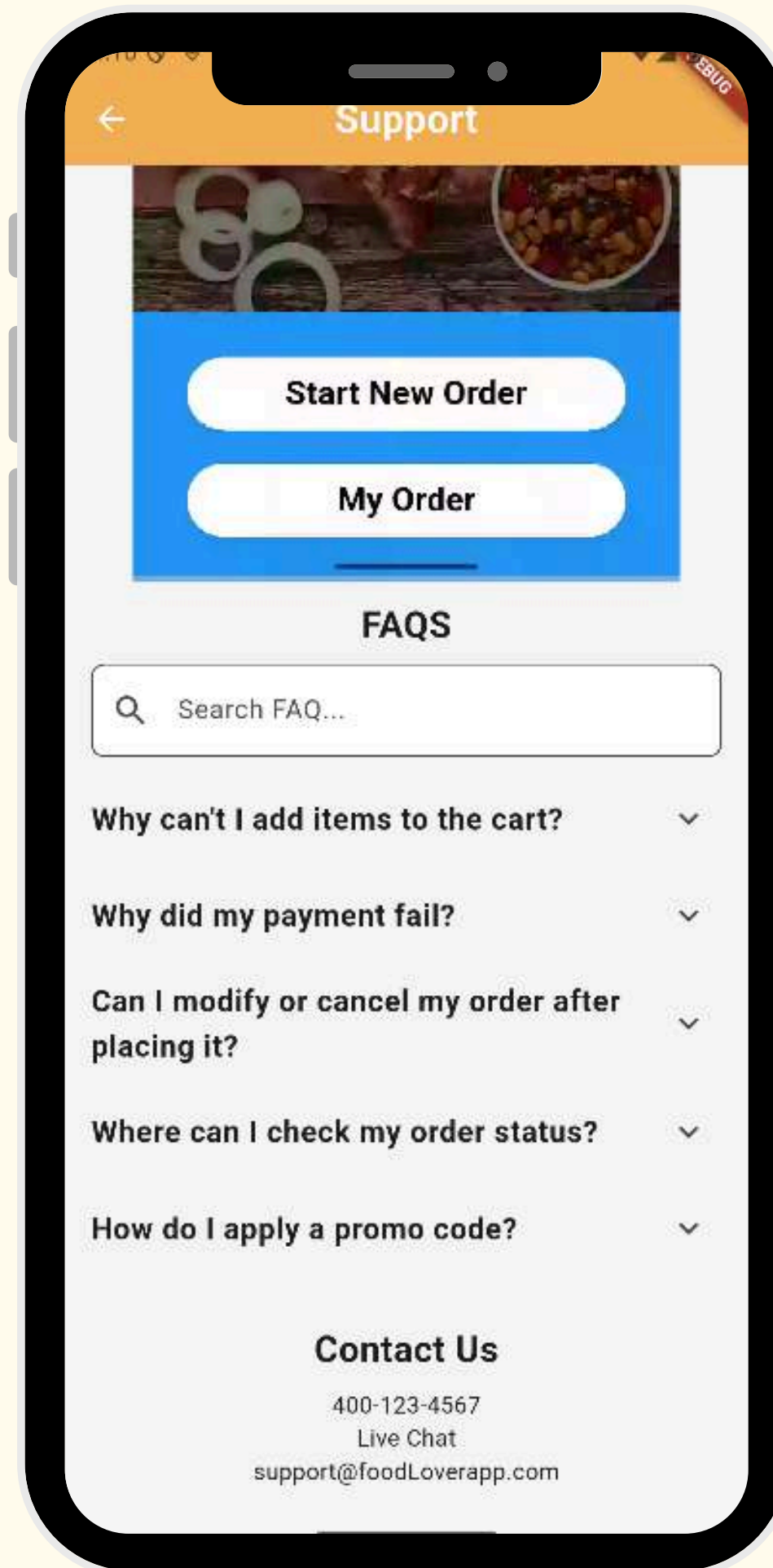
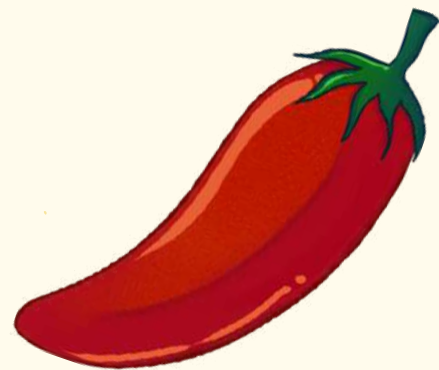
Multi-language Support – Dynamic translation system

```
// Get translated text
String translate(String key) {
  if (_language == 'English') {
    return _englishTexts[key] ?? key;
  } else {
    return _chineseTexts[key] ?? key;
  }
}
```

>Support & Help Page

Support & Help Page

- **Quick Guide Video**
- **FAQS** answer the basic questions
- **Contact part**



Technical Highlights

- **Video_player**
- **Using ListView** ensures that the page can be scrolled up and down.
- **Using ExpansionPanelList** shows hidden answer content
- **Search box**



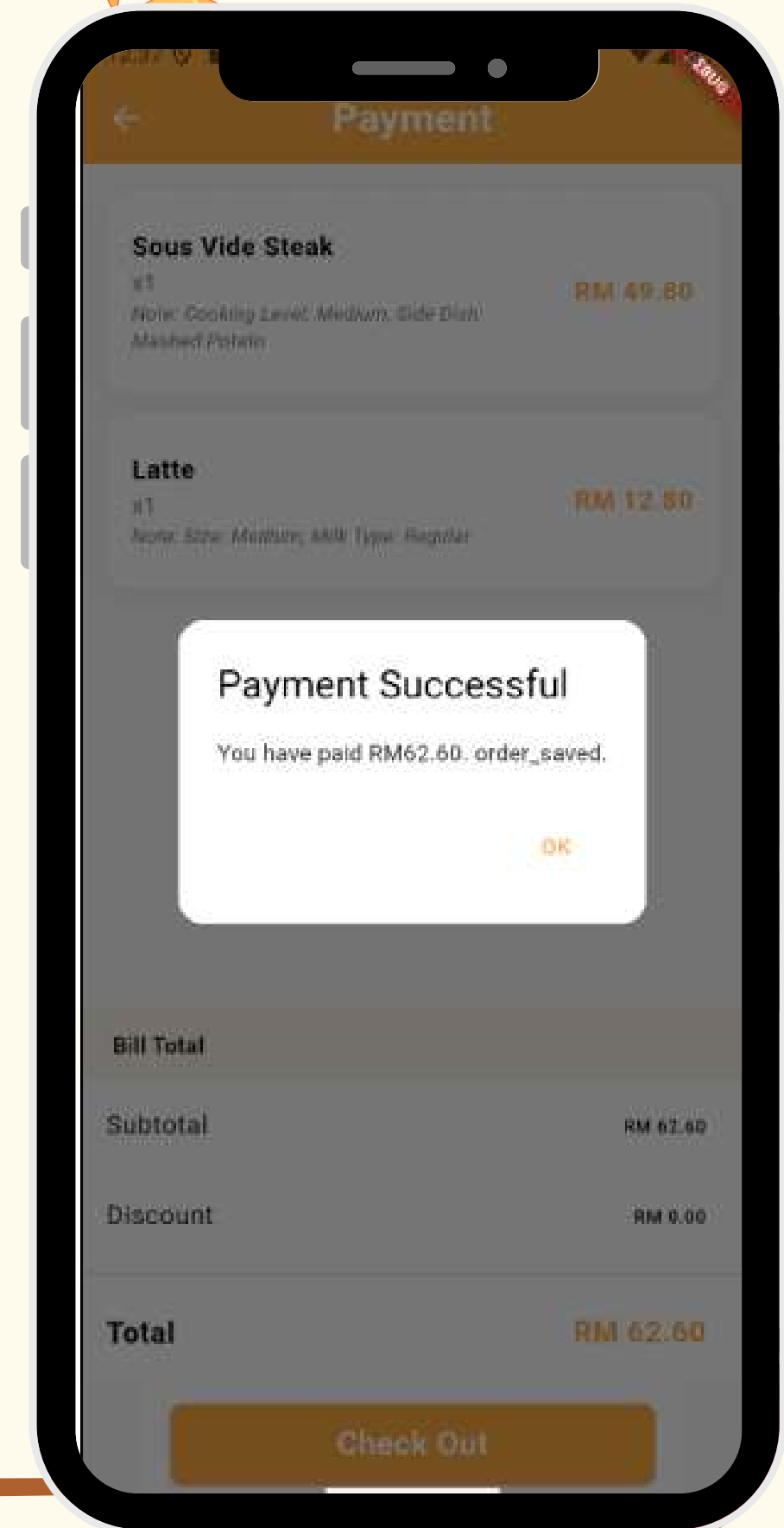
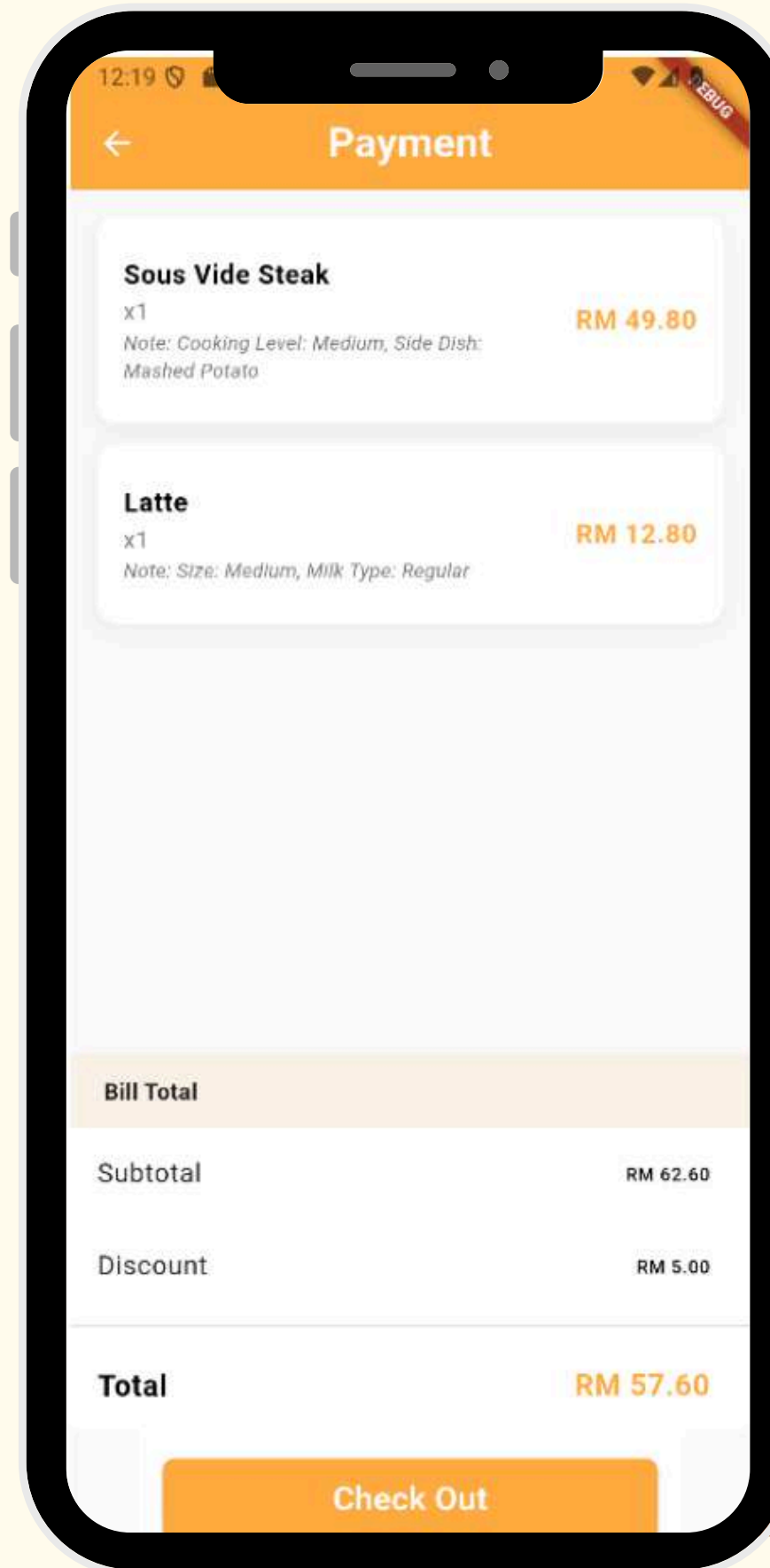
Payment Page

Payment Page

- Display specific products, along with their special notes and quantities.
- After successful payment, a page prompt will appear to help customers confirm.

Technical Highlights:

- Use `AlertDialog` + if (`paymentSuccess`) display the payment successful page
- Use `'List.reduce'` or `'for'` to accumulate logic to calculate total price.
- Use state management to control discount logic



Reflection



- **Should use version control**
- **Learned the importance of design before implementation**
- **Gain knowledge while implementing Firebase, Firestore, etc.**
- **Deadline is important for projects.**

This food ordering application can be better by implementing more features:

- **Like and comment section**
- **Better reward system**
- **User feedback section**



Team Contribution

SWE2209841	Design and implement Main Menu and Profile Page
SWE2209847	Design and implement Start Page, Login and Register and Cart Page
SWE2209543	Design and implement Support Page and Payment Page
DMT2209210	Design and implement Setting page and Food Detail Page

**Thank you for
your attention**

