

Bag It

Group Members:

Sabahat Tufail

Registration No: 23PWCSE2333

Ahlam Orakzai

Registration No: 23PWCSE2236

SUPERVISED BY :

Maam Summaeya Salahuddin

Tools and Technologies

- **Programming Language:** Dart
- **Framework:** Flutter
- **State Management:** Provider
- **IDE:** Visual Studio Code
- **Version Control:** GitHub
- **UI Libraries:** Google Fonts, Curved Navigation Bar

Objectives for the Grocery Shopping App:

1. **User-Friendly Navigation:**
 - a. Implement a **nav_bar** for seamless movement between pages.
 - b. Include a **drawer** for additional navigation options.
2. **Comprehensive Product Management:**
 - a. Display grocery items using **grocery_item_tile**.
 - b. Provide detailed product descriptions in **products_detailed**.
3. **Shopping Cart & Wishlist:**
 - a. Develop a **cart_page** to manage user-selected items.
 - b. Implement a **favorite_page** to save preferred products.
4. **Authentication & Onboarding:**
 - a. Enable user registration and login with **signup** and **signin** pages.
 - b. Create an engaging first-time user experience using **onboarding**.
5. **Backend & Networking:**
 - a. Manage data retrieval and API communication using **network.dart**.
 - b. Integrate Firebase for authentication and backend storage (**firebase_options.dart**).
6. **Settings & User Preferences:**
 - a. Allow users to adjust preferences via the **settings** page.
7. **State Management & Performance Optimization:**
 - a. Utilize the **view_model** layer for efficient data handling.
 - b. Implement **utils** for common functions and performance enhancements.

Screens and Features

1. Intro Screen

```
1 import 'package:flutter/material.dart';
2 import 'package:google_fonts/google_fonts.dart';
3 import 'view/nav_bar.dart';
4
5 class IntroScreen extends StatelessWidget {
6   const IntroScreen({super.key});
7
8   @override
9   Widget build(BuildContext context) {
10     return Scaffold(
11       body: Container(
12         // Gradient background
13         decoration: const BoxDecoration(
14           gradient: LinearGradient(
15             colors: [Color(0xFFBEC5FC), Color(0xFFE0C3FC)],
16             begin: Alignment.topCenter,
17             end: Alignment.bottomCenter,
18           ), // LinearGradient
19         ), // BoxDecoration
20         child: SafeArea(
21           child: Column(
22             crossAxisAlignment: CrossAxisAlignment.center,
23             children: [
24               // Logo Image
25               Padding(
26                 padding: const EdgeInsets.symmetric(
27                   horizontal: 80.0, vertical: 80.0), // EdgeInsets.s
28               child: Image.asset(
29                 'Images/avocado.png',
30                 height: 160,
31                 fit: BoxFit.cover,
32               ), // Image.asset
33             ), // Padding
34
35             // Headline Text
36             Padding(
37               padding: const EdgeInsets.symmetric(horizontal: 20.0),
38               child: Text(
39                 'We deliver groceries at your doorstep',
40                 textAlign: TextAlign.center,
41                 style: GoogleFonts.notoSerif(
42                   fontSize: 32,
43                   fontWeight: FontWeight.bold,
44                   color: Colors.white,
45                 ),
46               ), // Text
47             ), // Padding
48
49             const SizedBox(height: 16),
50
51             // Subheading Text
52             Padding(
53               padding: const EdgeInsets.symmetric(horizontal: 40.0),
54               child: Text(
55                 'Fresh items everyday, curated for your family', //s head
56                 textAlign: TextAlign.center,
57                 style: GoogleFonts.nunito(
58                   fontSize: 16,
59                   color: Colors.white.withOpacity(0.9), //withOp
60                 ),
61               ), // Text
62             ), // Padding
63
64             const Spacer(),
65
66             // Get Started Button
67             GestureDetector(
68               onTap: () => Navigator.pushReplacement(
69                 context,
70                 MaterialPageRoute(
71                   builder: (context) {
72                     return MainPage(); // Use 'const' with the constr
73                   },
74                 ),
75             ),
```

2. Home Page:

```
lib > pages > home_page.dart > HomePageState > build
1 import 'package:flutter/material.dart';
2 import 'package:google_fonts/google_fonts.dart';
3 import 'package:provider/provider.dart';
4 import '../components/grocery_item_tile.dart';
5 import '../model/cart_model.dart';
6 import 'cart_page.dart';
7 import 'products_detailed_page.dart';
8
9 class HomePage extends StatefulWidget {
10   const HomePage({super.key});
11
12   @override
13   State<HomePage> createState() => _HomePageState();
14 }
15
16 class _HomePageState extends State<HomePage> {
17   @override
18   Widget build(BuildContext context) {
19     return Scaffold(
20       appBar: AppBar(
21         backgroundColor:
22           const Color(0xFFBEC5FC), // Light blue color for the AppBar
23         elevation: 4,
24         leading: const Padding(
25           padding: EdgeInsets.only(left: 24.0),
26           child: Icon(
27             Icons.location_on,
28             color: Colors.white, // White icon for better contrast
29           ), // Icon
30         ), // Padding
31         title: Text(
32           'Peshawar, Pakistan',
33           style: GoogleFonts.poppins(
34             fontSize: 18,
35             fontWeight: FontWeight.w500,
36             color: Colors.white, // White text for contrast
37           ),
38         ), // Text
39         centerTitle: false,
40         actions: const [
41           Padding(
42             padding: EdgeInsets.only(right: 24.0),
43             child: CircleAvatar(
44               radius: 24,
45               backgroundColor: const Color(0xFFE0C3FC), // Light purple background
46               child: Icon(
47                 Icons.person,
48                 color: Colors.white, // White icon for contrast
49               ), // Icon
50             ), // CircleAvatar
51           ), // Padding
52         ],
53       ), // AppBar
54       floatingActionButton: FloatingActionButton(
55         backgroundColor:
56           const Color(0xFFBEC5FC), // Light blue color for the FAB
57         onPressed: () => Navigator.push(
58           context,
59           MaterialPageRoute(
60             builder: (context) {
61               return const CartPage();
62             },
63           ), // MaterialPageRoute
64         ),
65         child: const Icon(
66           Icons.shopping_bag,
67           color: Colors.white, // White icon for the FAB
68         ), // Icon
69       ), // FloatingActionButton
70       body: Container(
71         decoration: const BoxDecoration(
72           gradient: LinearGradient(
73             colors: [
```

Favorites

```
products_detailed_page.dart  intro_screen.dart  cart_page.dart  main.dart  favor
lib > pages > favorite_page.dart > FavoritePage > build
1  import 'package:flutter/material.dart';
2  import 'package:provider/provider.dart';
3  import '../model/favorite_model.dart';
4
5  class FavoritePage extends StatelessWidget {
6    const FavoritePage({super.key});
7
8    @override
9    Widget build(BuildContext context) {
10      return Scaffold(
11        appBar: AppBar(
12          title: const Text('Favorites'),
13        ), // AppBar
14        body: Consumer<FavoriteModel>(
15          builder: (context, favoriteModel, child) {
16            return favoriteModel.favoriteItems.isEmpty
17              ? const Center(child: Text("No favorites added yet.))
18              : ListView.builder(
19                itemCount: favoriteModel.favoriteItems.length,
20                itemBuilder: (context, index) {
21                  final item = favoriteModel.favoriteItems[index];
22                  return ListTile(
23                    title: Text(item['name']),
24                    subtitle: Text(item['price']),
25                    trailing: IconButton(
26                      icon: const Icon(Icons.delete),
27                      onPressed: () {
28                        // Remove item from favorites
29                        favoriteModel.removeItemFromFavorite(item['name']);
30                      },
31                    ), // IconButton
32                  ); // ListTile
33                },
34              ); // ListView.builder
35            },
36          ), // Consumer
37        ); // Scaffold
38      }
39    }
40
```

O

[illegible]

Settings Page:

- Allows personalization and theme switching.
- Leave space for code and screenshots.

```
lib > pages > settings.dart > MySettingsPage > _buildProfileSection
1  import 'package:flutter/material.dart';
2
3  import '../global.dart';
4
5  class MySettingsPage extends StatelessWidget {
6    const MySettingsPage({super.key});
7
8    @override
9    Widget build(BuildContext context) {
10     return Scaffold(
11       body: Padding(
12         padding: const EdgeInsets.all(16.0),
13         child: ListView(
14           children: [
15             // Profile Section
16             _buildProfileSection(),
17             const SizedBox(height: 20),
18             // Settings Options
19             _buildSettingsOption('Account', Icons.person),
20             _buildSettingsOption('Notifications', Icons.notifications),
21             _buildSettingsOption('Privacy', Icons.lock),
22             _buildSettingsOption('Help & Support', Icons.help),
23             _buildSettingsOption('About', Icons.info),
24             const SizedBox(height: 20),
25             // Logout Button
26             ElevatedButton(
27               onPressed: () {
28                 // Handle logout
29               },
30               style: ElevatedButton.styleFrom(
31                 backgroundColor: const Color(0xffd20606),
32                 padding: const EdgeInsets.symmetric(vertical: 15),
33                 textStyle: const TextStyle(fontSize: 18),
34               ),
35               child: Text( // Use 'const' with the constructor to improve performance.
36                 'Logout',
37                 style: TextStyle(color: Colors.white), // Use 'const' with the constructor
38               ), // Text
39             ), // ElevatedButton
40           ],
41         ), // ListView
42       ), // Padding
43     ); // Scaffold
44   }
45
46   Widget _buildProfileSection() {
47     return Container(
48       padding: const EdgeInsets.all(16),
49       decoration: BoxDecoration(
50         color: Colors.white,
51         borderRadius: BorderRadius.circular(10),
52         boxShadow: [
53           BoxShadow(
54             color: Colors.grey.withOpacity(0.2), // 'withOpacity' is deprecated and sh
55             spreadRadius: 2,
56             blurRadius: 5,
57             offset: const Offset(0, 3),
58           ), // BoxShadow
59         ],
60       ),
61     );
62   }
63 }
```

4. Navigation Bar:

```
1 import 'package:flutter/material.dart';
2 import 'package:google_fonts/google_fonts.dart';
3 import 'package:grocery_app/global.dart';
4 import 'package:grocery_app/pages/cart_page.dart';
5 import 'package:grocery_app/pages/favorite_page.dart';
6 import 'package:grocery_app/pages/home_page.dart';
7 import '../settings.dart';
8 import 'drawer.dart';
9 import 'package:curved_navigation_bar/curved_navigation_bar.dart';
10
11 class MainPage extends StatefulWidget {
12   const MainPage({super.key});
13
14   @override
15   State<MainPage> createState() => _MainPageState();
16 }
17
18 class _MainPageState extends State<MainPage> {
19   int _selectedIndex = 0;
20
21   final List<Widget> _pages = [
22     const HomePage(),
23     FavoritePage(), // Use 'const' with the constructor to improve performance.
24     const CartPage(),
25     MySettingsPage(), // Use 'const' with the constructor to improve performance.
26   ];
27
28   void _onItemTapped(int index) {
29     setState(() {
30       _selectedIndex = index;
31     });
32   }
33
34   @override
35   Widget build(BuildContext context) {
36     return Scaffold(
37       appBar: AppBar(
38         title: Text(
39           'Bag It',
40           style: GoogleFonts.notoSerif(fontSize: 20),
41           // Text
42           centerTitle: true,
43           backgroundColor: const Color(0xffe4e7f6),
44         ), // AppBar
45       drawer: AppDrawer(), // Use 'const' with the constructor to improve performance.
46       body: _pages[_selectedIndex],
47       bottomNavigationBar: CurvedNavigationBar(
48         index: _selectedIndex,
49         height: 60.0,
50         items: const [
51           Icon(Icons.home, size: 30, color: Colors.white),
52           Icon(Icons.favorite, size: 30, color: Colors.white),
53           Icon(Icons.shopping_cart, size: 30, color: Colors.white),
54           Icon(Icons.person, size: 30, color: Colors.white),
55         ],
56         color: Colors.primaryColor,
57         buttonBackgroundColor: const Color.fromARGB(255, 13, 116, 129),
58         backgroundColor: Colors.white,
59         animationCurve: Curves.easeInOut,
60         animationDuration: const Duration(milliseconds: 300),
61         onTap: _onItemTapped,
62       ), // CurvedNavigationBar
63     ); // Scaffold
64   }
65 }
66
```


Implementation Details

State Management:

- **Provider:** Used for managing app state (favorites, cart items, etc.).
 - Example: Provide code snippets for **CartModel** and **FavoriteModel**.

Color Theme and UI Enhancements:

- **Primary Colors:**
 - Background Gradient: Light Blue (#8EC5FC) and Light Purple (#E0C3FC).
 - Accent: Black and White for contrasts.
- **Consistent Styling:**
 - Used **Google Fonts** for typography and a gradient theme for a polished look.

Challenges and Solutions:

- **Challenge:** Managing state efficiently for favorites and the cart.
 - **Solution:** Used the **Provider** package to separate UI and logic, ensuring a clean and maintainable state management system.
- **Challenge:** Ensuring UI consistency across pages.
 - **Solution:** Defined a global theme in **global.dart**, which allows for a consistent look and feel across the app.

Performance and Optimization Recommendations:

1. **Lazy Loading:** For pages with a large list of items (like the Home page), consider using **Lazy Loading** with **ListView.builder** to improve performance.
2. **Error Handling:** Implement **try-catch** blocks for network requests or any potentially error-prone operations, especially in the cart and favorites management.
3. **User Feedback:** Introduce user feedback mechanisms like snack bars or dialog boxes when items are added to the cart or marked as favorite.
4. **Testing:** Add unit tests for models (e.g., cart and favorite models) and widget tests for UI elements to ensure robustness.

Your provided code for the **MainPage** and **IntroScreen** involves some key OOP principles in the **Flutter/Dart** environment. Here's an analysis of the Object-Oriented Programming (OOP) aspects in this code:

OOP Principles Used:

1. Classes and Objects:

- **MainPage** and **IntroScreen** are both **StatelessWidget** and **StatefulWidget** classes. These are fundamental constructs in Flutter that represent components of the user interface.
 - **MainPage** is a stateful widget, meaning it has mutable state (`_selectedIndex` and `_pages`) that can change based on user interaction.
 - **IntroScreen** is a stateless widget since it does not hold any state that changes during its lifecycle.

These classes encapsulate the UI structure and behavior of different parts of the app.

2. State Management:

- The **_MainPageState** class manages the state for the **MainPage** widget. Specifically, it keeps track of the currently selected tab in the bottom navigation bar (`_selectedIndex`). This is an example of **encapsulation** where the internal state (index) is kept private to the class and only modified through the method `_onItemTapped`.

3. Encapsulation:

- **_MainPageState** and **IntroScreen** encapsulate their respective UI logic and layout. The components within each of these classes (like `Text`, `GestureDetector`, and `Scaffold`) are structured and organized into self-contained units.
- For example, the `_pages` list holds the different pages to display in the main app, and the `_onItemTapped` method encapsulates the logic for changing between these pages.

4. Inheritance:

- The classes **MainPage** and **IntroScreen** inherit from **StatelessWidget** and **StatefulWidget**, which are base classes provided by Flutter. This inheritance allows these widgets to use properties and methods from these parent classes, like `build()` and `setState()`.

5. Abstraction:

- The use of widgets like `Scaffold`, `Column`, `Text`, and `Container` abstracts away the complexity of building UI components and allows the developer to focus on the layout and structure rather than the underlying implementation.

6. Constructor and Initialization:

- Both **MainPage** and **IntroScreen** have constructors (e.g., `const MainPage({super.key})` and `const IntroScreen({super.key})`), which is an example of **initialization** in OOP. The `super.key` is passed to the parent constructor of the `StatelessWidget` or `StatefulWidget`.

Abstraction in OOP

Abstraction is the concept of hiding the complexity of a system and exposing only the necessary details to the user. It allows developers to focus on the "what" the system should do, without worrying about the underlying details of how it performs those actions.

In Flutter, both `StatelessWidget` and `StatefulWidget` abstract away the complex details of managing UI rendering and state changes. As a developer, you don't need to worry about how the widget's internal state is managed or how the UI updates itself when state changes. You simply define how the widget should behave by overriding certain methods like `build()`.

StatelessWidget and StatefulWidget as Abstractions

1. StatelessWidget:

- **Abstraction:** A `StatelessWidget` represents a part of the UI that does not change over time. You define the UI in the `build()` method, and Flutter takes care of rendering it and managing its lifecycle.
- You don't need to manually manage when the widget should update because the widget doesn't maintain any mutable state.

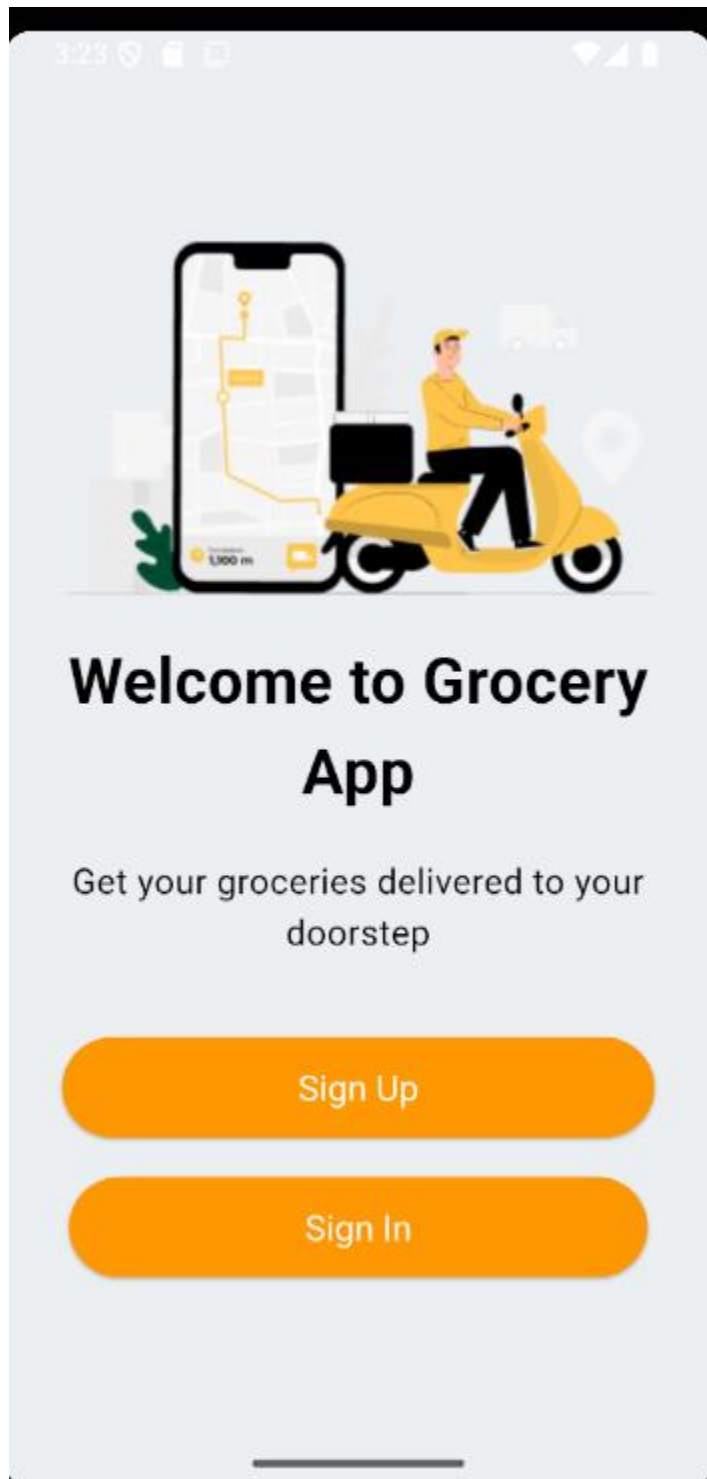
2. StatefulWidget:

- **Abstraction:** A `StatefulWidget` allows for dynamic changes in the UI based on mutable state. The state is encapsulated in a `State` object, and the `StatefulWidget` abstracts the complexity of managing state changes and triggering UI updates. You define how the widget looks when the state changes, and Flutter handles the rest, including calling `setState()` to update the UI when the state changes.
- As a developer, you don't need to manually manage the low-level details of when to redraw the UI or handle state transitions. You simply call `setState()` and define the state and UI updates.

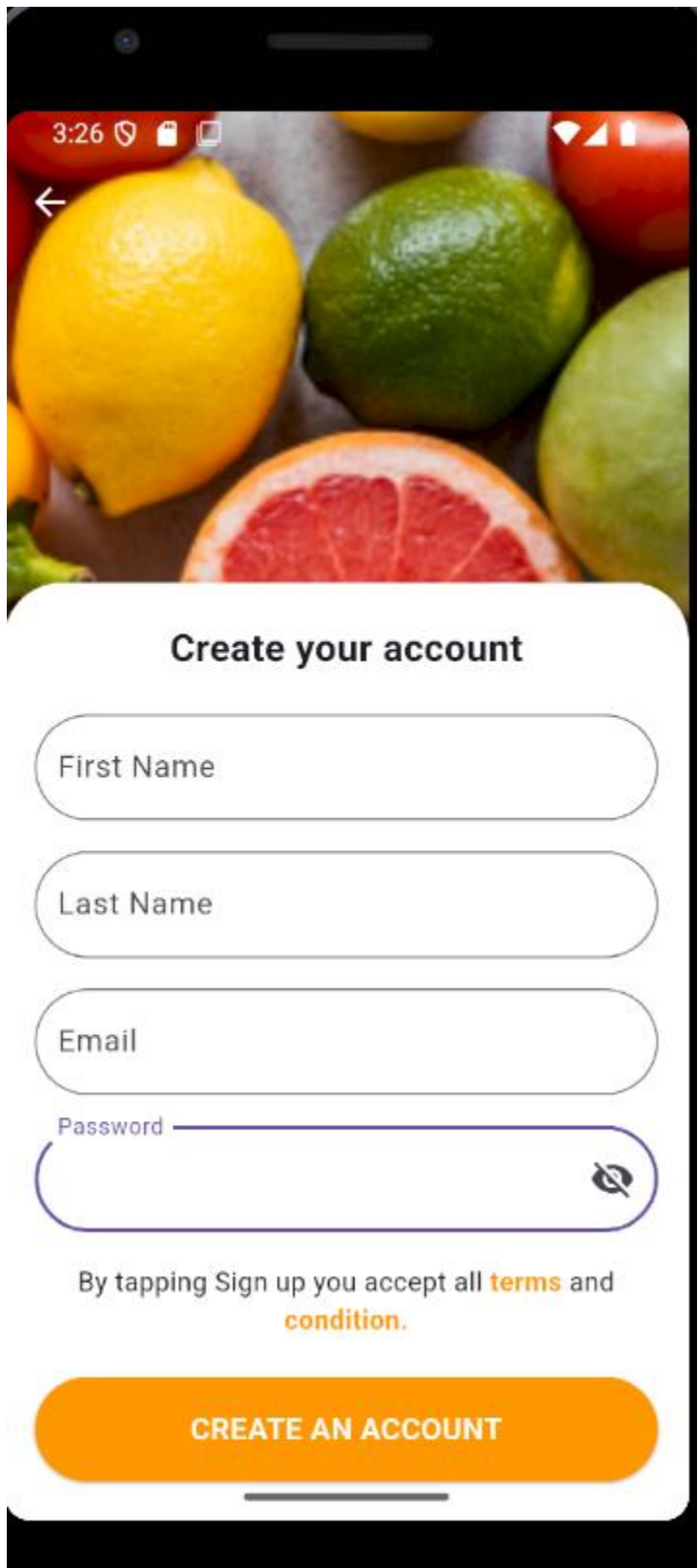
Successfully developed a feature-rich grocery app with an interactive UI and efficient state management.

3. Users can browse, favorite, and add items to their cart seamlessly.
4. The app is designed with a clean UI and optimized for usability, ensuring a positive user experience.
 -

SCREENS :



SCREEN 2



3:26

←

Create your account

First Name

Last Name

Email

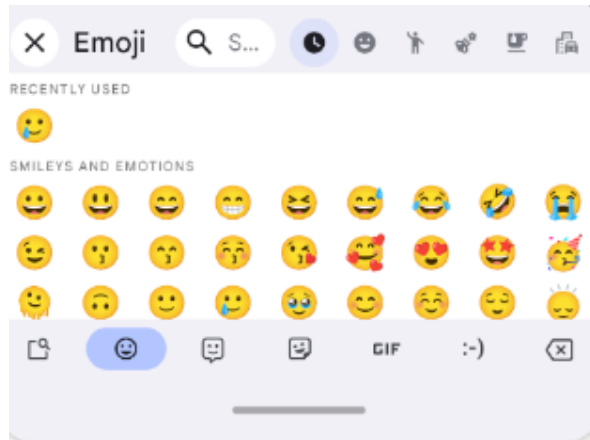
Password

By tapping Sign up you accept all **terms** and **condition.**

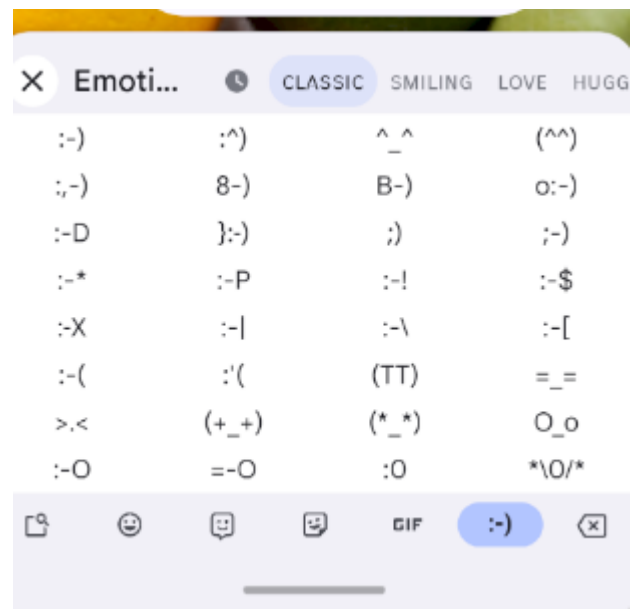
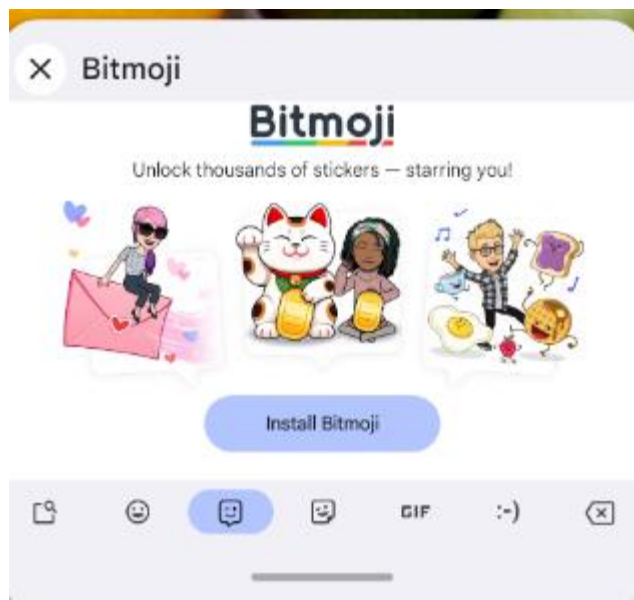
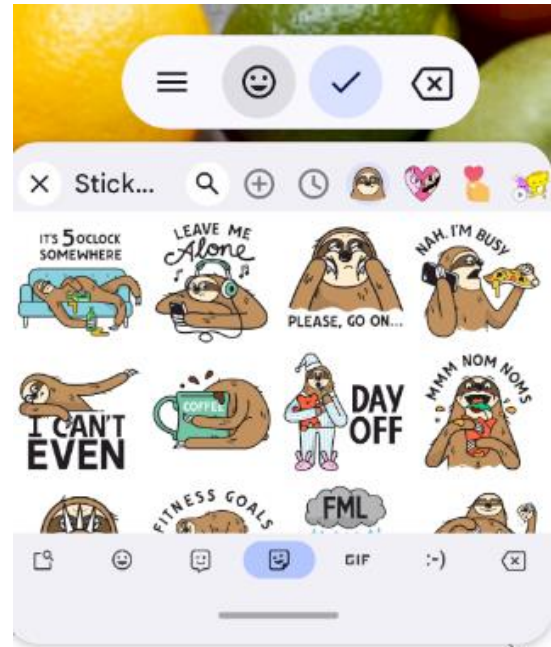
CREATE AN ACCOUNT

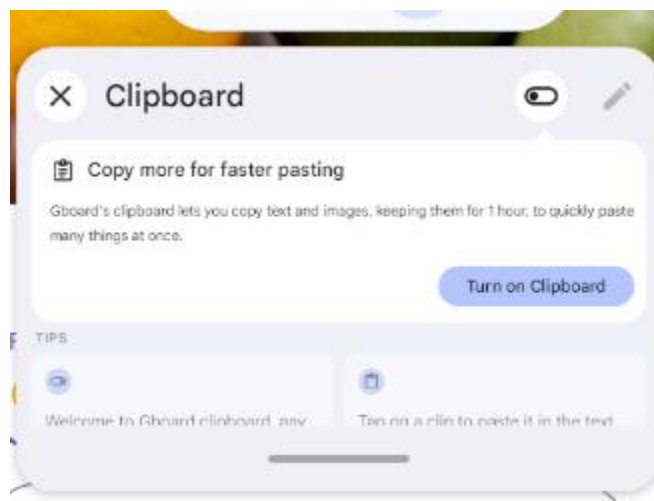
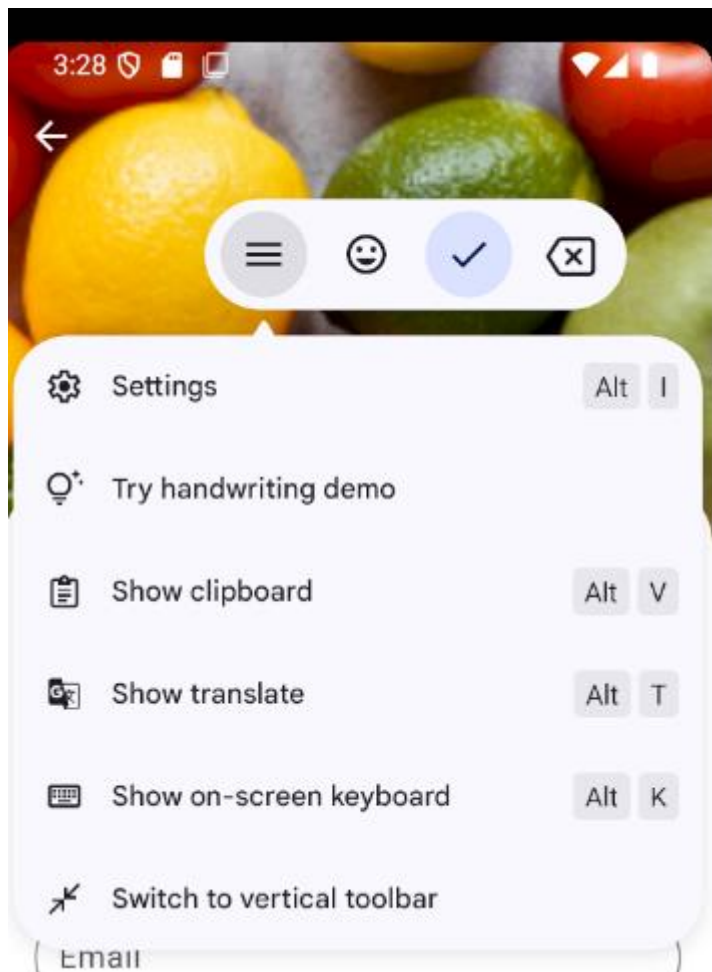
• SCREEN 3

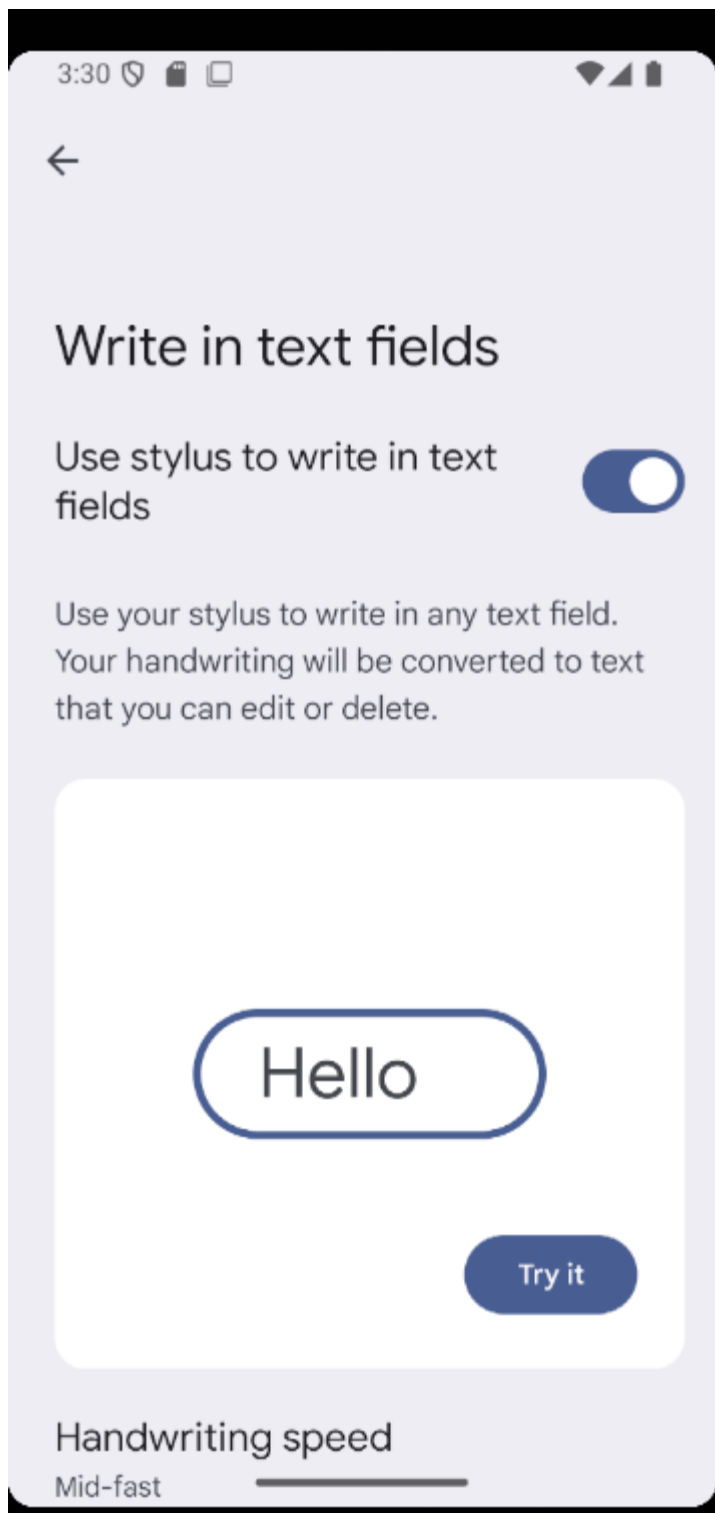
HORIZONTAL TAB AND FEATURES

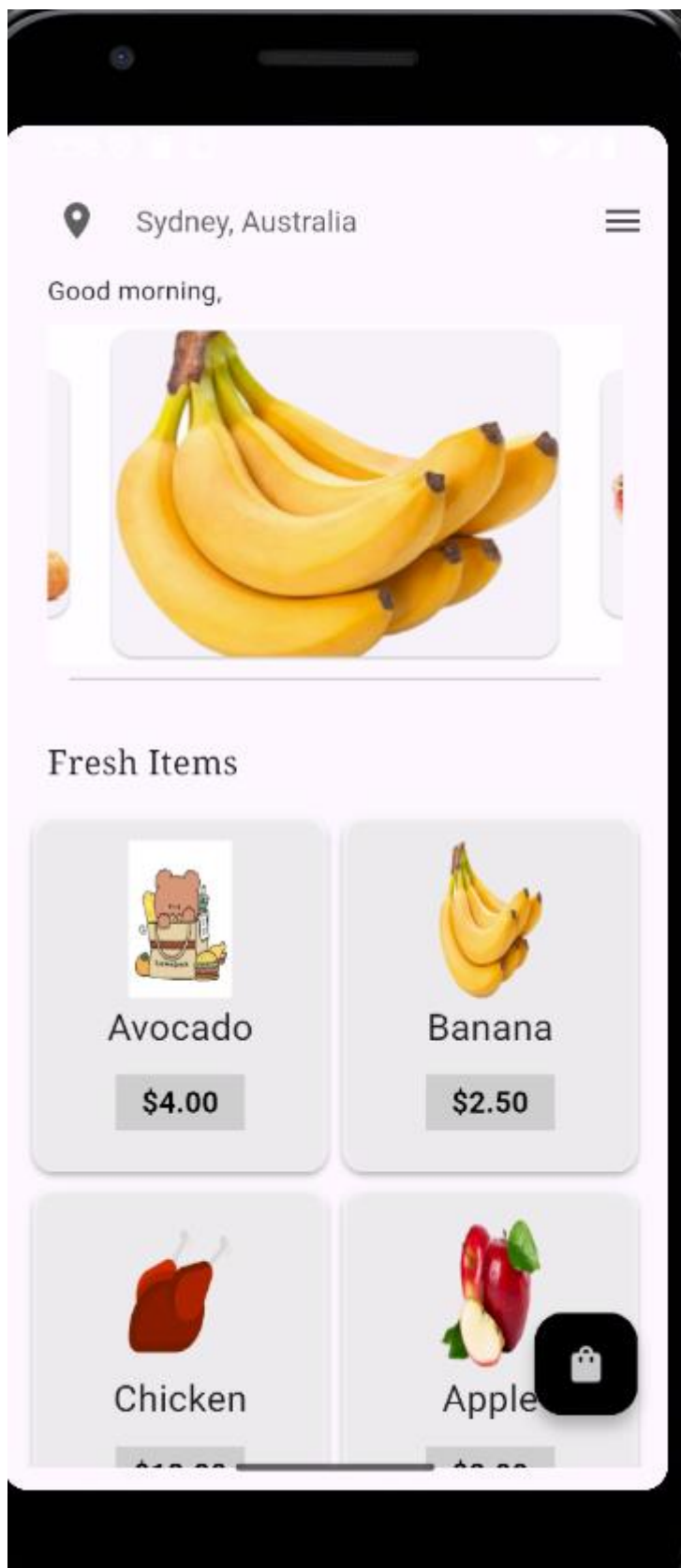


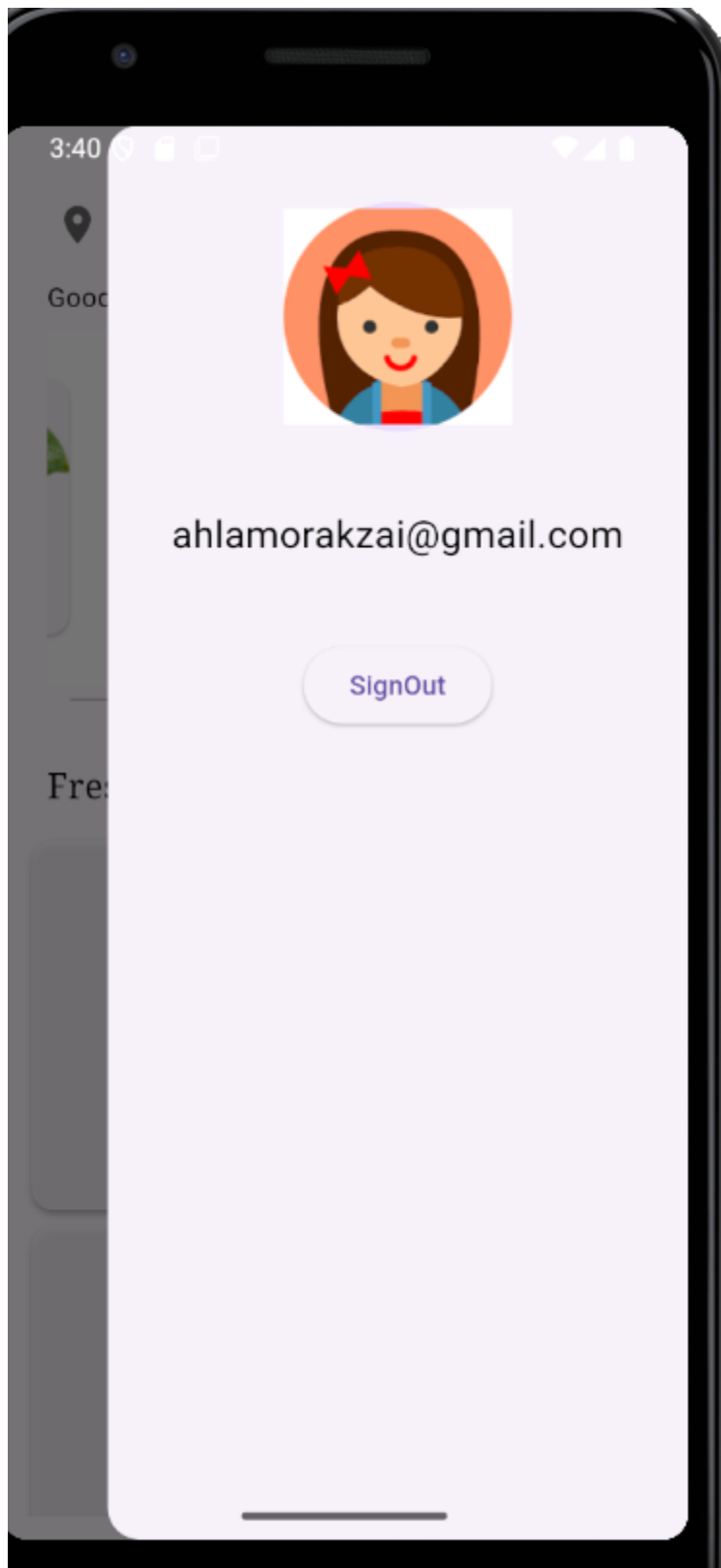
S

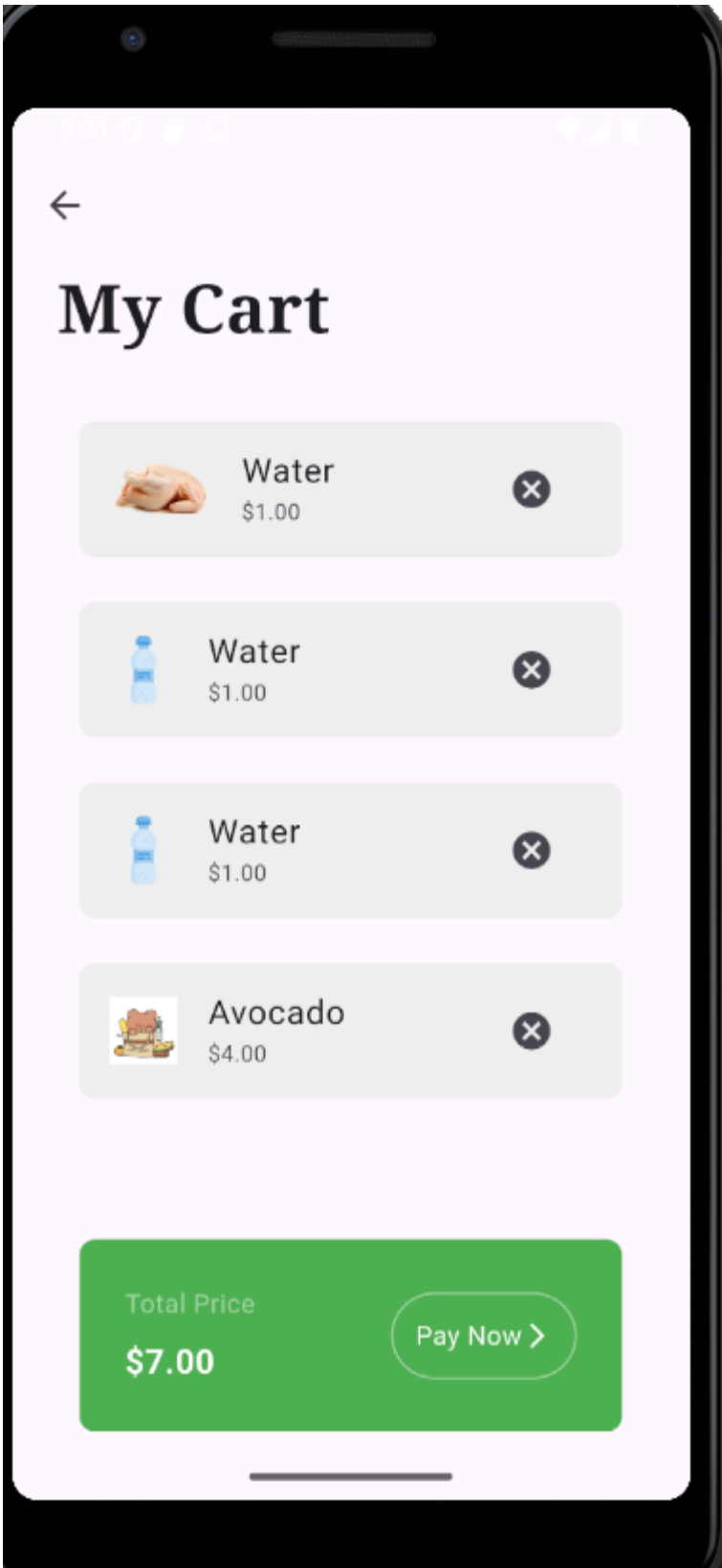












References

- **Flutter Documentation:** <https://flutter.dev/docs>
- **Provider Package:** <https://pub.dev/packages/provider>
- **Google Fonts:** <https://fonts.google.com>
- **Mitch Koko:** For insights on state management and Flutter best practices.
- **ChatGPT:** OpenAI's conversational AI used for code review and implementation suggestions.
- **Gemini AI:** Google's advanced language models for natural language understanding and AI-powered assistance.
- **GitHub Repositories:** Contributed by various open-source developers and repositories.