

BÁO CÁO DỰ ÁN HỌC KỲ 2, NĂM HỌC 2022-2023

HỌC PHẦN PHÁT TRIỂN ỨNG DỤNG DI ĐỘNG (CT484)

- **Tên dự án/ứng dụng:** Ứng dụng bán nước (cà phê, trà, trà sữa)
- **Link GitHub mã nguồn:**
<https://github.com/22-23Sem2-Courses/ct48401-project-trib1910320>
- **MSSV 1: B1910320**
- **Họ tên SV 1: Lê Dương Trí**
- **Lớp học phần: CT484-01**

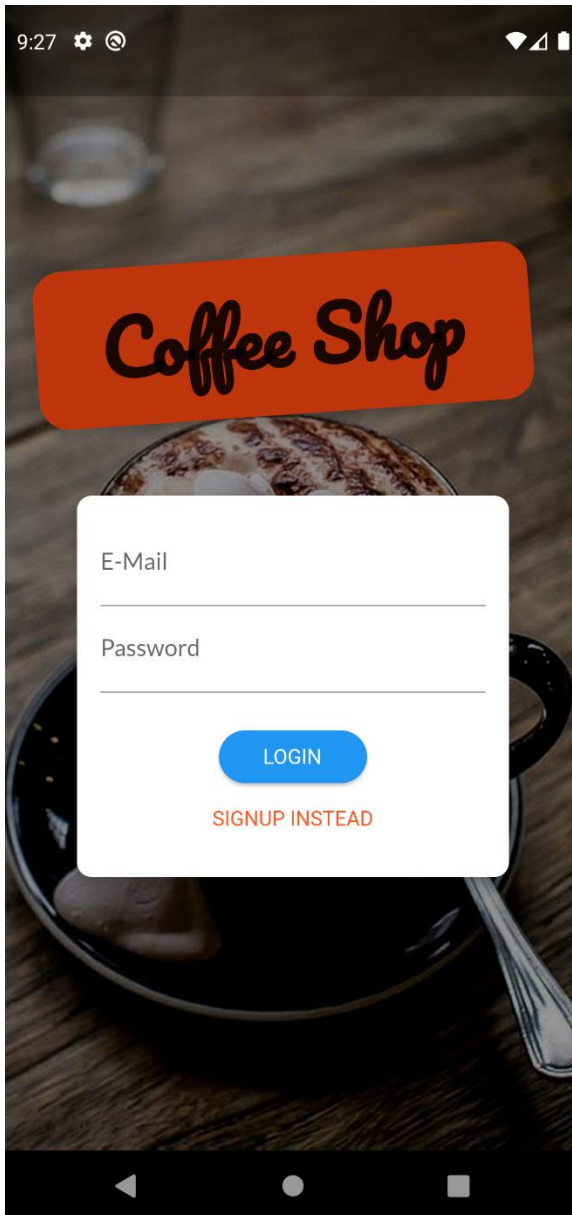
I. Tổng quan

- **Miêu tả dự án/ứng dụng:** ứng dụng bán các loại thức uống như cà phê máy, cà phê Việt Nam, trà trái cây, trà sữa ... Ứng dụng cho phép người dùng đăng ký, đăng nhập tài khoản, thêm sản phẩm yêu thích, thêm sản phẩm vào giỏ hàng, thêm thông tin nhận hàng.

II. Chi tiết các chức năng

1. Giao diện 1: đăng nhập

- **Miêu tả giao diện:** trang đăng nhập cho phép người dùng đăng nhập vào hệ thống.
- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**

+ Các widget được sử dụng:

Scaffold, Stack, Container, SingleChildScrollView, SizedBox, Column, Row, Flexible, Card, Text, IconButton, Material, ValueListenableBuilder, Form, TextFormField, ElevatedButton, TextButton

+ Các thư viện/plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

flutter/foundation.dart: Cung cấp các lớp và các hàm cơ bản để sử dụng trong việc xây dựng các ứng dụng từ Flutter.

+ Chức năng này có sử dụng giải pháp quản lý trạng thái chia sẻ.

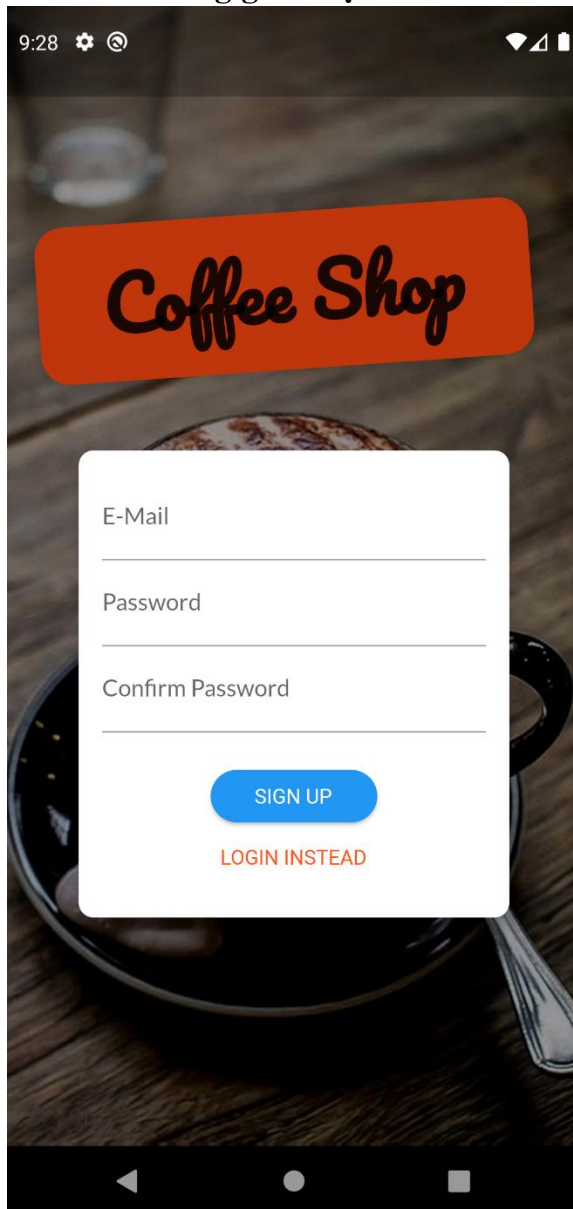
Provider lắng nghe AuthManager để thay đổi giao diện giữa trạng thái đăng nhập và đăng ký.

+ Chức năng này thực hiện đọc dữ liệu đăng nhập được lưu trữ trên Firebase để xác thực người dùng.

2. Giao diện 2: đăng ký

- **Miêu tả giao diện:** trang đăng ký cho phép người dùng tạo tài khoản.

- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**

+ Các widget được sử dụng:

Scaffold, Stack, Container, SingleChildScrollView, SizedBox, Column, Row, Flexible, Card, Text, IconButton, Material, ValueListenableBuilder, Form, TextFormField, ElevatedButton, TextButton

+ Các thư viện/plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

flutter/foundation.dart: Cung cấp các lớp và các hàm cơ bản để sử dụng trong việc xây dựng các ứng dụng từ Flutter.

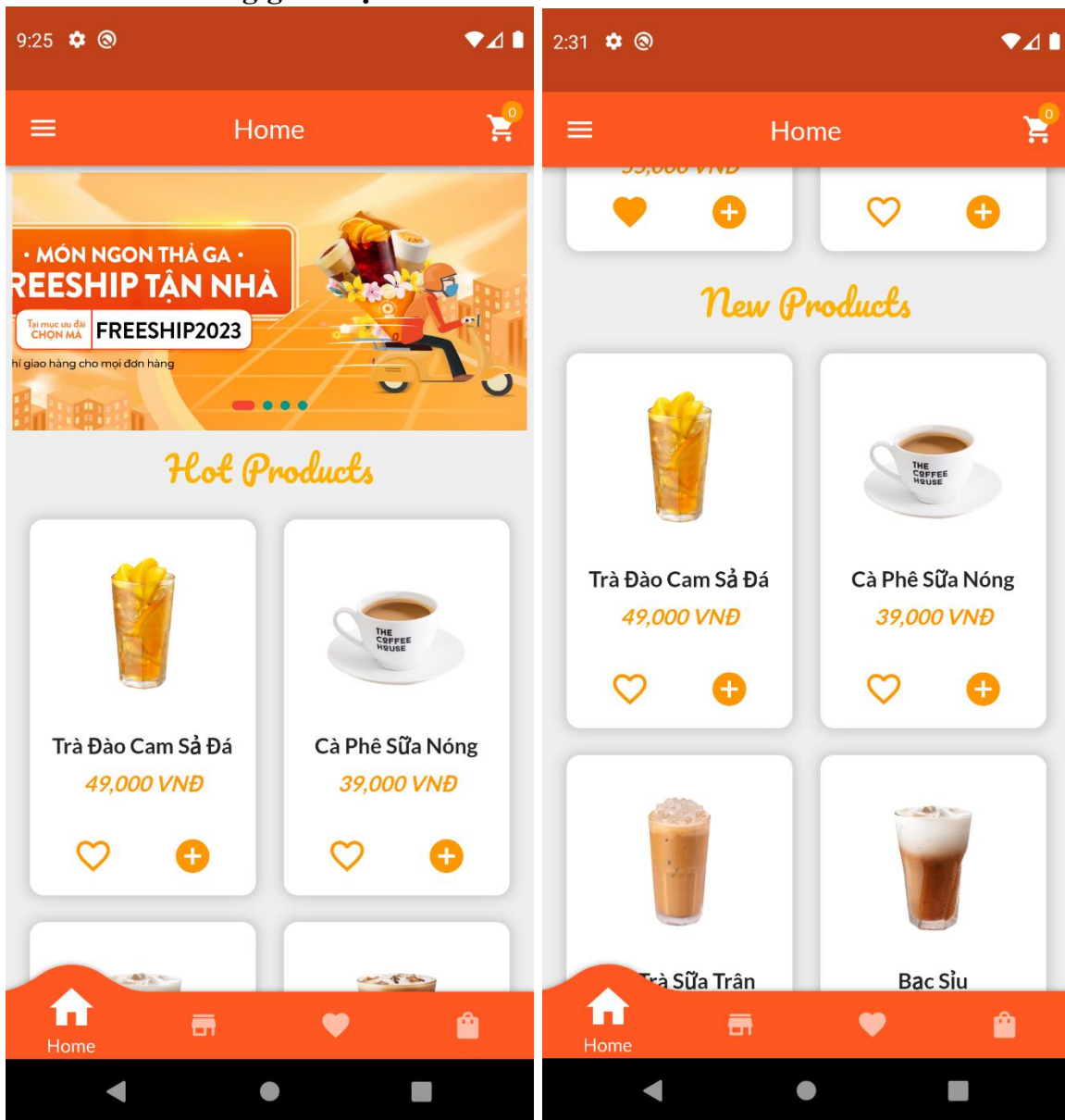
+ Chức năng này có sử dụng giải pháp quản lý trạng thái chia sẻ.

Provider lắng nghe AuthManager để thay đổi giao diện giữa trạng thái đăng nhập và đăng ký.

+ Chức năng này thực hiện ghi dữ liệu đăng ký lưu trữ lên Firebase.

3. Giao diện 3: trang chủ

- **Miêu tả giao diện:** hiển thị các sản phẩm mới và sản phẩm đang hot.
- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**

+ Các widget được sử dụng:

Scaffold, AppBar, ListView, Container, Stack, InkWell, Positioned, Column, Text, Center, FutureBuilder, Consumer, IconButton, GridView, Column, GestureDetector, Expanded, ValueListenableBuilder, Row

+ Các thư viện hay plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

caurousel_slider/caurousel_slider.dart: dùng để tạo ra thanh trượt bằng chuyển

+ Giải pháp quản lý trạng thái chia sẻ.

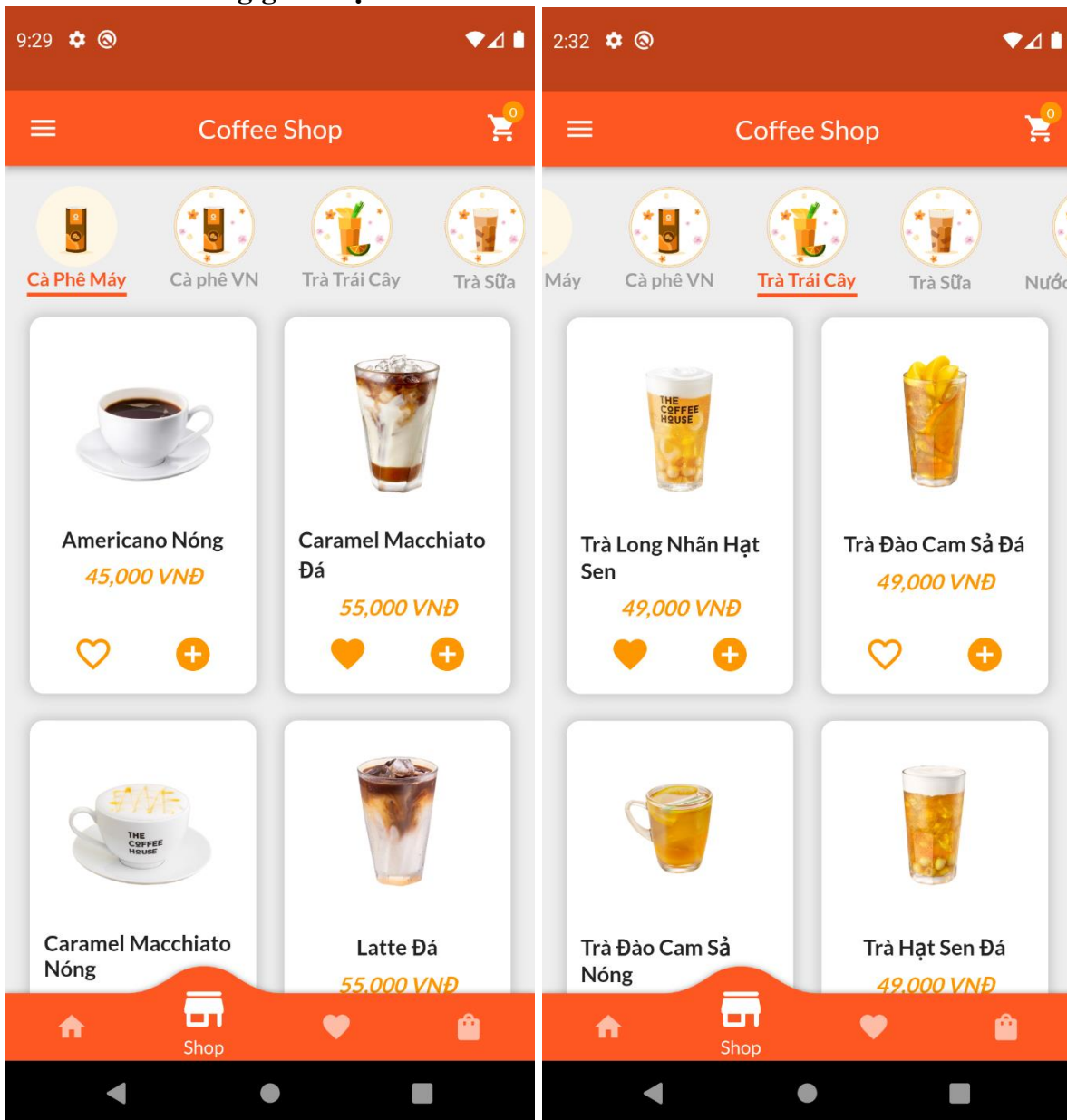
Provider lấy danh sách sản phẩm từ ProductsManager.

+ Chức năng này có thực hiện đọc được lưu trữ trên Firebase

```
"products": {
  ".indexOn": "creatorId",
  "$productId": {
    ".validate": "newData.hasChildren(['creatorId', 'title', 'description', 'imageUrl', 'dateTime', 'quantitySold', 'price', 'type'])",
    ".write": "!data.exists() || data.child('creatorId').val() == 'xKw5GWb1TyYdyMTHiURzhRRC1mI2'"
  }
},
```

4. Giao diện 4: trang cửa hàng

- **Miêu tả giao diện:** hiển thị tất cả sản phẩm theo từng loại sản phẩm.
- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**

- + Các widget được sử dụng:

Scaffold, AppBar, FutureBuilder, Consumer, ListView, TabBar, Tab, Column, SizedBox, Text, Center, ValueListenableBuilder, IconButton, GridView, Container, GestureDetector, Image, Expanded, Row.

- + Các thư viện hay plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

intl/intl.dart: thư viện này định dạng dữ liệu và thay thế, định dạng ngày và số và phân tích cú pháp, và các tiện ích để làm việc với văn bản hai chiều.

+ Giải pháp quản lý trạng thái chia sẻ.

Provider lấy danh sách sản phẩm từ ProductsManager và lắng nghe sự thay đổi của CartManager.

+ Chức năng này có thực hiện được lưu trữ trên Firebase

```
"products": {
  ".indexOn": "creatorId",
  "$productId": {
    ".validate": "newData.hasChildren(['creatorId', 'title', 'description', 'imageUrl', 'dateTime', 'quantitySold', 'price', 'type'])",
    ".write": "!data.exists() || data.child('creatorId').val() == 'xKw5GWb1TyYdyMTHiURzhRRC1mI2'"
  }
},
"types": {
  ".indexOn": "creatorId",
  "$typeId": {
    ".validate": "newData.hasChildren(['creatorId', 'title', 'imageUrl'])",
    ".write": "!data.exists() || data.child('creatorId').val() == 'xKw5GWb1TyYdyMTHiURzhRRC1mI2'"
  }
},
}
```


5. Giao diện 5: chi tiết sản phẩm

- **Miêu tả giao diện:** hiển thị thông tin của sản phẩm, cho phép thêm 1 số lượng sản phẩm và chọn kích thước sản phẩm vào giỏ hàng, thêm sản phẩm vào danh sách yêu thích
- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**
 - + Các widget được sử dụng:
Scaffold, Stack, Text, SingleChildScrollView, Container, Column, SizedBox, Image, Row, ValueListenableBuilder, IconButton, Wrap, List.generate, ChoiceChip, GestureDetector
 - + Các thư viện hay plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

intl/intl.dart: thư viện này định dạng dữ liệu và thay thế, định dạng ngày và số và phân tích cú pháp, và các tiện ích để làm việc với văn bản hai chiều.

+ Giải pháp quản lý trạng thái chia sẻ.

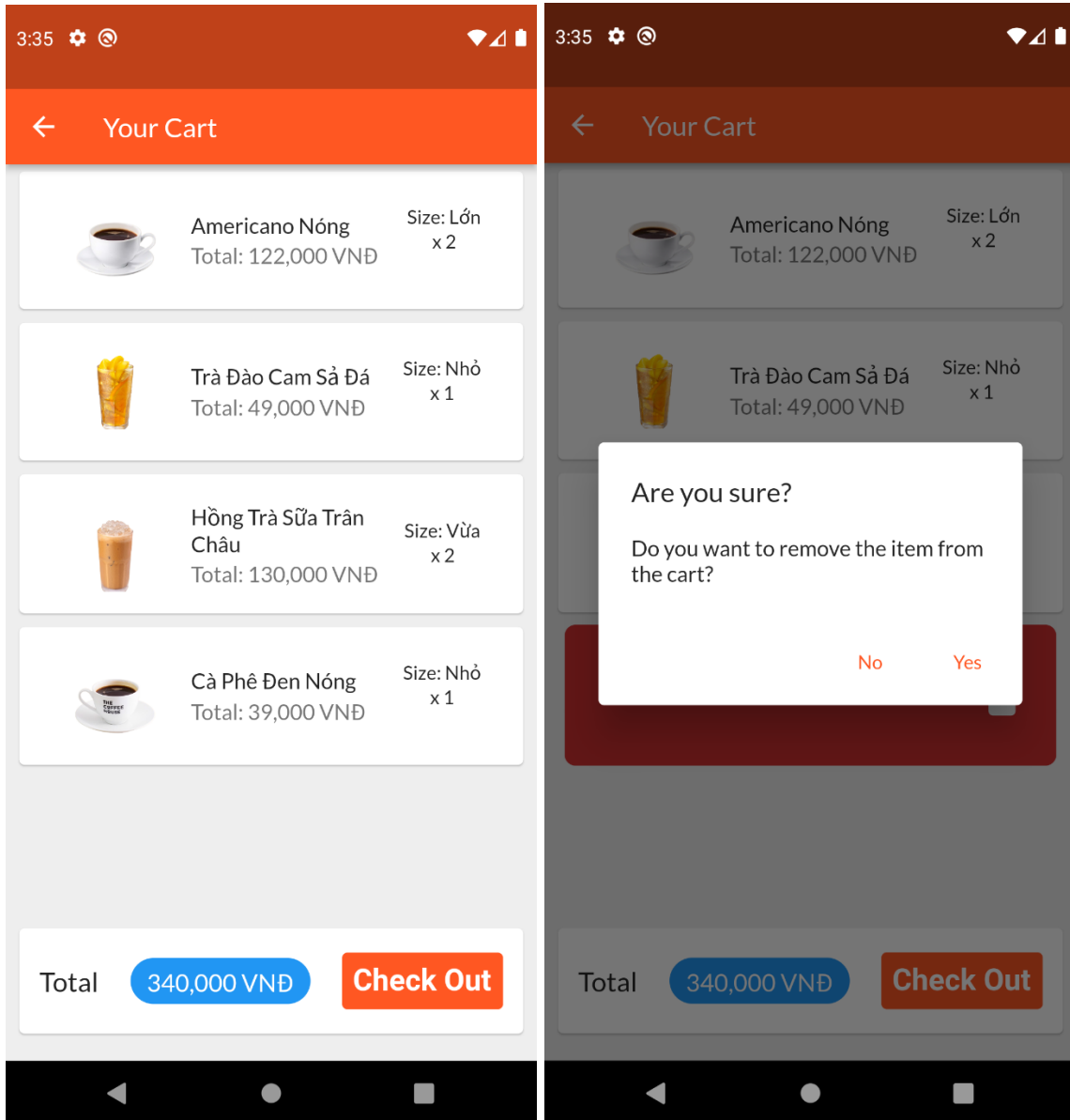
Provider lấy thông tin sản phẩm từ ProductsManager và thêm sản phẩm vào giỏ hàng từ CartManager.

+ Chức năng này có thực hiện đọc được lưu trữ trên Firebase và thêm sản yêu thích lưu trữ trên Firebase

```
"products": {
  ".indexOn": "creatorId",
  "$productId": {
    ".validate": "newData.hasChildren(['creatorId', 'title', 'description', 'imageUrl', 'dateTime', 'quantitySold', 'price', 'type'])",
    ".write": "!data.exists() || data.child('creatorId').val() == 'xKw5GWbTyYdyMTHiURzhRRC1mI2'"
  }
},
```

6. Giao diện 6: giỏ hàng

- **Miêu tả giao diện:** hiển thị các sản phẩm trong giỏ hàng, xóa sản phẩm trong giỏ hàng.
- **Ảnh chức năng/giao diện:**



- Chi tiết cài đặt:

+ Các widget được sử dụng:

Scaffold, AppBar, Column, Expanded, SizeBox, ListView, Card, Padding, Row, Text, Chip, Spacer, TextButton, Dismissible, ListTile, Image

+ Các thư viện hay plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

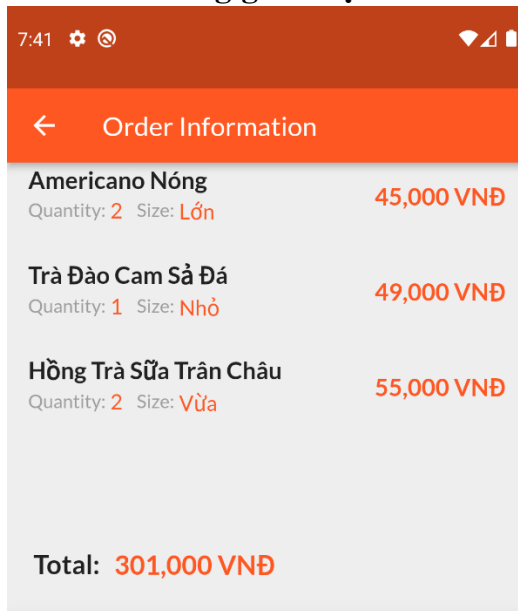
intl/intl.dart: thư viện này định dạng dữ liệu và thay thế, định dạng ngày và số và phân tích cú pháp, và các tiện ích để làm việc với văn bản hai chiều.

+ Giải pháp quản lý trạng thái chia sẻ.

Provider lấy danh sách sản phẩm của giỏ hàng từ CartManager

7. Giao diện 7: Điền thông tin đơn hàng và đặt hàng

- **Miêu tả giao diện:** điền thông tin chi tiết của người dùng
- **Ảnh chức năng/giao diện:**



Order information

- **Chi tiết cài đặt:**

+ Các widget được sử dụng:

Scaffold, AppBar, SingleChildScrollView, Column, SizedBox, ListView, ListTile, Container, Row, Text, Form, Padding, TextFormField, TextButton, showDialog, AlertDialog

+ Các thư viện/plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

intl/intl.dart: thư viện này định dạng dữ liệu và thay thế, định dạng ngày và số và phân tích cú pháp, và các tiện ích để làm việc với văn bản hai chiều.

+ Chức năng này có sử dụng giải pháp quản lý trạng thái chia sẻ.

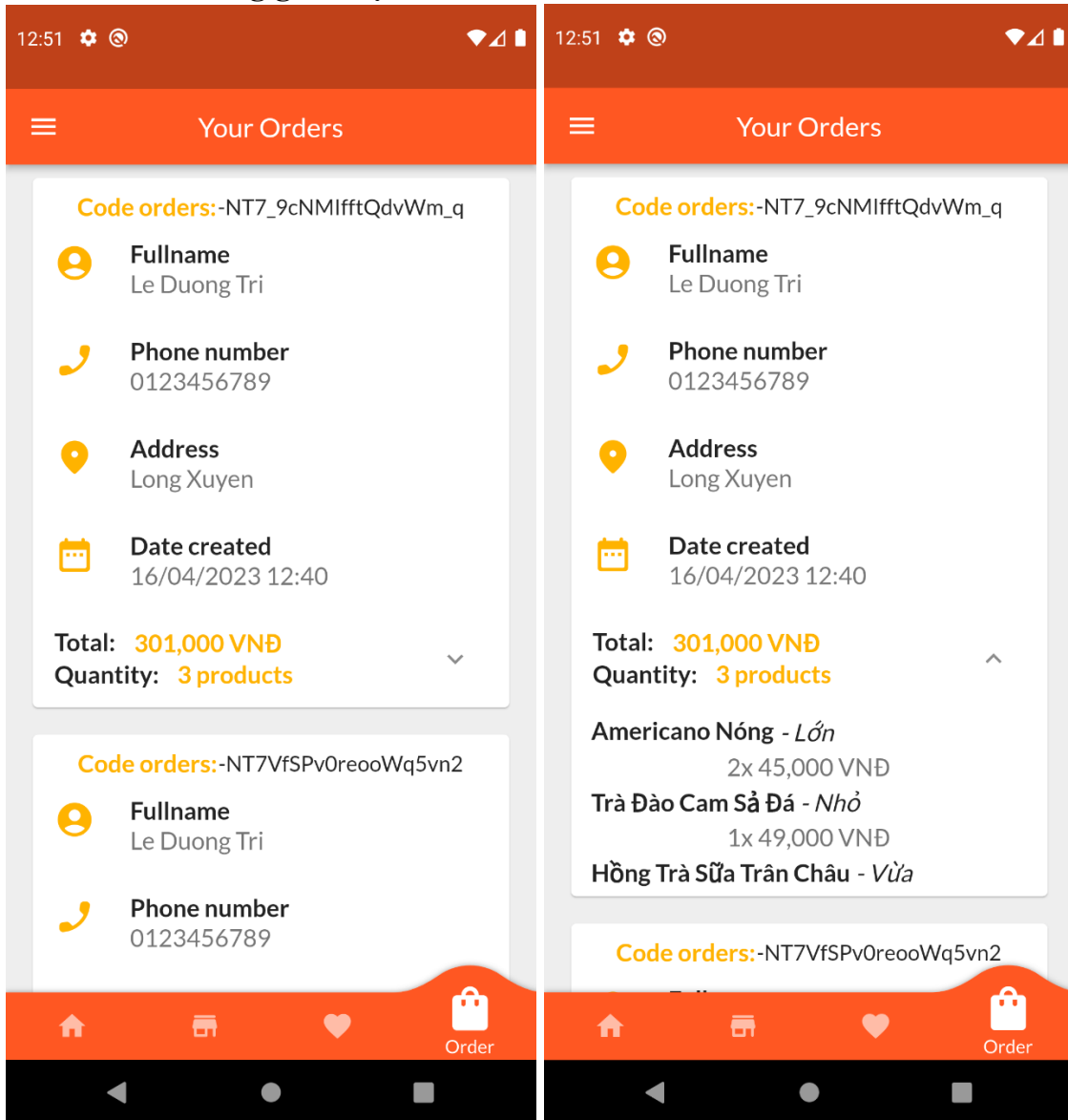
Provider lấy danh sách sản phẩm từ CartManager và truyền về cho OrderManager lắng nghe.

+ Chức năng này thực hiện ghi dữ liệu thông tin đơn hàng và lưu trữ trên Firebase.

```
"orders": {  
  ".indexOn": "creatorId",  
  "$orderId": {  
    ".validate": "newData.hasChildren(['creatorId', 'name', 'address', 'phone', 'amount', 'products', 'dateTime'])",  
    ".write": "!data.exists() || data.child('creatorId').val() == auth.uid"  
  }  
},
```

8. Giao diện 8: xem đơn hàng đã đặt

- **Miêu tả giao diện:** hiển thị danh sách đơn hàng đã đặt và thông tin chi tiết.
- **Ảnh chức năng/giao diện:**



- Chi tiết cài đặt:

+ Các widget được sử dụng:

Scaffold, AppBar, Text, FuterBuilder, Center, Consumer, ListView, Card, Container, Column, Row, SizeBox, Icon, IconButton, ListTile

+ Các thư viện/plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

intl/intl.dart: thư viện này định dạng dữ liệu và thay thế, định dạng ngày và số và phân tích cú pháp, và các tiện ích để làm việc với văn bản hai chiều.

+ Chức năng này có sử dụng giải pháp quản lý trạng thái chia sẻ.

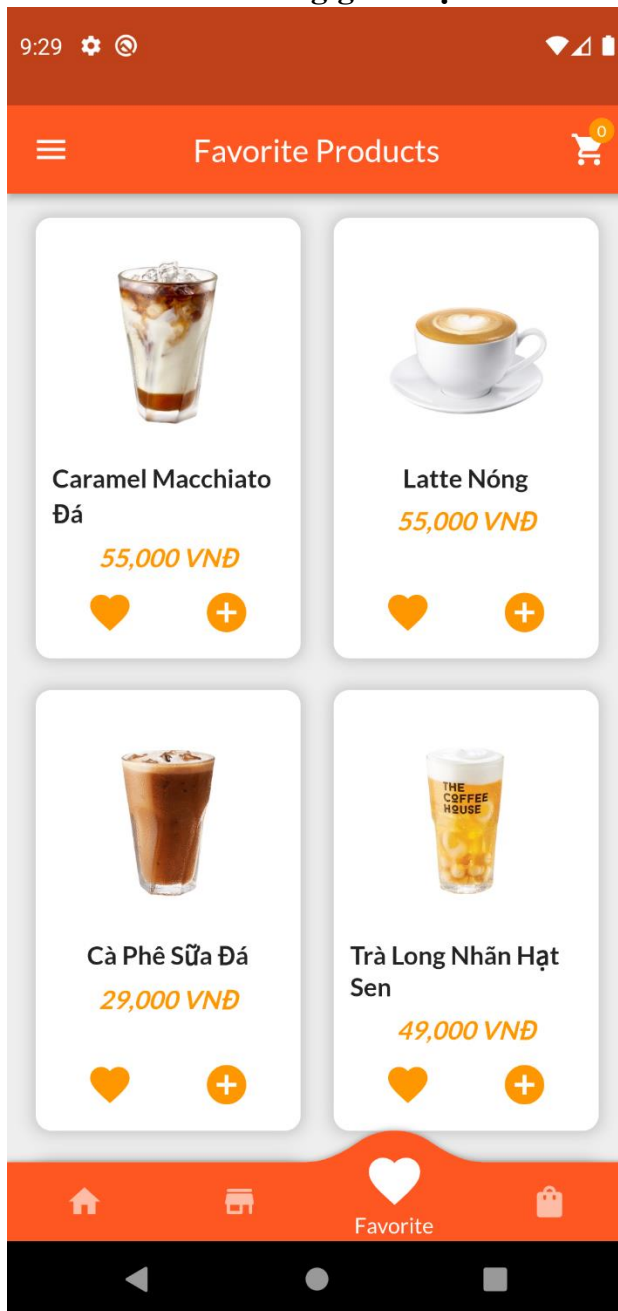
Provider lắng nghe OrderManager và AuthManager để hiển thị các sản phẩm đã đặt hàng.

+ Chức năng này thực hiện đọc dữ liệu thông tin đặt hàng được lưu trữ trên Firebase.

```
"orders": {  
  ".indexOn": "creatorId",  
  "$orderId": {  
    ".validate": "newData.hasChildren(['creatorId', 'name', 'address', 'phone', 'amount', 'products', 'dateTime'])",  
    ".write": "!data.exists() || data.child('creatorId').val() == auth.uid"  
  }  
},
```

9. Giao diện 9: xem sản phẩm yêu thích

- **Miêu tả giao diện:** hiển thị các sản phẩm yêu thích
- **Ảnh chức năng/giao diện:**



- Chi tiết cài đặt:

+ Các widget được sử dụng:

Scaffold, AppBar, ListView, FutureBuilder, ValueListenableBuilder, Container, Center, GridView, Column, GestureDetector, Image, SizeBox, Expanded, Row, IconButton

+ Các thư viện hay plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

intl/intl.dart: thư viện này định dạng dữ liệu và thay thế, định dạng ngày và số và phân tích cú pháp, và các tiện ích để làm việc với văn bản hai chiều.

- + Giải pháp quản lý trạng thái chia sẻ.

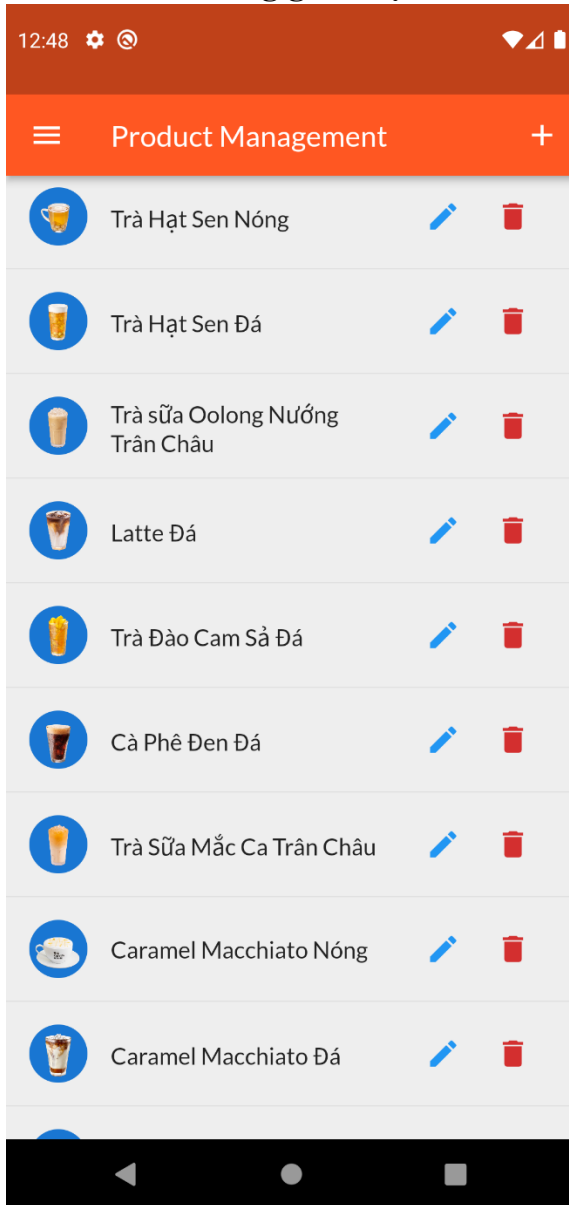
Provider lấy danh sách sản phẩm yêu thích từ ProductsManager.

- + Chức năng này thực hiện đọc dữ liệu lấy danh sách yêu thích của người dùng được lưu trữ trên Firebase.

```
"userFavorites": {  
  "$uid": {  
    ".write": "$uid == auth.uid"  
  }  
}
```

10. Giao diện 10: quản lý sản phẩm của admin

- **Miêu tả giao diện:** quản trị viên xem danh sách sản phẩm đã thêm, xóa sản phẩm
- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**

+ Các widget được sử dụng:

Scaffold, AppBar, FutureBuilder, Center, Text, Consumer, ListView, IconButton, Column, Divider, ListTile, CircleAvatar, SizeBox, RefreshIndicator.

+ Các thư viện/plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google.

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

+ Chức năng này có sử dụng giải pháp quản lý trạng thái chia sẻ.

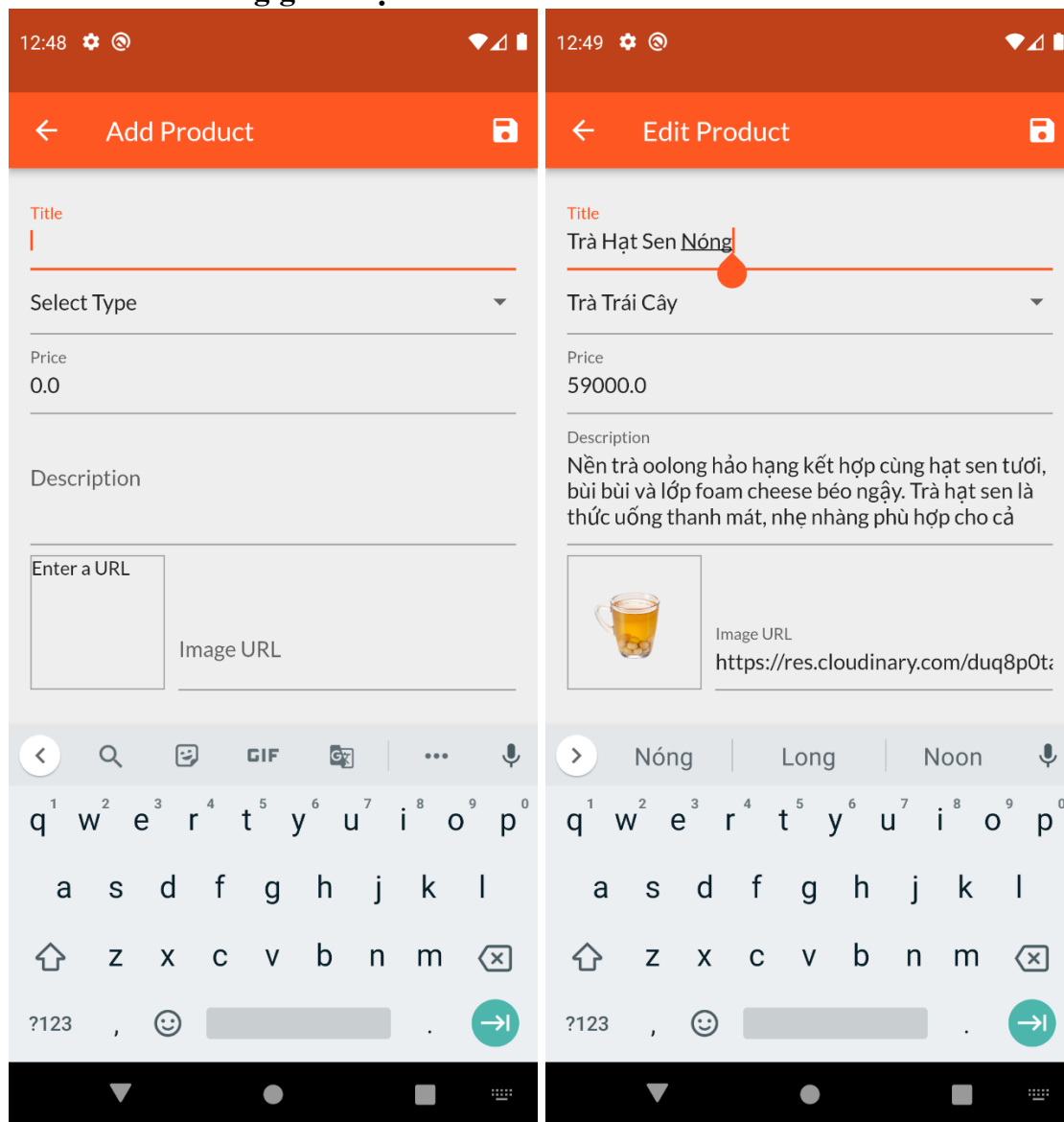
Provider lấy danh sách sản phẩm ProductsManager để hiển thị, lắng nghe việc xóa sản phẩm.

+ Chức năng này thực hiện đọc và ghi dữ liệu sản phẩm được lưu trữ trên Firebase.

```
"products": {  
  ".indexOn": "creatorId",  
  "$productId": {  
    ".validate": "newData.hasChildren(['creatorId', 'title', 'description', 'imageUrl', 'dateTime', 'quantitySold', 'price', 'type'])",  
    ".write": "!data.exists() || data.child('creatorId').val() == 'xKw5GWb1TyYdyMTHiURzhRRC1mI2'"  
  }  
},  
..
```

11. Giao diện 11: trang thêm/sửa sản phẩm

- **Miêu tả giao diện:** quản trị viên thêm, sửa sản phẩm.
- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**

+ Các widget được sử dụng:

Scaffold, AppBar, Center, Text, IconButton, Padding, Form, ListView, FutureBuilder, Consumer, Column, Divider, TextFormField, Container, DropdownButtonFormField, DropdownMenuItem, Row, Expanded

+ Các thư viện/plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google.

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

+ Chức năng này có sử dụng giải pháp quản lý trạng thái chia sẻ.

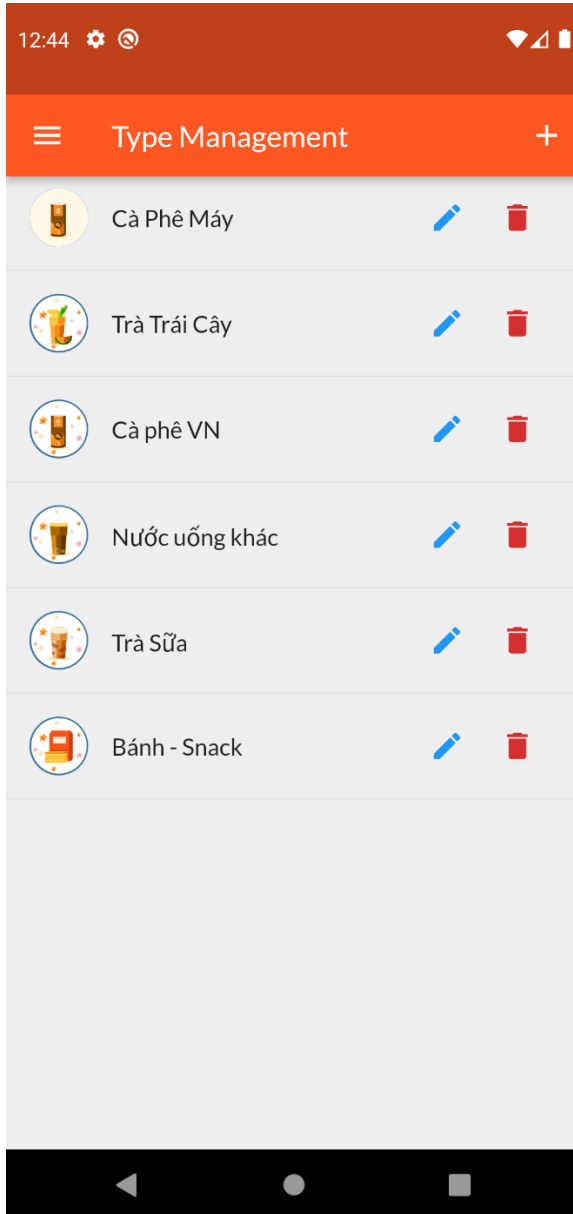
Provider lắng nghe ProductsManager để hiển thị, thêm, sửa sản phẩm.

+ Chức năng này thực hiện đọc và ghi dữ liệu sản phẩm được lưu trữ trên Firebase.

```
"products": {  
  ".indexOn": "creatorId",  
  "$productId": {  
    ".validate": "newData.hasChildren(['creatorId', 'title', 'description', 'imageUrl', 'dateTime', 'quantitySold', 'price', 'type'])",  
    ".write": "!data.exists() || data.child('creatorId').val() == 'xKw5GWb1TyYdyMTHiURzhRRC1mI2'"  
  }  
},  
..
```

12. Giao diện 12: quản lý loại sản phẩm của admin

- **Miêu tả giao diện:** quản trị viên xem danh sách loại sản phẩm đã thêm, xóa loại sản phẩm
- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**

+ Các widget được sử dụng:

Scaffold, AppBar, FutureBuilder, Center, Text, Consumer, ListView, IconButton, Column, Divider, ListTile, CircleAvatar, SizeBox, RefreshIndicator.

+ Các thư viện/plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google.

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

+ Chức năng này có sử dụng giải pháp quản lý trạng thái chia sẻ.

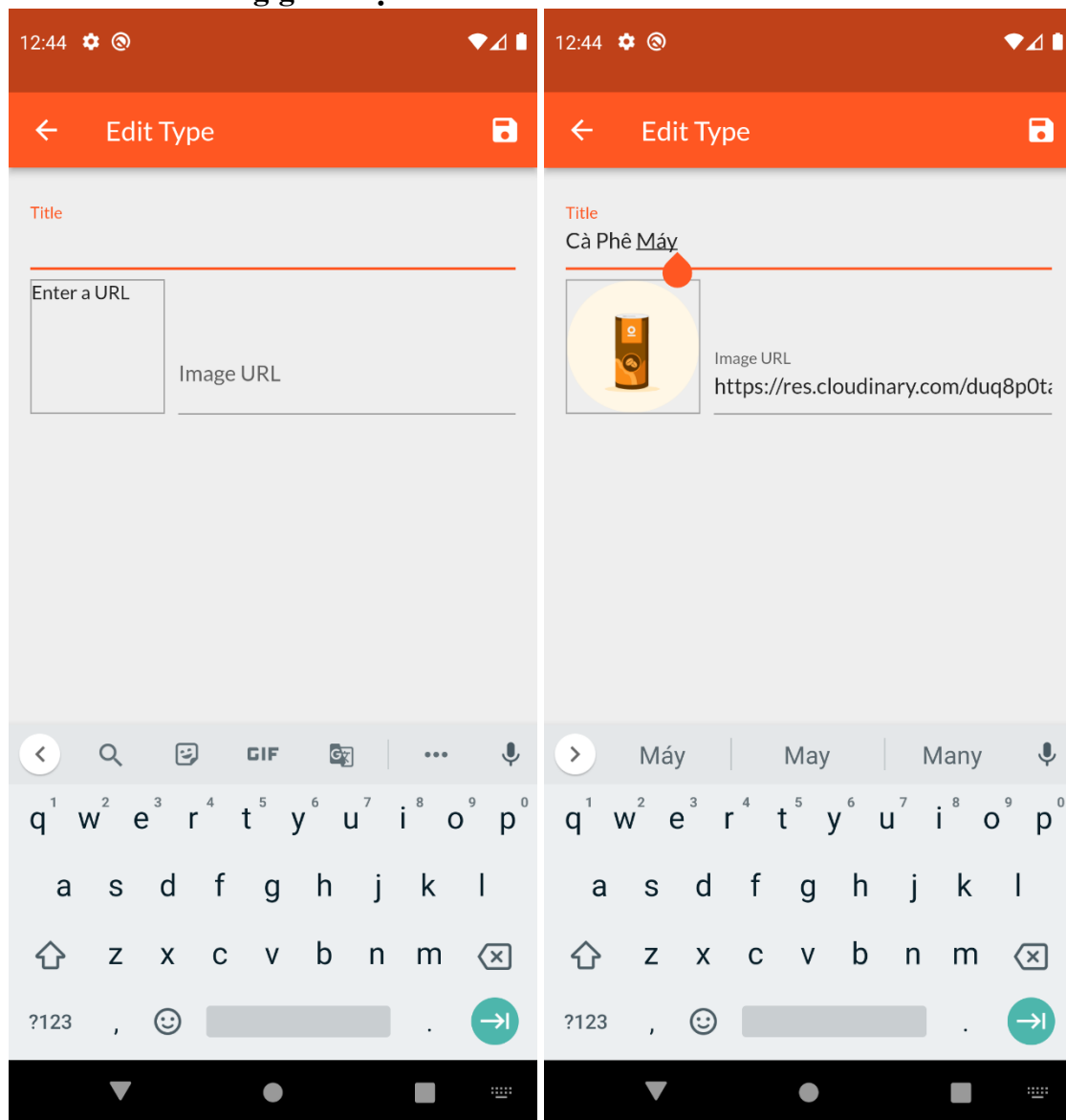
Provider lấy danh sách loại sản phẩm TypesManager để hiển thị, lắng nghe việc xóa loại sản phẩm.

+ Chức năng này thực hiện đọc và ghi dữ liệu sản phẩm được lưu trữ trên Firebase.

```
"types": {  
  ".indexOn": "creatorId",  
  "$typeId": {  
    ".validate": "newData.hasChildren(['creatorId', 'title', 'imageUrl'])",  
    ".write": "!data.exists() || data.child('creatorId').val() == 'xKw5GWb1TyYdyMTHiURzhRRC1mI2'"  
  }  
},
```

13. Giao diện 13: trang thêm/sửa loại sản phẩm

- **Miêu tả giao diện:** quản trị viên xem, thêm, sửa và xóa sản phẩm.
- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**

+ Các widget được sử dụng:

Scaffold, AppBar, Center, Text, IconButton, Padding, Consumer, ListView, Column, Divider, Padding, Form, TextFormField, Row, Expanded, Container

+ Các thư viện/plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google.

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

+ Chức năng này có sử dụng giải pháp quản lý trạng thái chia sẻ.

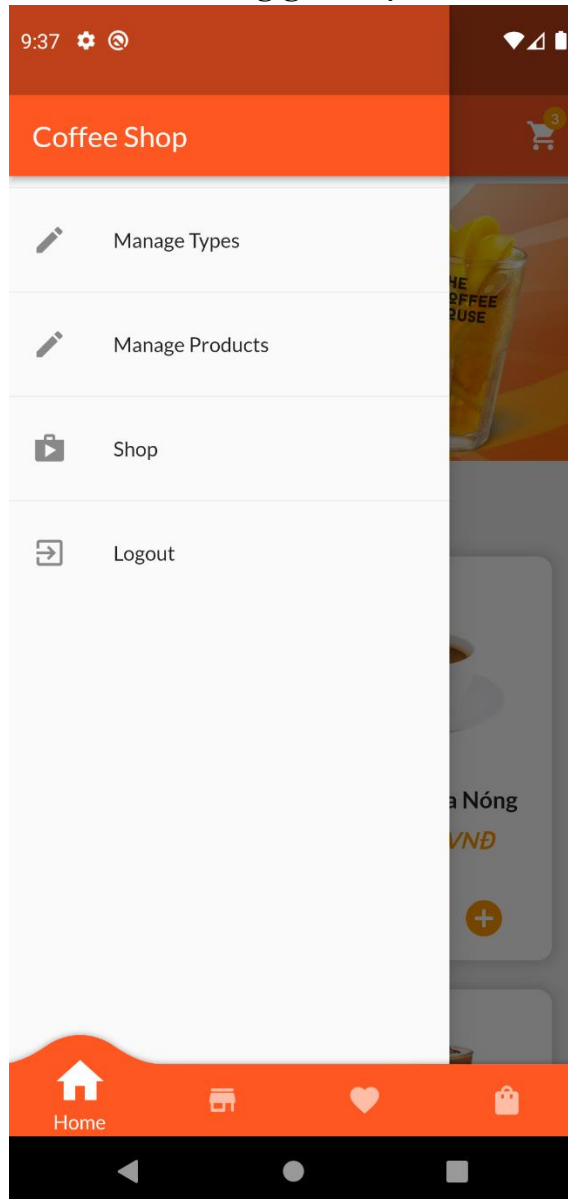
Provider lắng nghe TypesManager để hiển thị, thêm, sửa

+ Chức năng này thực hiện đọc và ghi dữ liệu sản phẩm được lưu trữ trên Firebase.

```
"types": {  
  ".indexOn": "creatorId",  
  "$typeId": {  
    ".validate": "newData.hasChildren(['creatorId', 'title', 'imageUrl'])",  
    ".write": "!data.exists() || data.child('creatorId').val() == 'xKw5GWb1TyYdyMTHiURzhRRC1mI2'"  
  }  
},
```

14. Giao diện 14: Thanh điều hướng

- **Miêu tả giao diện:** Cho phép người dùng thường logout và về trang chủ. Cho phép quản trị viên chuyển hướng đến các trang quản trị của ứng dụng
- **Ảnh chức năng/giao diện:**



- **Chi tiết cài đặt:**

+ Các widget được sử dụng:

Consumer, Drawer, Column, AppBar, Divider, ListTile, Icon

+ Các thư viện/plugin:

flutter/material.dart: thư viện cung cấp các widget và class để tạo giao diện người dùng theo thiết kế Material Design của Google.

provider/provider.dart: thư viện quản lý trạng thái và cung cấp các đối tượng dữ liệu cho Widget.

- + Chức năng này có sử dụng giải pháp quản lý trạng thái chia sẻ.
Provider lắng nghe AuthManager để hiển thị