

ĐẠI HỌC BÁCH KHOA HÀ NỘI

# BÁO CÁO THỰC TẬP KỸ THUẬT

## Xây dựng UI App E-Commerce

ĐẶNG NHẬT QUANG

quang.dn203755@sis.hust.edu.vn

Ngành Kỹ thuật điện tử viễn thông

Chuyên ngành Kỹ thuật máy tính

Công ty hướng dẫn:

Công Ty Cổ Phần LUMI Việt Nam

HÀ NỘI, 9/2023

## ĐỀ TÀI THỰC TẬP

Học và xây dựng UI cho App E-Commerce bằng Framework Flutter.

## **LỜI CẢM ƠN**

Đây là môn học đầu tiên giúp em được tiếp cận tới những kiến thức thực tiễn với công việc định hướng tương lai cũng như được trải nghiệm về môi trường doanh nghiệp, có thể học tập, tìm hiểu, liên kết với các anh chị đi trước ở trong Công ty LUMI Việt Nam. Em xin cảm ơn nhà trường đã tạo điều kiện cho bản thân sinh viên có những trải nghiệm quý báu, cảm ơn Công ty LUMI Việt Nam đã tiếp nhận cũng như chỉ bảo tận tình, giúp đỡ em trong thời gian vừa qua.

## MỤC LỤC

### Table of Contents

<b>CHƯƠNG 1. GIỚI THIỆU CHỨC NĂNG, NHIỆM VỤ, CƠ CẤU TỔ CHỨC CỦA ĐƠN VỊ TIẾP NHẬN.....</b>	<b>6</b>
<b>1.1 Giới thiệu chung .....</b>	<b>6</b>
Công ty cổ phần LUMI Việt Nam là một trong những công ty hàng đầu Việt Nam trong lĩnh vực nhà thông minh (Smarthome). Sở hữu công nghệ lõi và năng lực nghiên cứu nổi trội, LUMI đã và đang khẳng định vị thế tiên phong dẫn đầu thị trường với những cải tiến sản phẩm dẫn đầu xu hướng, có mặt tại hơn 150 nhà phân phối trên 63 tỉnh – thành và xuất khẩu ra nhiều quốc gia trên thế giới.....	
<b>1.2 Tầm nhìn – Sứ mệnh – Giá trị cốt lõi.....</b>	<b>6</b>
Tiên phong mang đến những giải pháp sống tiện nghi vượt trội và an toàn cho khách hàng, vận hành hiệu quả và hợp tác cùng phát triển bền vững với các bên đối tác song song với lan toả những nghĩa cử tích cực đến cộng đồng. ....	
<b>1.3 Cơ cấu tổ chức .....</b>	<b>6</b>
<b>CHƯƠNG 2. NỘI DUNG THỰC TẬP .....</b>	<b>7</b>
<b>2.1 Các vị trí công việc thực tập trong công ty .....</b>	<b>7</b>
<b>2.2 Vị trí lập trình App .....</b>	<b>7</b>
<b>2.3 Chi tiết tìm hiểu về Flutter .....</b>	<b>7</b>
<b>2.4 Xây dựng UI theo Figma .....</b>	<b>12</b>
<b>CHƯƠNG 3. Nhận xét, đề xuất về kỳ thực tập .....</b>	<b>20</b>
<b>3.1 Ưu điểm .....</b>	<b>20</b>
<b>3.2 Nhược điểm .....</b>	<b>20</b>
<b>3.3 Đề xuất.....</b>	<b>20</b>
<b>KẾT LUẬN .....</b>	<b>21</b>
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>22</b>

## **DANH MỤC HÌNH VẼ**

Hình 2.4.1 Menu Screen.....	13
Hình 2.4.2 Home Screen .....	13
Hình 2.4.3 Product Detail Screen.....	14
Hình 2.4.4 Favourite Screen.....	15
Hình 2.4.5 Cart Screen .....	16
Hình 2.4.6 Checkout Screen.....	17
Hình 2.4.7 Edit Profile Screen .....	18
Hình 2.4.8 Profile Screen .....	18
Hình 2.4.9 Login Screen .....	19

# **CHƯƠNG 1. GIỚI THIỆU CHỨC NĂNG, NHIỆM VỤ, CƠ CẤU TỔ CHỨC CỦA ĐƠN VỊ TIẾP NHẬN**

## **1.1 Giới thiệu chung**

Công ty cổ phần LUMI Việt Nam là một trong những công ty hàng đầu Việt Nam trong lĩnh vực nhà thông minh (Smarthome). Sở hữu công nghệ lõi và năng lực nghiên cứu nổi trội, LUMI đã và đang khẳng định vị thế tiên phong dẫn đầu thị trường với những cải tiến sản phẩm dẫn đầu xu hướng, có mặt tại hơn 150 nhà phân phối trên 63 tỉnh – thành và xuất khẩu ra nhiều quốc gia trên thế giới.

## **1.2 Tầm nhìn – Sứ mệnh – Giá trị cốt lõi**

### **1.2.1 Tầm nhìn**

LUMI Việt Nam có tầm nhìn khẳng định vị thế Thương hiệu nhà thông minh số 1 Việt Nam tại thị trường trong nước, đưa công nghệ Việt Nam vươn xa thị trường Thế giới.

### **1.2.2 Sứ mệnh**

Tiên phong mang đến những giải pháp sống tiện nghi vượt trội và an toàn cho khách hàng, vận hành hiệu quả và hợp tác cùng phát triển bền vững với các bên đối tác song song với lan toả những nghĩa cử tích cực đến cộng đồng.

### **1.2.3 Giá trị cốt lõi**

- Cam kết chất lượng: Chất lượng được nhận định là sống còn với Nhà thông minh LUMI Việt Nam.
- Không ngừng sáng tạo: Liên tục học hỏi và nắm bắt những thành tựu Khoa học – Công nghệ ứng dụng trên thế giới để áp dụng sáng tạo sản phẩm.
- Tự chủ nghiên cứu: Làm chủ công tác nghiên cứu và xây dựng đội ngũ kỹ sư tài năng đón đầu thị trường công nghệ.

## **1.3 Cơ cấu tổ chức**

### **1.3.1 Văn hoá doanh nghiệp**

- Chú trọng phát triển năng lực chuyên môn từng cá nhân
- Lấy thách thức làm động lực, lấy thất bại làm bài học để trưởng thành
- Hoạt động trên tinh thần tương hỗ - chia sẻ

### **1.3.2 Đối tác hợp tác**

- Đối tác cung cấp linh kiện: Samsung, Silicon Labs, Panasonic, Vishay,...
- Đối tác hợp tác kinh doanh: Viettel, FPT,...

## CHƯƠNG 2. NỘI DUNG THỰC TẬP

### 2.1 Các vị trí công việc thực tập trong công ty

Công ty LUMI Việt Nam là một công ty về Smarthome. LUMI có nhiều sản phẩm tiêu biểu như Chiếu sáng thông minh, Camera AI, các thiết bị thông minh như Đèn điều khiển bằng giọng nói, Điều khiển rèm thông minh, Hệ thống điều hoà – Tivi,...

Tại thời điểm bản thân đăng ký thực tập tại công ty, có những vị trí thực tập:

- Lập trình nhúng
- Lập trình App
- AI
- Hardware

Mỗi vị trí thực tập có những nội dung thực tập riêng biệt phù hợp.

### 2.2 Vị trí lập trình App

Đối với lập trình App, nội dung được giao là tìm hiểu về cách xây dựng UI bằng Framework Flutter, cụ thể:

- Tìm hiểu về Ngôn ngữ Dart
- Tìm hiểu về Lập trình hướng đối tượng
- Tìm hiểu về UI Materials
- Tìm hiểu về State, Stateful lifecycle
- Tìm hiểu về Navigation

Sau đó, xây dựng UI của một E-commerce App trên Figma.

### 2.3 Chi tiết tìm hiểu về Flutter

#### 2.3.1 Ngôn ngữ Dart

Dart là ngôn ngữ lập trình mới, được phát triển bởi Google, hiện đã được chấp thuận bởi tổ chức Ecma. Dart được sử dụng để xây dựng các loại ứng dụng: web, server, di động (IOS và Android với công cụ Flutter).

Dart là ngôn ngữ hỗ trợ lập trình hướng đối tượng, cú pháp kiểu C, mã code Dart có thể biên dịch thành JavaScript để chạy trên trình duyệt. Nó hỗ trợ những khái niệm lập trình hiện đại như giao diện lớp, lớp trừu tượng ...

Từ ngôn ngữ lập trình Dart, Google giới thiệu SDK FFlutter để sử dụng ngôn ngữ Dart phát triển ứng dụng di động chạy đa nền tảng (Một lần viết code build ra ứng dụng cho IOS và Android).

Các kiến thức về Dart cơ bản:

- Kiểu dữ liệu cơ bản:
  - `int` : số nguyên

- double: số thực
- char : ký tự
- String : chuỗi ký tự
- bool : boolean
- List : dữ liệu kiểu Array
- Set : dữ liệu kiểu Array nhưng không trùng phần tử đã có
- Map : chứa các key – value pair
- null : null
- Functions:
  - Parameters
  - Anonymous functions
  - Return values
- Operators:
  - Arithmetic operators: '+', '-', '\*', '/', '='
  - Equality and relational operators: '==', '!=', '>', '<', '>=', '<='
  - Typetest operators: as, is, is!
  - Logic operators: '||', '&&'
- Control flow statements:
  - Loops: for loops, while và do – while loop, break, continue
  - Branches: if – else, switch – case,
- Exceptions:
  - Throw
  - Catch
  - Try
  - Finally
- Asynchronous programming:
  - async – await
  - Future
  - Stream

### 2.3.2 Lập trình hướng đối tượng

Dart là ngôn ngữ hướng đối tượng.

Các chủ đề của lập trình hướng đối tượng:

- Constructors: Có nhiều dạng Constructor. Mặc định, Constructor được khai báo sẵn (Default Constructor), nhưng thông thường sẽ sử dụng Named Constructor để có thể điều chỉnh Constructor phù hợp với nhu cầu.
- Methods: Getter – Setter, Method function.



- Abstract classes: Sử dụng Abstract và Abstract Method để kế thừa với các Class khác.
- Extending: Class có thể Extends một class khác (Inheritance).
- Enumerate: enums là một loại class đại diện cho fixed number của constants value, mỗi value ứng với một index getter.

### 2.3.3 UI Materials

Trong Flutter, hầu hết mọi thứ là Widget. Flutter widgets được lấy cảm hứng từ React, và người dùng xây dựng UI với các Widgets. Widgets miêu tả hình ảnh của chúng sẽ thế nào dựa vào sự điều chỉnh và trạng thái của chính nó.

Widget Tree: Widget Tree là một cây chứa tất cả các Widget mà Project sử dụng. Một Widget Tree hoàn toàn do Lập trình viên kiểm soát. Lập trình viên khai báo các Widget và lồng ghép lại với nhau để tạo nên bố cục mong muốn.

- Basic widgets:
  1. Text: chứa 1 dòng chữ
  2. Container: padding, margins, decoration và 1 child widget
  3. Center: chứa 1 child widget, đặt child ở chính giữa
  4. Image: chứa 1 ảnh
  5. Icon: chứa 1 icon
  6. ElevatedButton: 1 nút có thể ấn vào
  7. Scaffold: khung của 1 màn hình layout, chứa appbar, body, bottom
- Multi-child widgets:
  - Stack: chồng và đặt vị trí các widgets
  - Row: chứa các widget theo hàng ngang
  - Column: chứa các widget theo hàng dọc
  - ListView: chứa 1 list các widget nối tiếp nhau, theo 1 scroll direction
  - GridView: giống với ListView nhưng chứa các widget theo các hàng, cột với số lượng nhất định
- Assets, media: Flutter sử dụng file pubspec.yaml file ở gốc của project, có nhiệm vụ định dạng project, chứa các địa chỉ assets, media và các packages mà project sử dụng. Khi điền các địa chỉ file assets, media mà lập trình viên muốn sử dụng, lập trình viên có thể sử dụng được các assets đó để lập trình.
- Gestures: Gestures đại diện cho các cử chỉ chạm, ấn, kéo,... được nhận ra bởi những event. Có những gestures cơ bản như onTap, onPressed, onDoubleTap, onLongPress, onVerticalDragStart,... Ta có thể thêm những gesture mong muốn bằng cách lồng Widget GestureDetector và gesture cần thêm lên Widget bản thân muốn sử dụng. Ngoài ra, có những Widget có sẵn các gestures như các Buttons.

- Forms, vùng điền Form: Trong Flutter, có hai Widget giúp ta điền Form chính là TextField, TextFormField.

#### 2.3.4 State, State Lifecycle

State trong tiếng Anh là trạng thái. Trong Flutter, State là trạng thái của một thứ gì đó trong bộ nhớ, ví dụ như một Widget, một Screen. Khi State thay đổi, Flutter sẽ tự kích hoạt Redraw đối với UI của thứ đó và xây dựng lại UI với State mới.

Đối với Widget, có 2 loại Widget là Stateless Widget và Stateful Widget:

- Stateless Widget: Stateless Widget giống với tên gọi của nó, một widget không có State, chỉ nhận dữ liệu và hiển thị dữ liệu thụ động. Vì thế, một Stateless Widget là một Widget tĩnh, không Rebuild lại UI và sẽ không thay đổi trong toàn bộ quá trình sử dụng UI.
- Stateful Widget: Stateful Widget là một Widget có State. Một Stateful Widget có một State Lifecycle như sau:
  1. createState(): Khi Flutter xây dựng StatefulWidget, nó sẽ ngay lập tức gọi createState() đầu tiên nhằm khởi tạo State cho Widget.
  2. mounted == true: Tất cả các Widget đều có thuộc tính this.mounted. Khi một Widget có State của nó, Flutter sẽ gán State của Widget với BuildContext, sau đó Widget sẽ không bao giờ cần khởi tạo lại State và nó sẽ trả về giá trị true với thuộc tính this.mounted( khi mounted == false sẽ không thể gọi setState để thay đổi State của Widget).
  3. initState(): Đây là phương thức đầu tiên được gọi khi Widget được khởi tạo( ở 2 bước kia, Widget vẫn chưa được tạo ra trên Widgets Tree). InitState chỉ được gọi một lần, ta có thể can thiệp vào initState để khởi tạo giá trị cho state với @override và super.initState().
  4. didChangeDependencies(): Phương thức didChangeDependencies được gọi ngay sau khi initState cho lần đầu tiên widget được xây dựng. Nó cũng sẽ được gọi lại bất cứ khi nào một đối tượng mà Widget này phụ thuộc vào được gọi build.
  5. build(): Phương thức này được gọi thường xuyên để render Widget. Đây chính là hàm chứa nội dung của Widget.
  6. didUpdateWidget(): Hàm này được gọi nếu Widget được thay đổi và cần phải build lại.
  7. setState(): Phương thức setState() này thường được Lập trình viên sử dụng để thay đổi State của Widget. Lập trình viên có thể gọi hàm setState() này và gán giá trị mới cho State. Khi sử dụng, setState sẽ thông báo cho Widget rằng State của Widget sẽ thay đổi và Widget cần phải build lại để State được cập nhật.

8. `deactivate()`: `deactivate()` được gọi khi State được xoá khỏi Widgets Tree, nhưng điều này có thể được xác nhận lại trước khi quá trình xoá kết thúc. Phương thức này tồn tại về cơ bản là vì các State có thể được di chuyển từ Tree này sang Tree khác.
9. `dispose()`: `dispose()` được gọi khi đối tượng State bị xoá vĩnh viễn.
10. `mounted == false`: Điều chỉnh `mounted` về `false`.

Quản lý State: State của một ứng dụng chính là hầu hết mọi thứ được lưu trong bộ nhớ khi ứng dụng đó đang chạy, bao gồm Assets, các biến của UI, State của Animation, fonts chữ,... Lập trình viên không quản lý một vài State mà Flutter sẽ làm việc đó, ví dụ như Texture States. Những State mà Lập trình viên cần quản lý sẽ có 2 loại: Ephemeral State và App State.

- Ephemeral State: Ephemeral State hay còn gọi là UI State, Local State là State của chính một Widget. Ví dụ điển hình cho Ephemeral State là State ở trong một Stateful Widget. Tuy nhiên, Ephemeral State chỉ cho phép Widget thay đổi State của chính nó.
- App State: App State chính là những State mà App sử dụng xuyên suốt các thành phần của App, ví dụ như thông tin người dùng, trạng thái đăng nhập,... Khác với việc sử dụng `setState` như Ephemeral State, có nhiều sự lựa chọn cho Lập trình viên để quản lý App State.

Không có một quy luật bắt buộc nào đối với ứng dụng và Lập trình viên có thể sử dụng hoàn toàn `setState()` để quản lý tất cả các State của App. Tuy nhiên, đối với những App lớn, có nhiều State, việc quản lý sẽ dần trở nên khó khăn và Lập trình viên nên sử dụng App State.

### 2.3.5 Navigation

Navigation, với tên gọi của nó, chính là điều hướng. Mọi ứng dụng đều có các trang riêng biệt, và người dùng có thể điều hướng để chuyển sang những trang khác nhau. Trong Flutter, những màn hình được sắp xếp theo một Stack để Flutter có thể nhận biết màn hình trước và sau. Có nhiều cách để Lập trình viên điều hướng các màn hình:

- Navigator: Để điều hướng một màn hình, Lập trình viên có thể sử dụng `push()` để đẩy màn hình mới lên Navigator Stack hoặc `pop()` để pop màn hình hiện tại khỏi Navigator Stack()
- NamedRoute: Có thể đăng ký trước các màn hình ở Routes và khi `push`, ta sử dụng Name ứng với các màn hình mà mình đã đăng ký trước đó. Đây không phải là cách Navigate được khuyến dùng.

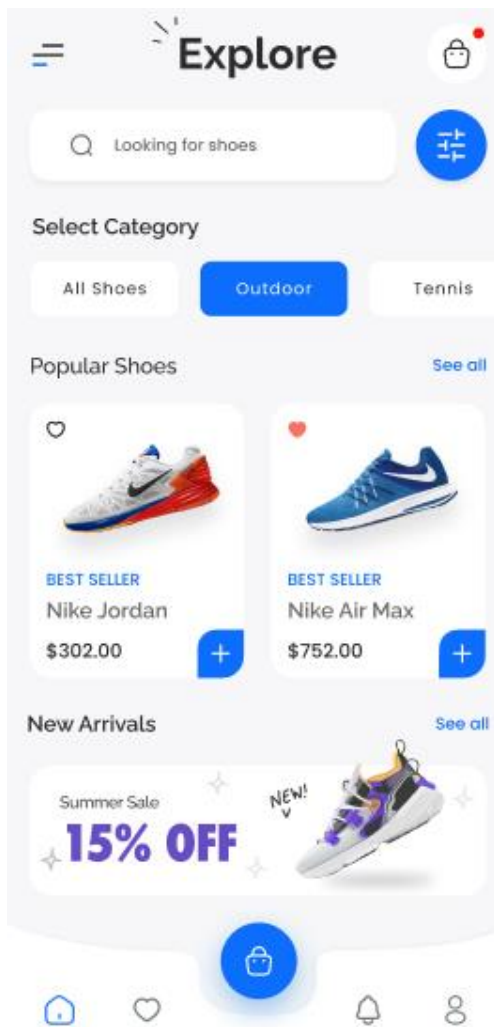
- Router: Có những package routing, navigating nổi tiếng và nâng cao như go\_router được Flutter tin dùng. Go\_router cung cấp thêm nhiều tiện ích như query, parsing path cho Deep Linking; Redirection; ShellRoute,...

Lập trình viên còn có thể sử dụng kết hợp giữa Router và Navigator. Khi sử dụng go\_router, các route ở trong Navigator sẽ là page-backed còn đối với Navigator.push sẽ là pageless. Đối với page-backed route, khi route bị thu hồi khỏi Navigator, những pageless routes đằng sau nó cũng sẽ bị thu hồi cho đến tận page-backed route gần nhất.

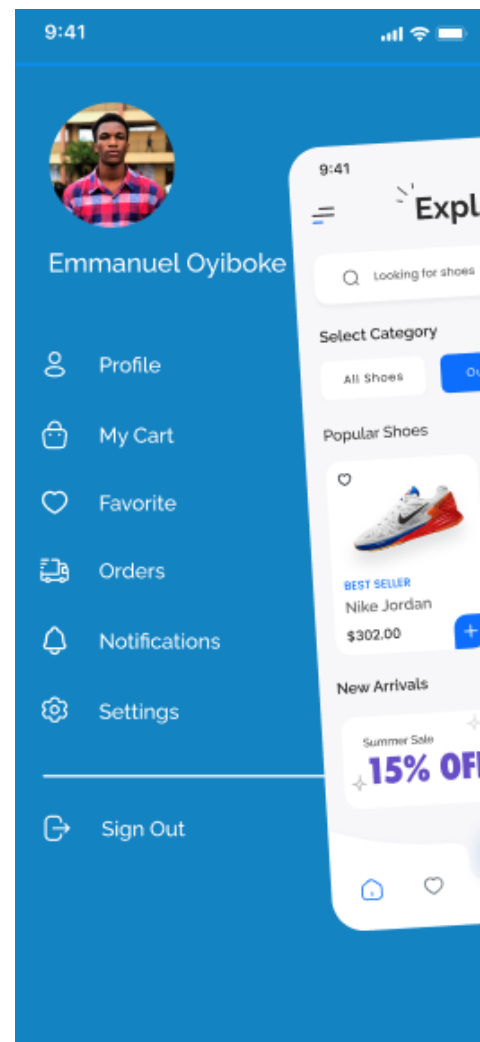
## 2.4 Xây dựng UI theo Figma

Link Figma: [https://www.figma.com/file/yxsaT6vfbDaVBuoAEaf17u/Nike-Sneaker-Store-App-\(Community\)?type=design&node-id=43-6458&mode=design](https://www.figma.com/file/yxsaT6vfbDaVBuoAEaf17u/Nike-Sneaker-Store-App-(Community)?type=design&node-id=43-6458&mode=design)

Phân tích UI: Đầu tiên, ta sẽ phân tích bố cục UI từng Screen



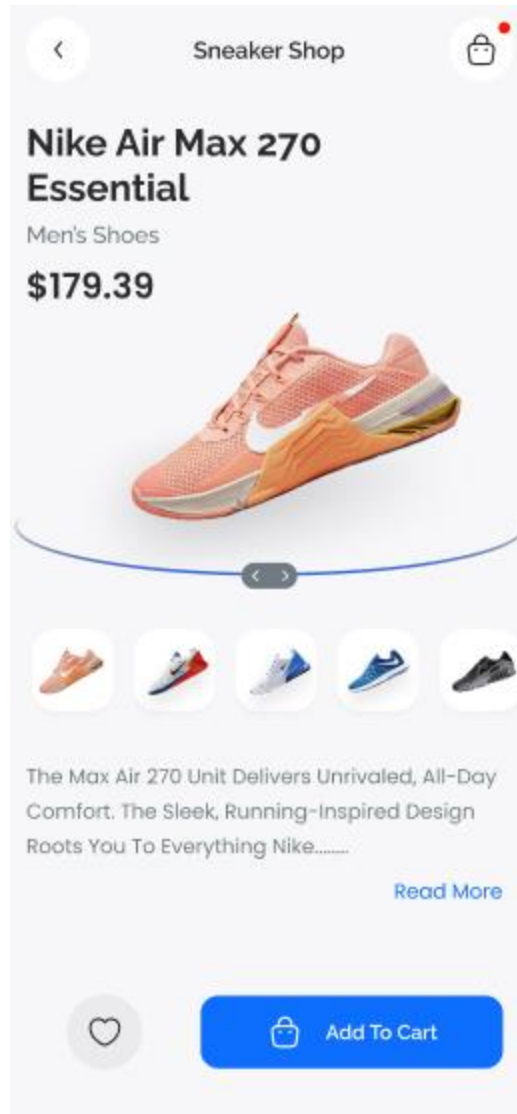
Hình 2.4.2 Home Screen



Hình 2.4.1 Menu Screen

Đầu tiên, trên Scaffold sẽ có AppBar, Body và BottomNavigationBar:

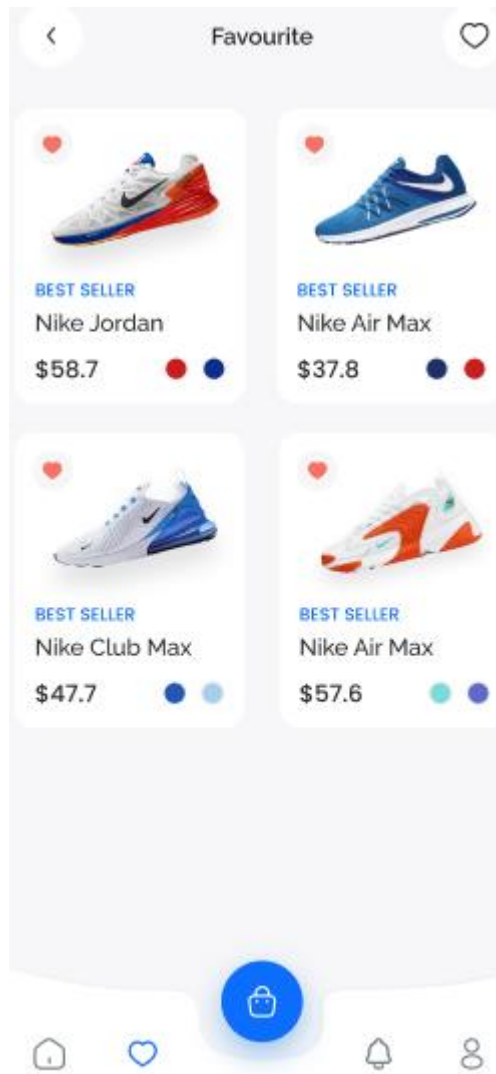
- AppBar: Leading có 1 Menu Icon Button, ở giữa có Title 'Explore' và action có Shopping Bag Button chứa Badge tròn màu đỏ để Notification người dùng có sản phẩm trong giỏ hàng. Khi ấn vào Menu Button sẽ ra Menu Screen.
- Body: Body là 1 Column gồm:
  1. Row có Search TextField và Hamburger Icon Button
  2. Dòng Text 'Category'
  3. ListView nằm ngang gồm các Category
  4. Row có Text 'Popular Shoes' và Text Button 'See all'
  5. List View các sản phẩm
  6. Row có Text 'New Arrivals' và Text Button 'See all'
  7. Banner quảng cáo
- BottomNavigationBar: BottomNavBar có 4 IconButton và 1 FloatingActionButton ở giữa.



Hình 2.4.3 Product Detail Screen

Tiếp theo là ProductDetailScreen:

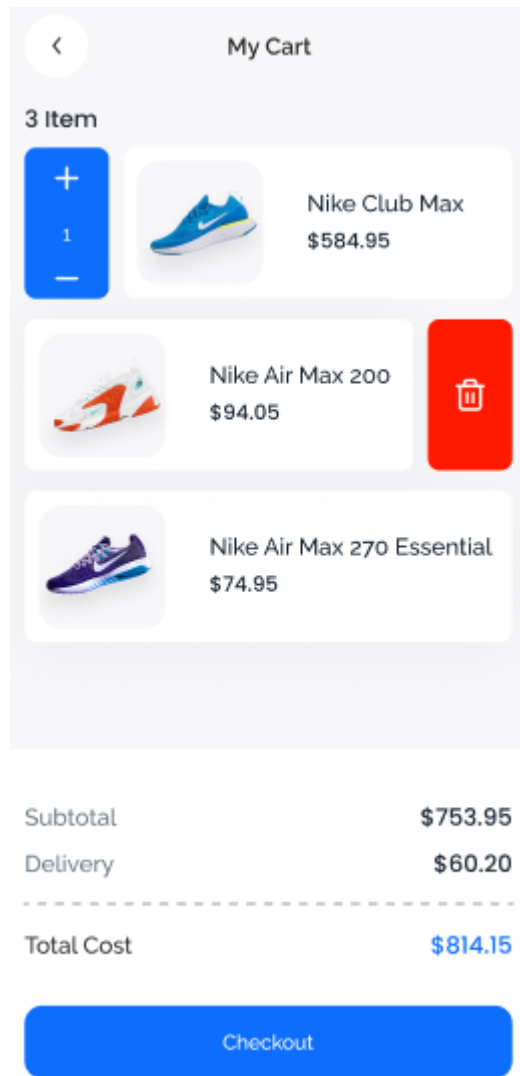
- AppBar: Leading với BackButton, title 'Sneaker Shop' ở giữa và action Shopping Bag Button
- Body: Body là 1 Column với:
  1. Dòng Text title 'Product Name'
  2. Dòng Text category
  3. Dòng Text price
  4. Ảnh của sản phẩm
  5. ListView các sản phẩm khác
  6. Dòng Text miêu tả sản phẩm
  7. Text Button 'Read More'
  8. Row với Favourite Icon Button và 'Add To Cart' Button



Hình 2.4.4 Favourite Screen

Với Favourite Screen:

- AppBar: Leading với Back Button, title 'Favourite' và action Favourite Button
- Body: GridView các sản phẩm Favourite
- BottomNavigationBar: giống với Home Screen

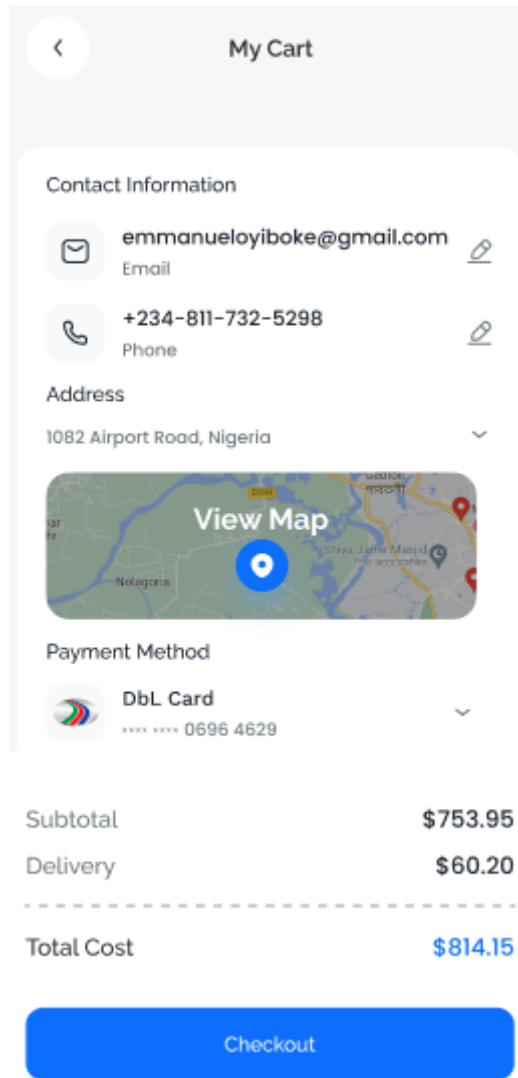


Hình 2.4.5 Cart Screen

Cart Screen:

- AppBar: Back Button và title 'My Cart'
- Body: Column với:
  1. Text số lượng item trong Cart
  2. ListTile đối với từng item
  3. Dòng Text tính tiền các item
  4. Dòng Text tính phí delivery
  5. Dòng Text tính tổng chi phí
  6. 'Checkout' button

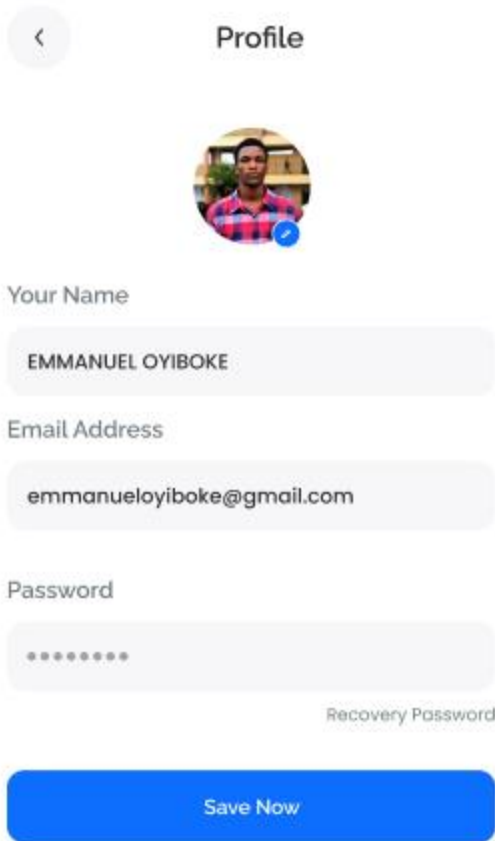




Hình 2.4.6 Checkout Screen

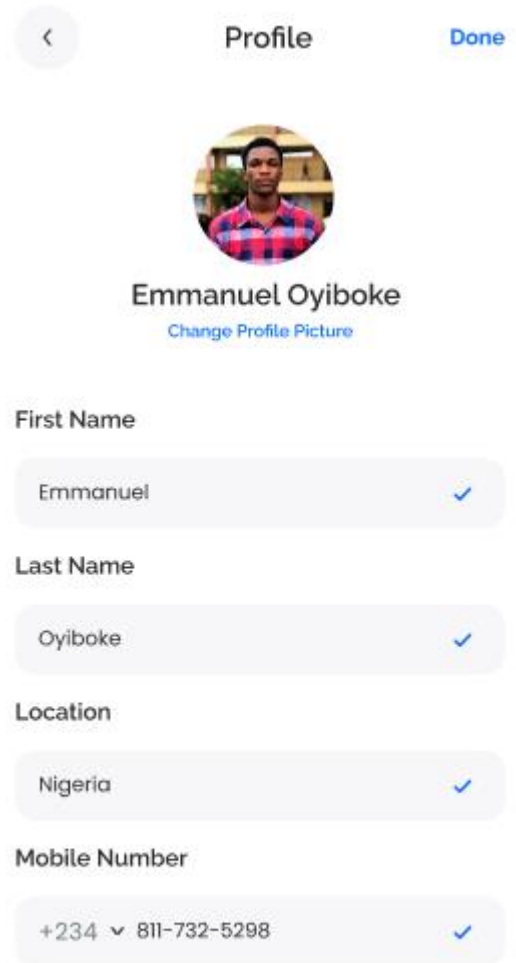
Checkout Screen:

- AppBar: Back Button, title 'My Cart'
- Body: Column với:
  1. Dòng Text 'Contact Information'
  2. TextField Email
  3. TextField PhoneNumber
  4. Dòng Text 'Address'
  5. TextField Address
  6. Map
  7. Dòng Text 'Payment Method'
  8. Phần Cost và Button như Cart Screen



The Profile Screen features a back button in the top left corner. The title 'Profile' is centered at the top. Below the title is a circular profile picture of a man in a red and blue plaid shirt, with a blue checkmark at the bottom right. Underneath the picture, the text 'Your Name' is followed by a text field containing 'EMMANUEL OYIBOKE'. Below this, 'Email Address' is followed by a text field containing 'emmanueloyiboke@gmail.com'. Then, 'Password' is followed by a text field with masked characters '.....'. A 'Recovery Password' link is positioned below the password field. At the bottom, there is a prominent blue 'Save Now' button.

Hình 2.4.8 Profile Screen

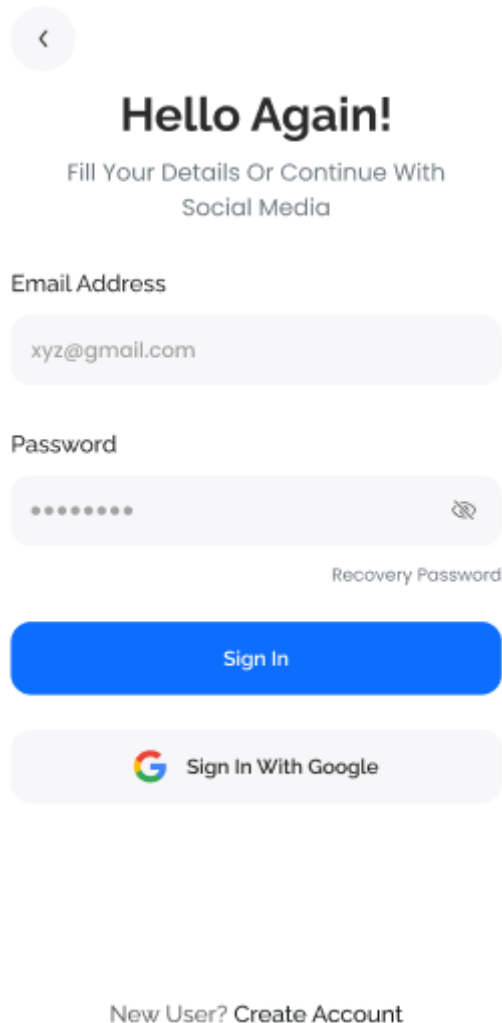


The Edit Profile Screen has a back button in the top left corner. The title 'Profile' is centered at the top, and a 'Done' button is in the top right corner. Below the title is the same circular profile picture of the man in the plaid shirt. Underneath the picture, the text 'Emmanuel Oyiboke' is displayed, with a 'Change Profile Picture' link below it. The screen contains several form fields: 'First Name' with 'Emmanuel', 'Last Name' with 'Oyiboke', 'Location' with 'Nigeria', and 'Mobile Number' with '+234 811-732-5298'. Each of these fields has a blue checkmark at the end, indicating they are valid or confirmed.

Hình 2.4.7 Edit Profile Screen

Profile Screen và Edit Profile Screen khá giống nhau:

- AppBar: Back Button, title 'Profile' và 'Done' Button với Edit Profile Screen
- Body: Column với:
  1. Circle Avatar
  2. Dòng Text User Name với Edit Profile Screen
  3. Các Title và TextField về User
  4. 'Recovery' Text Button cho Profile Screen
  5. 'Save Now' Button cho Profile Screen



The image shows a mobile login screen. At the top left is a back arrow button. The main heading is 'Hello Again!' in bold. Below it is the subtitle 'Fill Your Details Or Continue With Social Media'. There are two input fields: 'Email Address' with the placeholder 'xyz@gmail.com' and 'Password' with masked characters and a toggle icon. Below the password field is a link 'Recovery Password'. There are two buttons: a blue 'Sign In' button and a white 'Sign In With Google' button with the Google logo. At the bottom is a link 'New User? Create Account'.

Hình 2.4.9 Login Screen

Cuối cùng với Login Screen:

- AppBar: Back Button
- Body: Column với:
  1. Title 'Hello Again!' và 'Fill Your Detail Or Continue With Social Media'
  2. TextField và Title 'Email Address'
  3. TextField và Title 'Password'
  4. 2 Button 'Sign In'
  5. Dòng 'New User?' và Text Button 'Create Account'

### **CHƯƠNG 3. Nhận xét, đề xuất về kỳ thực tập**

#### **3.1 Ưu điểm**

Có cấu trúc học và ôn tập rõ ràng để sinh viên có thể học theo từ ban đầu

Được học tập và trải nghiệm các ngành nghề trong tương lai, được kết nối với các anh chị đi trước

#### **3.2 Nhược điểm**

Thời gian thực tập khá ngắn, không thể học đủ hay thực hành đầy đủ kiến thức

#### **3.3 Đề xuất**

Đối với thời gian thực tập quá ngắn, nên tăng thời gian thực tập để đảm bảo đủ sinh viên có thể ôn tập tốt hơn.

## KẾT LUẬN

Kỳ thực tập vừa qua là một trong những trải nghiệm quý báu sẽ giúp đỡ em rất nhiều trong khoảng thời gian tiếp theo khi bước vào quãng thời gian học tập cuối cùng và làm việc. Trong kỳ học vừa qua, em đã được học nhiều về chuyên môn của ngành nghề, trải nghiệm về môi trường doanh nghiệp, rèn luyện các kỹ năng chuyên ngành và kỹ năng mềm. Trải nghiệm về nghề App Developer cũng như học những kiến thức về Flutter đều rất đáng giá, khiến em muốn học và tìm hiểu thêm về ngành nghề này trong tương lai. Em xin cảm ơn nhà trường đã tạo điều kiện để cho sinh viên có những kiến thức và trải nghiệm quan trọng cho tương lai, cảm ơn Công ty LUMI Việt Nam đã nhiệt tình giúp đỡ, chỉ bảo em và các bạn sinh viên tận tình, chuyên nghiệp.

## TÀI LIỆU THAM KHẢO

*Flutter documentation.* (n.d.). Flutter. <https://docs.flutter.dev/>

*Dart documentation.* (n.d.). Dart. <https://dart.dev/guides>