**TRƯỜNG CAO ĐẲNG CÔNG NGHỆ THỦ ĐỨC**



**KHOA CÔNG NGHỆ THÔNG TIN**

**🙜🕮🙞**

****

**BÁO CÁO MÔN HỌC**

**Lập trình di động 3**

**Giảng viên hướng dẫn:** Trương Bá Thái

**Nhóm sinh viên thực hiện**

* + - * Nguyễn Văn An 17211TT3235
      * Nguyễn Công Nhân 17211TT3350
      * Phạm Văn Nhân 17211TT3073

*Tp. Hồ Chí Minh*, ngày 17 tháng 10 năm 2019

**LỜI NÓI ĐẦU**

Với cuộc sống công nghệ ngày càng phát triển như hiện nay, những bé thú cưng đang dần chiếm lại vị thế của mình. Nhưng song song với đó việc mua bán những bé thú cưng lại rất cổ điển và chưa đáp ứng được nhu cầu mà người tiêu dùng mong muốn.

Cũng chính vì xu hướng phát triển như vậy, nhóm em đã quyết định chọn xây app PetShop đem đến cho mọi người sự tiện lợi nhất để thỏa mãn tình yêu của mình với những bé thú cưng.

Dù đã cố gắng hoàn thành đồ án đúng yêu cầu, nhưng do thời gian có hạn và khả năng còn hạn chế nên chắc chắn sẽ không tránh khỏi sai sót, nhóm em rất mong nhận được ý kiến đóng góp của thầy để đồ án được hoàn thiện hơn.

Sau cùng, nhóm chúng em xin gửi lời cảm ơn chân thành đến thầy Tiêu Kim Cương. Trong quá trình nghiên cứu đề tài, thầy đã tạo điều kiện cung cấp những kiến thức liên quan, tận tình hướng dẫn nhóm cũng như tạo mọi điều kiện thuận lợi để chúng em có thể hoàn thành tốt đề tài.

**Trân Trọng**

Table of Contents

[CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI 5](#_Toc16422)

[1. Giới thiệu 5](#_Toc14808)

[a) Mô tả 5](#_Toc22177)

[b) Phạm vi 5](#_Toc31956)

[2. Chức năng yêu cầu 5](#_Toc1541)

[a) Tính năng/ Thành phần #1: Màn hình đăng nhập 5](#_Toc25292)

[i. Giao diện người dùng 6](#_Toc24662)

[ii. Yêu cầu chức năng 6](#_Toc30022)

[b) Tính năng/ Thành phần #2: Màn hình đăng ký 8](#_Toc29242)

[i. Giao diện người dùng 8](#_Toc23233)

[ii. Yêu cầu chức năng 9](#_Toc2809)

[c) Tính năng/ Thành phần #3: Màn hình Home 11](#_Toc25882)

[i. Giao diện người dùng 11](#_Toc24005)

[ii. Yêu cầu chức năng 12](#_Toc18505)

[d) Tính năng/ Thành phần #4: Màn hình Navigation Menu 14](#_Toc3619)

[i. Giao diện người dùng 14](#_Toc25495)

[ii. Yêu cầu chức năng 15](#_Toc2209)

[Button Lịch sử mua hàng 15](#_Toc6661)

[Button Thay đổi thông tin 15](#_Toc21058)

[e) Tính năng/ Thành phần #5: Màn hình lịch sử mua hàng 16](#_Toc23171)

[i. Giao diện người dùng 16](#_Toc3143)

[ii. Yêu cầu chức năng 16](#_Toc16152)

[Button Back 17](#_Toc5948)

[Textview lịch sử mua hàng 17](#_Toc8762)

[f) Tính năng/ Thành phần #6: Màn hình giỏ hàng 18](#_Toc22387)

[i. Giao diện người dùng 18](#_Toc17372)

[ii. Yêu cầu chức năng 19](#_Toc28245)

[ButtonImage Cart 19](#_Toc9217)

[ButtonImage Contact 19](#_Toc5930)

[g) Tính năng/ Thành phần #7: Màn hình Search 20](#_Toc9940)

[i. Giao diện người dùng 20](#_Toc2898)

[ii. Yêu cầu chức năng 21](#_Toc30359)

[ButtonImage Cart 21](#_Toc12540)

[ButtonImage Contact 21](#_Toc31202)

[h) Tính năng/ Thành phần #8: Màn hình Contact 22](#_Toc12510)

[i. Giao diện người dùng 22](#_Toc10509)

[ii. Yêu cầu chức năng 24](#_Toc12935)

[ButtonImage Cart 24](#_Toc9146)

[ButtonImage Contact 24](#_Toc15406)

[CHƯƠNG II: BÀI TẬP Ở LỚP 26](#_Toc12248)

[3. Phạm Văn Nhân 26](#_Toc23100)

[4. Nguyễn Công Nhân 40](#_Toc21911)

[5. Nguyễn Văn An 41](#_Toc5089)

[CHƯƠNG III: TÌM HIỂU CÁC COMPONEMT 42](#_Toc10277)

[PROPS 42](#_Toc7526)

[STATE 45](#_Toc2349)

[Check Box 47](#_Toc32298)

[STYLE 50](#_Toc25964)

[Layout with Flexbox 52](#_Toc20327)

[Handling Text Input 53](#_Toc6878)

[Handling Touches 54](#_Toc8752)

[Using a ScrollView 57](#_Toc32072)

[Progressbar 60](#_Toc20037)

[ListView 63](#_Toc21552)

[Picker 66](#_Toc14517)

[Switch Button 68](#_Toc17798)

[CHƯƠNG IV: CHI TIẾT ỨNG DỤNG VÀ CÔNG VIỆC CỦA MỖI THÀNH VIÊN 70](#_Toc22630)

[1. Phạm Văn Nhân 70](#_Toc31359)

[2. Nguyễn Công Nhân 76](#_Toc12278)

[3. Nguyễn Văn An 89](#_Toc30843)

[CHƯƠNG V: KẾT LUẬN KIẾN NGHỊ 104](#_Toc28568)

# **CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI**

Kỷ nguyên công nghệ 4.0 đã mở ra cùng với sự phát triển không ngừng của công nghệ thông tin, kinh doanh đồ uống online đã và đang là một hướng đi khá mới mẻ F&B, ngày càn có nhiều nhà hàng, quán cafe tập chung vào phát triển các kênh bán hàng online qua web cũng như qua ứng dụng trên điện thoại.

Ứng dụng ĐỒ UỐNG ONLINE là một ứng dụng bán các loại đồ uống online nơi mà mọi người có thể tìm, lựa chọn và mua cho mình những loại đồ uống mà mình thích khi không thể đến trực tiếp cửa hàng để mua. Bạn chỉ cần ở nhà và mở ứng dụng ĐỒ UỐNG ONLINE, đồ uống sẽ được giao đến nhà của bạn.

1. **Giới thiệu**
   1. **Mô tả**

Ứng dụng cho phép người dùng đặt đồ uống online

* 1. **Phạm vi**

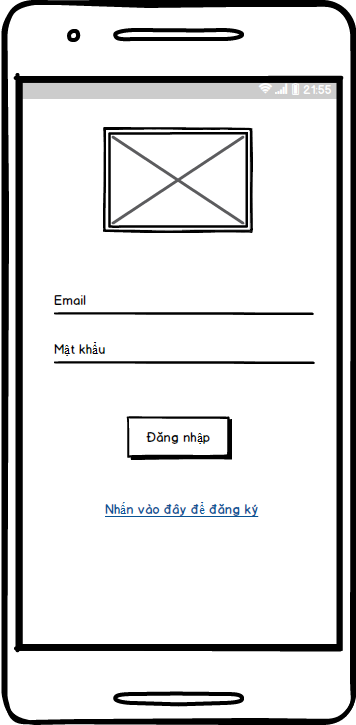
Phiên bản demo

Phiên bản demo chỉ dành cho mục đích hiển thị, không liên kết đến các trang web liên kết Android OS: Android: OS 5.0.X đến 10.X.X

* Android test devices: Pixel 2

Android screen resolutions: 480x800, 720x1280

1. **Chức năng yêu cầu**
   1. **Tính năng/ Thành phần #1: Màn hình đăng nhập**
      1. **Giao diện người dùng**

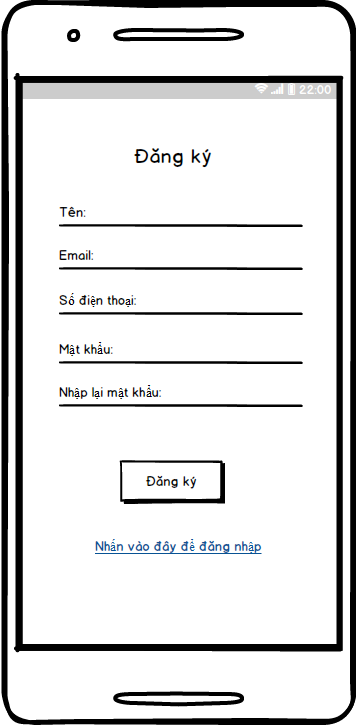


*Image1: Đăng nhập*

* + 1. **Yêu cầu chức năng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình đăng nhập** | Màn hinh đăng nhập gồm có:   * Logo * EditText Email * EditText Mật khẩu * Button Đăng Nhập * TextView Nhấn vào đây để đăng ký | N/A | N/A |
| **EditText**  **Email** | Cho phép người dùng nhập Email | Chạm vào EditText Email | Hiển thị con trỏ chuột vào ô email cho phép người dùng nhập |
| **EditText**  **Mật khẩu** | Cho phép người dùng nhập Password | Chạm vào EditText Password | Hiển thị con trỏ chuột vào ô password cho phép người dùng nhập |
| **Button Đăng nhập** | Cho phép người dùng đăng nhập vào ứng dụng | Chạm vào nút đăng nhập | 1. Lấy dữ liệu từ EditText Email và EditText Mật khẩu 2. So sánh dữ liệu vừa lấy với dữ liệu trong Database 3. Nếu trùng khớp thì sẽ chuyển hướng đến màn hình “Home”.   Nếu không trùng khớp thì sẽ hiển thị thông báo đăng nhập không thành công |
| **TextView**  **Nhấn vào đây để đăng ký** | Cho phép người dùng chuyển đến màn hình đăng ký tài khoản | Chạm vào edittext Chạm vào đây để đăng ký | Hiển thị màn hình đăng ký tài khoản |

* 1. **Tính năng/ Thành phần #2: Màn hình đăng ký**
     1. **Giao diện người dùng**



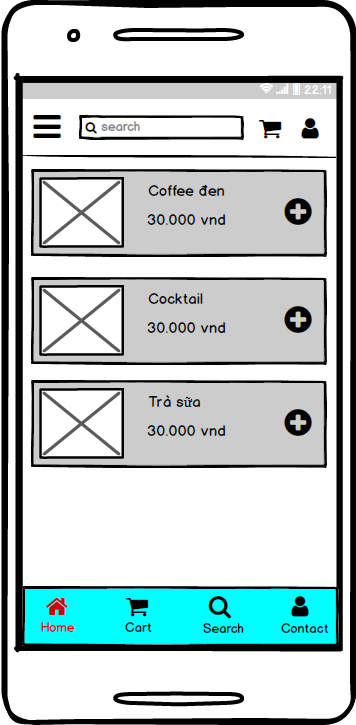
*Image2: Đăng ký*

* + 1. **Yêu cầu chức năng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình đăng ký** | Màn hình đăng ký gồm có:   * Textview Đăng ký * Edittext Tên * Edittext Email * Edittext Số điện thoại * Edittext Mật khẩu * Edittext Nhập lại mật khẩu * Button Đăng ký * TextView Nhấn vào đây để đăng nhập | N/A | N/A |
| **EditText**  **Tên** | Cho phép người dùng nhập Tên | Chạm vào EditText Tên | Hiển thị con trỏ chuột vào ô “Tên” cho phép người dùng nhập vào ô |
| **EditText**  **Email** | Cho phép người dùng nhập Email | Chạm vào EditText Email | Hiển thị con trỏ chuột vào ô “Email” cho phép người dùng nhập vào ô |
| **EditText**  **Số điện thoại** | Cho phép người dùng nhập Số điện thoại | Chạm vào EditText Số điện thoại | Hiển thị con trỏ chuột vào ô “Số điện thoại” cho phép người dùng nhập vào ô |
| **EditText**  **Mật khẩu** | Cho phép người dùng nhập Mật khẩu | Chạm vào EditText Mật khẩu | Hiển thị con trỏ chuột vào ô “Mật khẩu” cho phép người dùng nhập vào ô |
| **EditText**  **Nhập lại mật khẩu** | Cho phép người dùng nhập lại mật khẩu | Chạm vào EditText Nhập lại mật khẩu | Hiển thị con trỏ chuột vào ô “Nhập lại mật khẩu” cho phép người dùng nhập vào ô |

|  |  |  |  |
| --- | --- | --- | --- |
| **Button Đăng ký** | Cho phép người dùng đăng ký tài khoản mới | Chạm vào nút đăng ký | 1. Lấy dữ liệu từ EditText Tên, Email, Số điện thoại, Mật khẩu. 2. Thêm các dữ liệu đã lấy được vào Databasse |
| **TextView**  **Nhấn vào đây để đăng nhập** | Cho phép người dùng chuyển đến màn hình đăng nhập tài khoản | Chạm vào edittext Nhấn vào đây để đăng nhập | Hiển thị màn hình đăng nhập tài khoản |

* 1. **Tính năng/ Thành phần #3: Màn hình Home**
     1. **Giao diện người dùng**

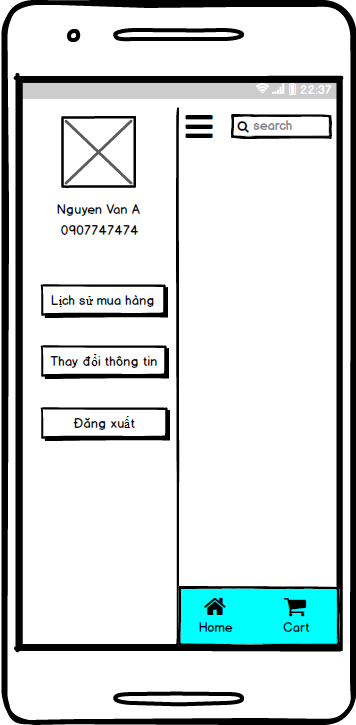


*Image3: Home*

* + 1. **Yêu cầu chức năng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình Home** | Màn hình Home gồm có:   * ImageButton Menu * EditText Search * ButtonImage Cart * ButtonImage Contact * ListView các sản phẩm * Trong listview có   + Image sản phẩm  + Textview tên sản phẩm  + Textview giá tiền  + Button thêm vào Giỏ hàng   * Button Home * Button Cart * Button Search * Button Contact | N/A | N/A |
| **ButtonImage Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào ButtonImage Cart | Hiển thị màn hình Cart |
| **ButtonImage Contact** | Cho phép người dùng chuyển đến màn hình Contact | Chạm vào ButtonImage Contact | Hiển thị màn hình Contact |
| **Button Home** | Cho phép người dùng chuyển đến màn hình Home | Chạm vào Button Home | Hiển thị màn hình Home |
| **Button Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào Button Cart | Hiển thị màn hình Cart |
| **Button Contact** | Cho phép người dùng chuyển đến màn hình Contact | Chạm vào Button Contact | Hiển thị màn hình Contact |
| **Button thêm vào giỏ hàng** | Cho phép người dùng thêm sản phẩm vào Cart | Chạm vào Button thêm | 1. Lấy dữ liệu từ TextView Tên tên sản phẩm, Số tiền, hình ảnh . 2. Thêm các dữ liệu đã lấy được vào Databasse 3. Sản phẩm được thêm vào màn hình giỏ hàng |
| **ImageButton Menu** | Cho phép màn hình Home hiện ra Navigation menu | Chạm vào ImageButton Menu | Hiển thị Navigation menu |
| **EditText**  **Search** | Cho phép người dùng nhập Tên sản phẩm | Chạm vào EditText Search | Hiển thị con trỏ chuột vào ô “Search” cho phép người dùng nhập vào ô |

* 1. **Tính năng/ Thành phần #4: Màn hình Navigation Menu**
     1. **Giao diện người dùng**



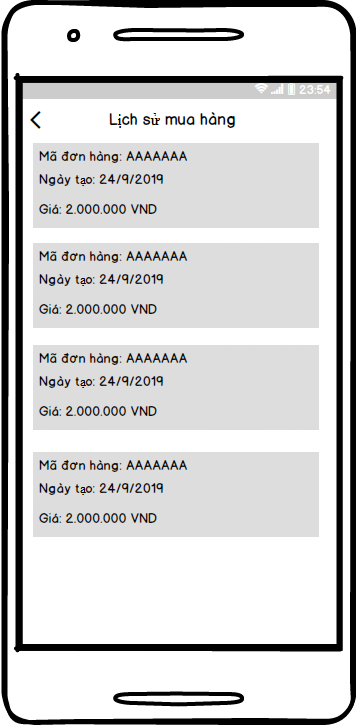
*Image4: Màn hình Navigation menu*

* + 1. **Yêu cầu chức năng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình Navigation menu)** | Màn hình Navigation menu gồm có:   * Image User * TextView Tên * TextView Số điện thoại * Button Lịch sử mua hàng * Button Thay đổi thông tin * Button Đăng xuất * Button Home * Button Cart * EditText Search | N/A | N/A |
| **Button Lịch sử mua hàng** | Cho phép người dùng chuyển đến màn hình lịch sử mua hàng | Chạm vào Button Lịch sử mua hàng | Chuyển hướng đến màn hình lịch sử mua hàng |
| **Button Thay đổi thông tin** | Cho phép người dùng chuyển đến màn hình Thay dổi thông tin | Chạm vào Button Thay đổi thông tin | Chuyển hướng đên màn hình Thay đổi thông tin |
| **Button Đăng xuất** | Cho phép người dùng chuyển hướng đến màn hình Đăng nhập | Chạm vào Button Đăng xuất | Chuyển hướng đến màn hình Đăng nhập |
| **Button Home** | Cho phép người dùng chuyển đến màn hình Home | Chạm vào Button Home | Hiển thị màn hình Home |
| **Button Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào Button Cart | Hiển thị màn hình Cart |
| **EditText**  **Search** | Cho phép người dùng nhập Tên sản phẩm | Chạm vào EditText Search | Hiển thị con trỏ chuột vào ô “Search” cho phép người dùng nhập vào ô |

### 

* 1. **Tính năng/ Thành phần #5: Màn hình lịch sử mua hàng**
     1. **Giao diện người dùng**

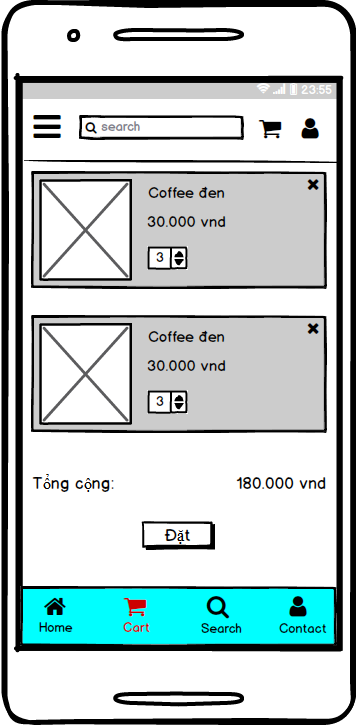


*Image5: Màn hình Lịch sử mua hàng*

* + 1. **Yêu cầu chức năng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình lịch sử mua hàng** | Màn hình lịch sử mua hàng gồm có:   * Button back * TextView Lịch sử mua hàng * Listview danh sách | N/A | N/A |
| **Button Back** | Cho phép người dùng chuyển về màn hình Navigation | Chạm vào Button back | Chuyển hướng đến màn hình Navigaytion |
| **Textview lịch sử mua hàng** | Cho phép hiển thị thông tin | N/A | N/A |
| **Listview danh sách** | Cho phép người dùng xem danh sách lịch sử mua hàng | N/A | N/A |

* 1. **Tính năng/ Thành phần #6: Màn hình giỏ hàng**
     1. **Giao diện người dùng**

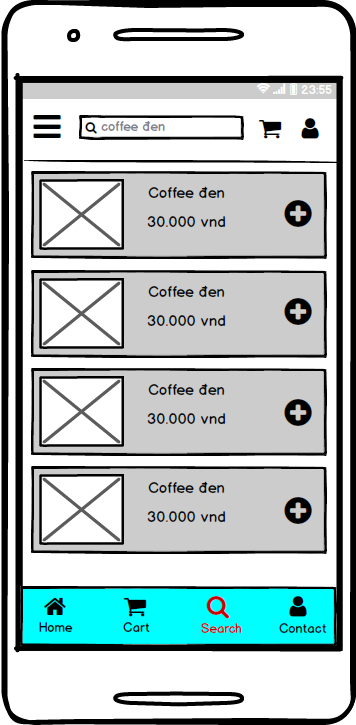


*Image6: Màn hình Cart*

* + 1. **Yêu cầu chức năng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình giỏ hàng** | Màn hình giỏ hàng gồm có:   * Button Home * Button Cart * Button Search * Button Contact * EditText Search * ButtonImage Cart * ButtonImage Contact * Listview hiển thị * Textview tổng cộng * Textview số tiền * Button đặt | N/A | N/A |
| **ButtonImage Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào ButtonImage Cart | Hiển thị màn hình Cart |
| **ButtonImage Contact** | Cho phép người dùng chuyển đến màn hình Contact | Chạm vào ButtonImage Contact | Hiển thị màn hình Contact |
| **Button Home** | Cho phép người dùng chuyển đến màn hình Home | Chạm vào Button Home | Hiển thị màn hình Home |
| **Button Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào Button Cart | Hiển thị màn hình Cart |
| **Button Contact** | Cho phép người dùng chuyển đến màn hình Contact | Chạm vào Button Contact | Hiển thị màn hình Contact |
| **ButtonImage Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào ButtonImage Cart | Hiển thị màn hình Cart |
| **ButtonImage Contact** | Cho phép người dùng chuyển đến màn hình Contact | Chạm vào ButtonImage Contact | Hiển thị màn hình Contact |
| **Button đặt** | Cho phép người dùng đặt | Chạm vào | Đặt thành công |

* 1. **Tính năng/ Thành phần #7: Màn hình Search**
     1. **Giao diện người dùng**



*Image7: Màn hình Search*

* + 1. **Yêu cầu chức năng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình Search** | Màn hình search gồm có:   * ImageButton Menu * EditText Search * ButtonImage Cart * ButtonImage Contact * ListView các sản phẩm * Trong listview có   + Image sản phẩm  + Textview tên sản phẩm  + Textview giá tiền  + Button thêm vào Giỏ hàng   * Button Home * Button Cart * Button Search * Button Contact | N/A | N/A |
| **ButtonImage Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào ButtonImage Cart | Hiển thị màn hình Cart |
| **ButtonImage Contact** | Cho phép người dùng chuyển đến màn hình Contact | Chạm vào ButtonImage Contact | Hiển thị màn hình Contact |
| **Button Home** | Cho phép người dùng chuyển đến màn hình Home | Chạm vào Button Home | Hiển thị màn hình Home |
| **Button Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào Button Cart | Hiển thị màn hình Cart |
| **Button Contact** | Cho phép người dùng chuyển đến màn hình Contact | Chạm vào Button Contact | Hiển thị màn hình Contact |
| **Button thêm vào giỏ hàng** | Cho phép người dùng thêm sản phẩm vào Cart | Chạm vào Button thêm | 1. Lấy dữ liệu từ TextView Tên tên sản phẩm, Số tiền, hình ảnh . 2. Thêm các dữ liệu đã lấy được vào Databasse   Sản phẩm được thêm vào màn hình giỏ hàng |
| **ImageButton Menu** | Cho phép màn hình Home hiện ra Navigation menu | Chạm vào ImageButton Menu | Hiển thị Navigation menu |
| **EditText**  **Search** | Cho phép người dùng nhập Tên sản phẩm | Chạm vào EditText Search | Hiển thị con trỏ chuột vào ô “Search” cho phép người dùng nhập vào ô |

* 1. **Tính năng/ Thành phần #8: Màn hình Contact**
     1. **Giao diện người dùng**



*Image8: Màn hình Contact*

* + 1. **Yêu cầu chức năng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình Contact** | Màn hình contact gồm có:   * ImageButton Menu * EditText Search * ButtonImage Cart * ButtonImage Contact * Button Home * Button Cart * Button Search * Button Contact * Editext tên * Editext emial * Editext số điện thoại * Editext địa chỉ * Icon tên * Icon email * Icon điện thoại * Icon địa chỉ * Button thay đổi thông tin | N/A | N/A |
| **ButtonImage Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào ButtonImage Cart | Hiển thị màn hình Cart |
| **ButtonImage Contact** | Cho phép người dùng chuyển đến màn hình Contact | Chạm vào ButtonImage Contact | Hiển thị màn hình Contact |
| **Button Home** | Cho phép người dùng chuyển đến màn hình Home | Chạm vào Button Home | Hiển thị màn hình Home |
| **Button Cart** | Cho phép người dùng chuyển đến màn hình Cart | Chạm vào Button Cart | Hiển thị màn hình Cart |
| **Button Contact** | Cho phép người dùng chuyển đến màn hình Contact | Chạm vào Button Contact | Hiển thị màn hình Contact |
| **Editext tên** | Cho phép hiển thị tên | N/A | N/A |
| **Editext email** | Cho phép hiển thị email | N/A | N/A |
| **Editext số điện thoại** | Cho phép hiểm thị số điện thoại | N/A | N/A |
| **Editext địa chỉ** | Cho phép hiển thị địa chỉ | N/A | N/A |
| **Button thay đổi thông tin** | Cho phép thay đổi thông tin | Chạm vào | Thay đổi thông tin từ các editext |
|  |  |  |  |

# **CHƯƠNG II: BÀI TẬP Ở LỚP**

1. **Phạm Văn Nhân**
2. **Exe2 ( Es6)**

// Hàm này có tác dụng sao chép thuộc tính của các đối tượng sources vào target

const a = { name: 'foo' }

const b = Object.assign({}, a, { name: 'bar', age: 1 }, { id: 9 })

console.log(b) // { name: 'bar', age: 1, id: 9 }

//

const developer = {

    firstName: 'Nhân',

    lastName: 'Phạm Văn',

    developer: true,

    age: 20,

  }

const { firstName, lastName } = developer;

console.log(firstName);

console.log(lastName);

console.log(developer);

//Arow

var hello = (name, massage) =>{

    console.log("Chào " + name + ", bạn thật " + massage);

};

hello('Nhân', 'đẹp trai quá đi')

// dem so

function Person() {

    this.age = 0;

    setInterval(() => {

        this.age++;

        console.log(this.age);

    }, 1000);

}

var user = new Person();

1. **Exe1 và exe3**

import React, { Component } from 'react';

import {

StyleSheet,

Text,

View,

TouchableOpacity,

Image,

Alert,

ScrollView,

FlatList,

Button,

} from 'react-native';

export default class ProductDetail extends Component {

constructor(props) {

super(props);

}

clickEventListener() {

Alert.alert("Success", "Product has beed added to cart")

}

render() {

return (

<View style={styles.container}>

<ScrollView>

<View style={{alignItems:'center', marginHorizontal:30}}>

<Image style={styles.productImg} source={require('./img/ao1.jpg')}/>

<Text style={styles.name}>TDC T-Shirt</Text>

<Text style={styles.price}>$ 12.22</Text>

<Text style={styles.description}>

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Aenean commodo ligula eget dolor. Aenean massa. Cum sociis

natoque penatibus et magnis dis parturient montes,

nascetur ridiculus mus. Donec quam felis, ultricies nec

</Text>

</View>

<View style={styles.starContainer}>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

</View>

<View style={styles.contentColors}>

<TouchableOpacity style={[styles.btnColor, {backgroundColor:"#000000"}]}></TouchableOpacity>

<TouchableOpacity style={[styles.btnColor, {backgroundColor:"#FFFF00"}]}></TouchableOpacity>

<TouchableOpacity style={[styles.btnColor, {backgroundColor:"#FF0000"}]}></TouchableOpacity>

<TouchableOpacity style={[styles.btnColor, {backgroundColor:"#228B22"}]}></TouchableOpacity>

<TouchableOpacity style={[styles.btnColor, {backgroundColor:"#20B2AA"}]}></TouchableOpacity>

<TouchableOpacity style={[styles.btnColor, {backgroundColor:"#FF4500"}]}></TouchableOpacity>

</View>

<View style={styles.contentSize}>

<TouchableOpacity style={styles.btnSize}><Text>S</Text></TouchableOpacity>

<TouchableOpacity style={styles.btnSize}><Text>M</Text></TouchableOpacity>

<TouchableOpacity style={styles.btnSize}><Text>L</Text></TouchableOpacity>

<TouchableOpacity style={styles.btnSize}><Text>XL</Text></TouchableOpacity>

</View>

<View style={styles.separator}></View>

<View style={styles.addToCarContainer}>

<TouchableOpacity style={styles.shareButton} onPress={()=> this.clickEventListener()}>

<Text style={styles.shareButtonText}>Add To Cart</Text>

</TouchableOpacity>

</View>

</ScrollView>

</View>

);

}

}

const styles = StyleSheet.create({

container:{

flex:1,

marginTop:20,

},

productImg:{

width:350,

height:250,

},

name:{

fontSize:28,

color:"#696969",

fontWeight:'bold'

},

price:{

marginTop:10,

fontSize:18,

color:"green",

fontWeight:'bold'

},

description:{

textAlign:'center',

marginTop:10,

color:"#696969",

},

star:{

width:40,

height:40,

},

btnColor: {

height:30,

width:30,

borderRadius:30,

marginHorizontal:3

},

btnSize: {

height:40,

width:40,

borderRadius:40,

borderColor:'#778899',

borderWidth:1,

marginHorizontal:3,

backgroundColor:'white',

flexDirection: 'row',

justifyContent: 'center',

alignItems: 'center',

},

starContainer:{

justifyContent:'center',

marginHorizontal:30,

flexDirection:'row',

marginTop:20

},

contentColors:{

justifyContent:'center',

marginHorizontal:30,

flexDirection:'row',

marginTop:20

},

contentSize:{

justifyContent:'center',

marginHorizontal:30,

flexDirection:'row',

marginTop:20

},

separator:{

height:2,

backgroundColor:"#eeeeee",

marginTop:20,

marginHorizontal:30

},

shareButton: {

marginTop:10,

height:45,

flexDirection: 'row',

justifyContent: 'center',

alignItems: 'center',

borderRadius:30,

backgroundColor: "#00BFFF",

},

shareButtonText:{

color: "#FFFFFF",

fontSize:20,

},

addToCarContainer:{

marginHorizontal:30

}

});

1. **Exe4**

import React, { Component } from 'react';

import {

StyleSheet,

Text,

View,

TouchableOpacity,

Image,

Alert,

ScrollView,

FlatList,

Button,

} from 'react-native';

var URL = "http://10.0.3.2:81/DoUong/LoadData.php"

export default class Exe4 extends Component {

constructor(props) {

super(props);

this.state = {

myData:[]

,loaded: false

}

}

clickEventListener() {

Alert.alert("Success", "Product has beed added to cart")

}

componentDidMount(){

fetch(URL)

.then((response) => response.json())

.then((responseData) => {

this.setState({

myData: responseData

});

this.setState({

loaded: true

})

})

.done()

}

render() {

if(this.state.loaded){

return (

<View style={styles.container}>

<ScrollView>

<View style={{alignItems:'center', marginHorizontal:30}}>

<Image style={styles.productImg} source={{uri: this.state.myData[1]["image"]}}/>

<Text style={styles.name}>{this.state.myData[1]["title"]}</Text>

<Text style={styles.price}>{this.state.myData[1]["price"]}</Text>

<Text style={styles.description}>

{this.state.myData[1]["content"]}

</Text>

</View>

<View style={styles.starContainer}>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

<Image style={styles.star} source={{uri:"https://img.icons8.com/color/40/000000/star.png"}}/>

</View>

<View style={styles.contentSize}>

<TouchableOpacity style={styles.btnSize}><Text>S</Text></TouchableOpacity>

<TouchableOpacity style={styles.btnSize}><Text>M</Text></TouchableOpacity>

<TouchableOpacity style={styles.btnSize}><Text>L</Text></TouchableOpacity>

</View>

<View style={styles.separator}></View>

<View style={styles.addToCarContainer}>

<TouchableOpacity style={styles.shareButton} onPress={()=> this.clickEventListener()}>

<Text style={styles.shareButtonText}>Add To Cart</Text>

</TouchableOpacity>

</View>

</ScrollView>

</View>

);

}

else{

return null;

}

}

}

const styles = StyleSheet.create({

container:{

flex:1,

marginTop:20,

},

productImg:{

width:300,

height:300,

},

name:{

fontSize:28,

color:"#696969",

fontWeight:'bold'

},

price:{

marginTop:10,

fontSize:18,

color:"green",

fontWeight:'bold'

},

description:{

textAlign:'center',

marginTop:10,

color:"#696969",

},

star:{

width:40,

height:40,

},

btnColor: {

height:30,

width:30,

borderRadius:30,

marginHorizontal:3

},

btnSize: {

height:40,

width:40,

borderRadius:40,

borderColor:'#778899',

borderWidth:1,

marginHorizontal:3,

backgroundColor:'white',

flexDirection: 'row',

justifyContent: 'center',

alignItems: 'center',

},

starContainer:{

justifyContent:'center',

marginHorizontal:30,

flexDirection:'row',

marginTop:20

},

contentColors:{

justifyContent:'center',

marginHorizontal:30,

flexDirection:'row',

marginTop:20

},

contentSize:{

justifyContent:'center',

marginHorizontal:30,

flexDirection:'row',

marginTop:20

},

separator:{

height:2,

backgroundColor:"#eeeeee",

marginTop:20,

marginHorizontal:30

},

shareButton: {

marginTop:10,

height:45,

flexDirection: 'row',

justifyContent: 'center',

alignItems: 'center',

borderRadius:30,

backgroundColor: "#00BFFF",

},

shareButtonText:{

color: "#FFFFFF",

fontSize:20,

},

addToCarContainer:{

marginHorizontal:30

}

});

1. **Nguyễn Công Nhân**
2. **Exe2 ( Es6)**

// WeakMap

// Khởi tạo

var weak = new WeakMap();

// h Danh sáckey

var key1 = {};

var key2 = {};

// Thêm phần tử

weak.set(key1, "Giá trị 01");

weak.set(key2, "Giá trị 02");

// Lấy giá trị

console.log(weak.get(key1));

console.log(weak.get(key2));

// Kiểm tra tồn tại

var other\_key = {};

console.log(weak.has(key1));

console.log(weak.has(other\_key));

// Xóa phần tử

weak.delete(key1);

var PhepToan = (a, b) => {

var pheptinh = a + b;

console.log("Phép Cộng: " + pheptinh);

var pheptinh2 = a - b;

console.log("Phép Trừ: " + pheptinh2);

var pheptinh3 = a \* b;

console.log("Phép Nhân: " + pheptinh3);

var pheptinh4 = a / b;

console.log("Phép Chia: " + pheptinh4);

};

PhepToan(5, 2);

1. **Exe1 và ese3**
2. **Nguyễn Văn An**

const person = {

Ten: 'Nguyen Van An',

Lop: 'CD17TT8',

Ma:"17211TT3235",

Nganh: "Cong nghe thong tin",

Tuoi: 20,

Sothich: "Soccer, Game, Watch TV",

}

const { Ten, Lop, Ma, Nganh, Tuoi, Sothich } = person;

console.log("\tDANH SACH SINH VIEN");

console.log("Ho va ten: " + Ten);

console.log("Lop: "+ Lop);

console.log("Ma: "+ Ma);

console.log("Nganh: "+ Nganh);

console.log("Tuoi: "+ Tuoi);

console.log(person);

var other1 = ["Ho va ten: " + Ten + "\t" + "Lop: " +Lop + "\tMa: " + Ma + "\tNganh: " + Nganh + "\tTuoi: " + Tuoi + "\tSo Thich: " + Sothich ];

var params = [ "hello!"].concat(other1);

console.log(params);

# **CHƯƠNG III: TÌM HIỂU CÁC COMPONEMT**

# PROPS

Hầu hết các thành phần có thể được tùy chỉnh khi chúng được tạo ra, với các tham số khác nhau. Các tham số tạo này được gọi props, viết tắt của các thuộc tính.

import React, { Component } from 'react';

import { Image } from 'react-native';

export default class Bananas extends Component {

render() {

let pic = {

uri: 'https://upload.wikimedia.org/wikipedia/commons/d/de/Bananavarieties.jpg'

};

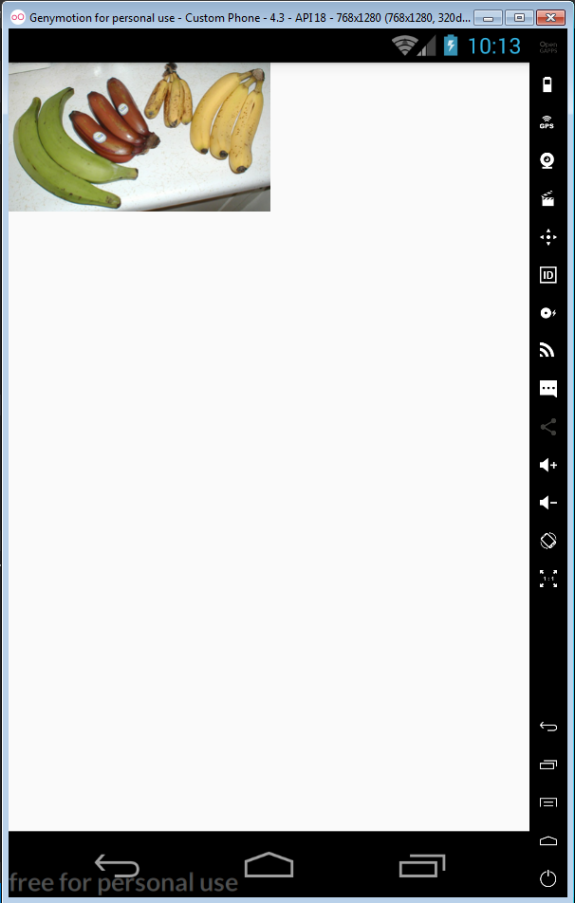
return (

<Image source={pic} style={{width: 193, height: 110}}/>

);

}

}



Các thành phần của riêng bạn cũng có thể sử dụng props. Điều này cho phép bạn tạo một thành phần duy nhất được sử dụng ở nhiều nơi khác nhau trong ứng dụng của bạn, với các thuộc tính hơi khác nhau ở mỗi nơi. Chỉ cần tham khảo this.props trong render chức năng  . Đây là một ví dụ:

import React, { Component } from 'react';

import { Text, View } from 'react-native';

class Greeting extends Component {

render() {

return (

<View style={{alignItems: 'center'}}>

<Text>Hello {this.props.name}!</Text>

</View>

);

}

}

export default class LotsOfGreetings extends Component {

render() {

return (

<View style={{alignItems: 'center', top: 50}}>

<Greeting name='Rexxar' />

<Greeting name='Jaina' />

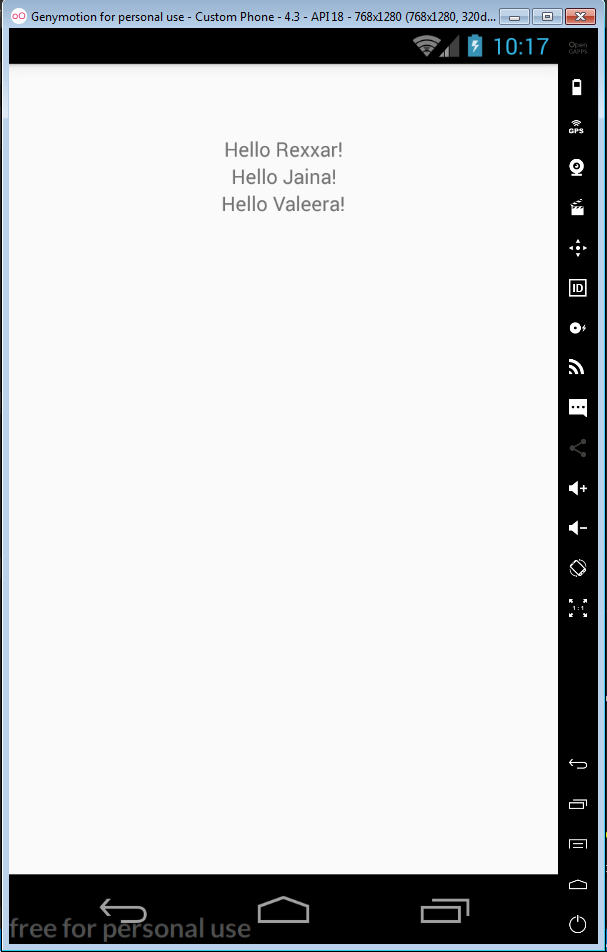
<Greeting name='Valeera' />

</View>

);

}

}



# STATE

Có hai loại dữ liệu kiểm soát một thành phần: props và state. propsđược thiết lập bởi cha mẹ và chúng được cố định trong suốt vòng đời của một thành phần. Đối với dữ liệu sẽ thay đổi, chúng ta phải sử dụng state.

import React, { Component } from 'react';

import { Text, View } from 'react-native';

class Blink extends Component {

componentDidMount(){

// Toggle the state every second

setInterval(() => (

this.setState(previousState => (

{ isShowingText: !previousState.isShowingText }

))

), 1000);

}

//state object

state = { isShowingText: true };

render() {

if (!this.state.isShowingText) {

return null;

}

return (

<Text>{this.props.text}</Text>

);

}

}

export default class BlinkApp extends Component {

render() {

return (

<View>

<Blink text='I love to blink' />

<Blink text='Yes blinking is so great' />

<Blink text='Why did they ever take this out of HTML' />

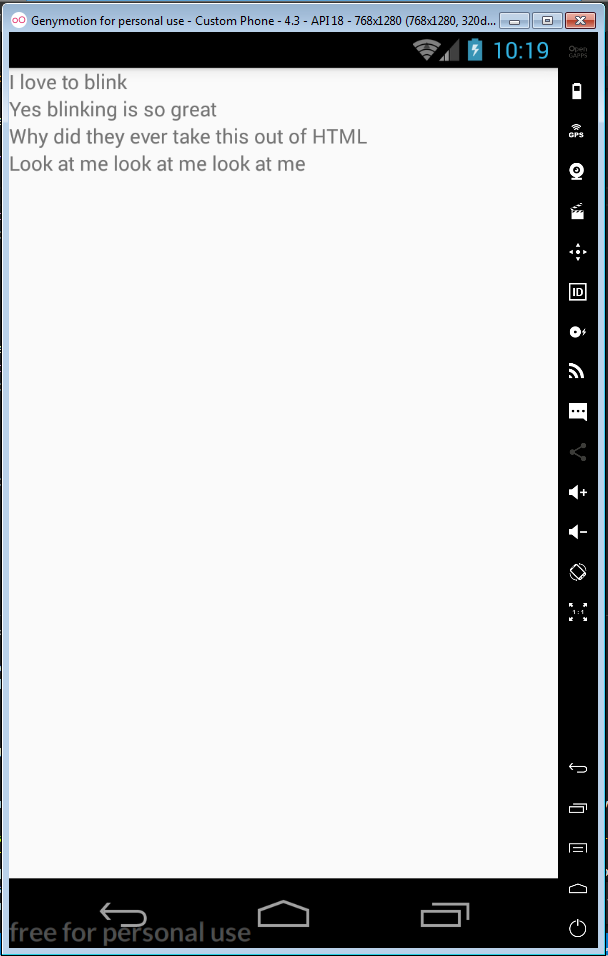
<Blink text='Look at me look at me look at me' />

</View>

);

}

}



# Check Box

import React, { Component } from 'react';

import { Platform, View, Text, CheckBox } from 'react-native';

var tempCheckValues = [];

export default class App extends Component {

  constructor(props) {

    super(props);

    this.state = {

      checkBoxChecked: []

    }

  }

  checkBoxChanged(id, value) {

    this.setState({

      checkBoxChecked: tempCheckValues

    })

    var tempCheckBoxChecked = this.state.checkBoxChecked;

    tempCheckBoxChecked[id] = !value;

    this.setState({

      checkBoxChecked: tempCheckBoxChecked

    })

  }

  render() {

    const products = [{

      id: 1

    },

    {

      id: 2

    },

    {

      id: 3

    }];

    return (

      products.map((val) => {

        { tempCheckValues[val.id] = false }

        return (

          <View key={val.id} style={{ flexDirection: 'column' }}>

            <CheckBox

              value={this.state.checkBoxChecked[val.id]}

              onValueChange={() => this.checkBoxChanged(val.id, this.state.checkBoxChecked[val.id])}

            />

             <Text style={{marginTop: 5}}> this is checkbox</Text>

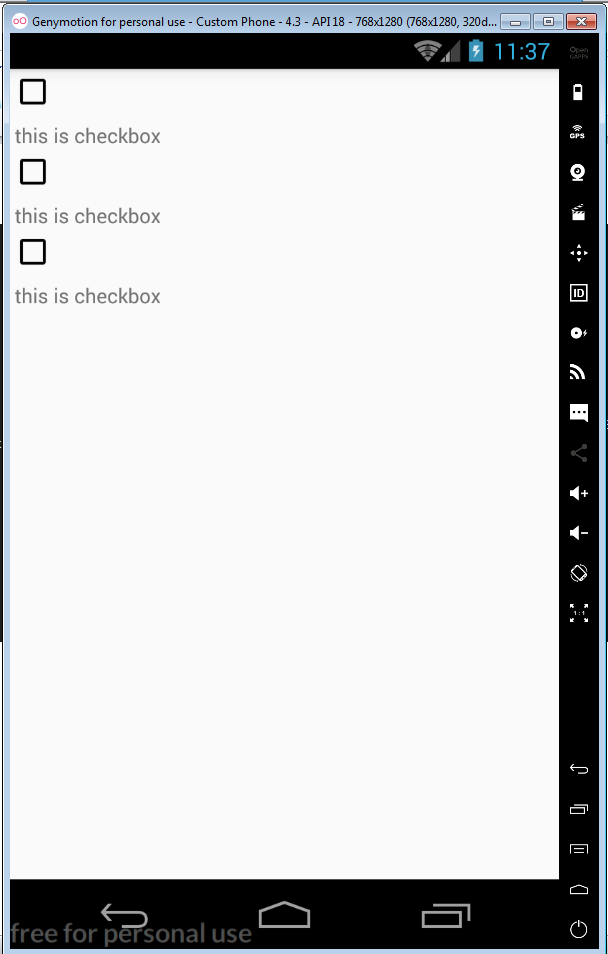
          </View >

        )})

    );

  }

}



# STYLE

Với React Native, bạn không sử dụng ngôn ngữ hoặc cú pháp đặc biệt để xác định kiểu. Bạn chỉ cần tạo kiểu cho ứng dụng của mình bằng JavaScript. Tất cả các thành phần cốt lõi chấp nhận một prop có tên style.

Khi một thành phần phát triển phức tạp, nó thường được sử dụng StyleSheet.createđể xác định một số kiểu ở một nơi. Đây là một ví dụ:

import React, { Component } from 'react';

import { StyleSheet, Text, View } from 'react-native';

const styles = StyleSheet.create({

bigBlue: {

color: 'blue',

fontWeight: 'bold',

fontSize: 30,

},

red: {

color: 'red',

},

});

export default class LotsOfStyles extends Component {

render() {

return (

<View>

<Text style={styles.red}>just red</Text>

<Text style={styles.bigBlue}>just bigBlue</Text>

<Text style={[styles.bigBlue, styles.red]}>bigBlue, then red</Text>

<Text style={[styles.red, styles.bigBlue]}>red, then bigBlue</Text>

</View>

);

}

}



# Layout with Flexbox

Flex Direction

Flex Direction sẽ xác định cách các mặt hàng của bạn sẽ được chuyển đến vùng lấp đầy trên không gian có sẵn dọc theo trục chính của bạn. Không gian sẽ được phân chia theo thuộc tính flex của từng yếu tố.

import React, { Component } from 'react';

import { View } from 'react-native';

export default class FlexDirectionBasics extends Component {

render() {

return (

// Try setting `flexDirection` to `column`.

<View style={{flex: 1, flexDirection: 'row'}}>

<View style={{width: 50, height: 50, backgroundColor: 'powderblue'}} />

<View style={{width: 50, height: 50, backgroundColor: 'skyblue'}} />

<View style={{width: 50, height: 50, backgroundColor: 'steelblue'}} />

</View>

);

}

};



# Handling Text Input

**Text input** là một thành phần cơ bản cho phép người dùng nhập văn bản. Nó có một prop onChangeText có chức năng được gọi mỗi khi văn bản thay đổi và prop onSubmitEditing có chức năng được gọi khi văn bản được gửi.

import React, { Component } from 'react';

import { Text, TextInput, View } from 'react-native';

export default class PizzaTranslator extends Component {

constructor(props) {

super(props);

this.state = {text: ''};

}

render() {

return (

<View style={{padding: 10}}>

<TextInput

style={{height: 40}}

placeholder="Type here to translate!"

onChangeText={(text) => this.setState({text})}

value={this.state.text}

/>

<Text style={{padding: 10, fontSize: 42}}>

{this.state.text.split(' ').map((word) => word && '🍕').join(' ')}

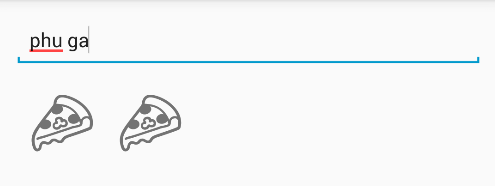
</Text>

</View>

);

}

}



# Handling Touches

Người dùng tương tác với các ứng dụng di động chủ yếu thông qua cảm ứng. Họ có thể sử dụng kết hợp các cử chỉ, chẳng hạn như chạm vào nút, cuộn danh sách hoặc phóng to bản đồ. React Native cung cấp các thành phần để xử lý tất cả các loại cử chỉ thông thường, cũng như hệ thống phản hồi cử chỉ toàn diện để cho phép nhận dạng cử chỉ nâng cao hơn, nhưng một thành phần bạn rất có thể quan tâm là Nút cơ bản.

import React, { Component } from 'react';

import { Platform, StyleSheet, Text, TouchableHighlight, TouchableOpacity, TouchableNativeFeedback, TouchableWithoutFeedback, View } from 'react-native';

export default class Touchables extends Component {

\_onPressButton() {

alert('You tapped the button!')

}

\_onLongPressButton() {

alert('You long-pressed the button!')

}

render() {

return (

<View style={styles.container}>

<TouchableHighlight onPress={this.\_onPressButton} underlayColor="white">

<View style={styles.button}>

<Text style={styles.buttonText}>TouchableHighlight</Text>

</View>

</TouchableHighlight>

<TouchableOpacity onPress={this.\_onPressButton}>

<View style={styles.button}>

<Text style={styles.buttonText}>TouchableOpacity</Text>

</View>

</TouchableOpacity>

<TouchableNativeFeedback

onPress={this.\_onPressButton}

background={Platform.OS === 'android' ? TouchableNativeFeedback.SelectableBackground() : ''}>

<View style={styles.button}>

<Text style={styles.buttonText}>TouchableNativeFeedback {Platform.OS !== 'android' ? '(Android only)' : ''}</Text>

</View>

</TouchableNativeFeedback>

<TouchableWithoutFeedback

onPress={this.\_onPressButton}

>

<View style={styles.button}>

<Text style={styles.buttonText}>TouchableWithoutFeedback</Text>

</View>

</TouchableWithoutFeedback>

<TouchableHighlight onPress={this.\_onPressButton} onLongPress={this.\_onLongPressButton} underlayColor="white">

<View style={styles.button}>

<Text style={styles.buttonText}>Touchable with Long Press</Text>

</View>

</TouchableHighlight>

</View>

);

}

}

const styles = StyleSheet.create({

container: {

paddingTop: 60,

alignItems: 'center'

},

button: {

marginBottom: 30,

width: 260,

alignItems: 'center',

backgroundColor: '#2196F3'

},

buttonText: {

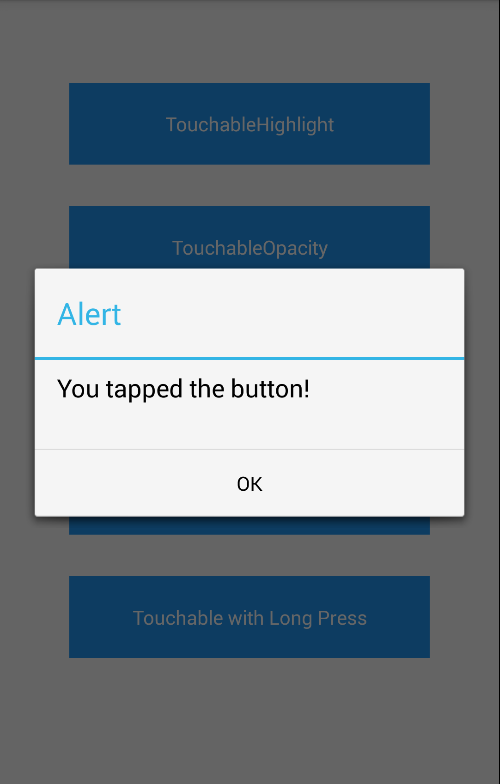
textAlign: 'center',

padding: 20,

color: 'white'

}

});



# Using a ScrollView

ScrollView là một thùng chứa cuộn chung có thể lưu trữ nhiều thành phần và khung nhìn. Các mục có thể cuộn không cần phải đồng nhất và bạn có thể cuộn cả theo chiều dọc và chiều ngang (bằng cách đặt thuộc tính ngang).

import React, { Component } from 'react';

import { ScrollView, Image, Text } from 'react-native';

export default class IScrolledDownAndWhatHappenedNextShockedMe extends Component {

render() {

return (

<ScrollView>

<Text style={{fontSize:96}}>Scroll me plz</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Text style={{fontSize:96}}>If you like</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Text style={{fontSize:96}}>Scrolling down</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Text style={{fontSize:96}}>What's the best</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Text style={{fontSize:96}}>Framework around?</Text>

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

<Image source={{uri: "https://facebook.github.io/react-native/img/tiny\_logo.png", width: 64, height: 64}} />

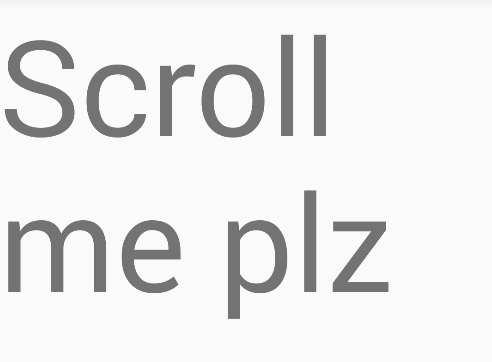
<Text style={{fontSize:80}}>React Native</Text>

</ScrollView>

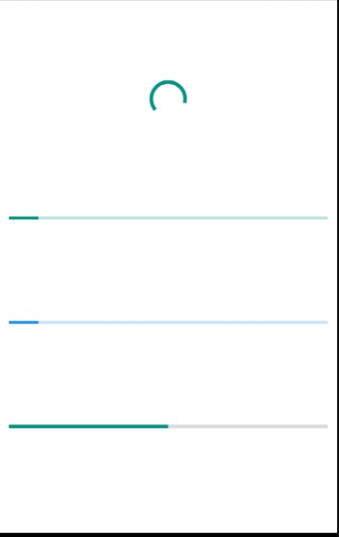
);

}

}



# Progressbar

**- Image :**

- code :

import React, {Component} from 'react';

import {ProgressBarAndroid, StyleSheet, View} from 'react-native';

export default class App extends Component {

  render() {

    return (

      <View style={styles.container}>

        <ProgressBarAndroid />

        <ProgressBarAndroid styleAttr="Horizontal" />

        <ProgressBarAndroid styleAttr="Horizontal" color="#2196F3" />

        <ProgressBarAndroid

          styleAttr="Horizontal"

          indeterminate={false}

          progress={0.5}

        />

      </View>

    );

  }

}

const styles = StyleSheet.create({

  container: {

    flex: 1,

    justifyContent: 'space-evenly',

    padding: 10,

  },

});

# ListView

**- Image :**



- Code :

import React, { Component } from 'react'

import { Text, ListView, StyleSheet } from 'react-native'

export default class MyListComponent extends Component {

    constructor() {

        super();

        const ds = new ListView.DataSource({rowHasChanged: (r1, r2) => r1 !== r2});

        this.state = {

            dataSource: ds.cloneWithRows(['Android','iOS', 'Java','Php', 'Hadoop',

                'Sap', 'Python','Ajax', 'C++',

                'Ruby', 'Rails','.Net', 'Perl']),

        };

    }

    render() {

        return (

            <ListView

                dataSource={this.state.dataSource}

                renderRow={

                    (rowData) =>

                        <Text style={{fontSize: 20}}>{rowData}</Text>}

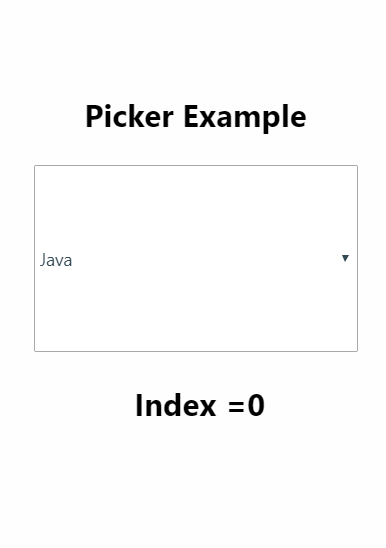
            />

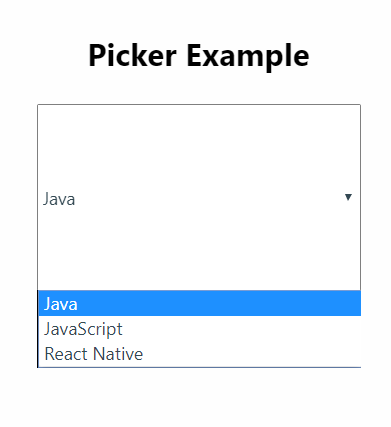
        );

    }

}

# Picker

**- Image :**



- Code :

import React, { Component } from 'react'

import {StyleSheet,View, Text,Picker} from 'react-native'

export default class SwitchExample extends Component {

    state = {

        choosenIndex: 0

    };

    render() {

        return (

            <View style={styles.container}>

                <Text style={styles.textStyle}>Picker Example</Text>

                <Picker style={styles.pickerStyle}

                        selectedValue={this.state.language}

                        onValueChange={(itemValue, itemPosition) =>

                            this.setState({language: itemValue, choosenIndex: itemPosition})}

                    >

                    <Picker.Item label="Java" value="java" />

                    <Picker.Item label="JavaScript" value="js" />

                    <Picker.Item label="React Native" value="rn" />

                </Picker>

                <Text style={styles.textStyle}> {"Index ="+this.state.choosenIndex}</Text>

            </View>

        );

    }

}

const styles = StyleSheet.create ({

     container: {

         flex: 1,

         alignItems: 'center',

         justifyContent: 'center',

     },

    textStyle:{

        margin: 24,

        fontSize: 25,

        fontWeight: 'bold',

        textAlign: 'center',

    },

    pickerStyle:{

        height: 150,

        width: "80%",

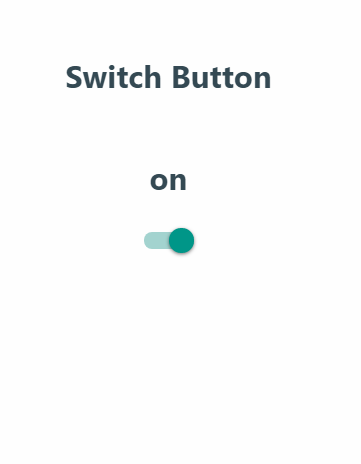
        color: '#344953',

        justifyContent: 'center',

    }

})

# Switch Button

**- Image :**

- Code :

import React, { Component } from 'react'

import {StyleSheet, Switch, View, Text} from 'react-native'

export default class SwitchExample extends Component {

    state = {

        switchValue: false

    };

    render() {

        return (

            <View style={styles.container}>

                <Text style={styles.textStyle}>Switch Button</Text>

                <Text style={styles.textStyle}>{this.state.switchValue ? 'on' :'off'}</Text>

                <Switch

                    value={this.state.switchValue}

                    onValueChange ={(switchValue)=>this.setState({switchValue})}/>

            </View>

        );

    }

}

const styles = StyleSheet.create ({

     container: {

         flex: 1,

         alignItems: 'center',

         justifyContent: 'center',

     },

    textStyle:{

        margin: 24,

        fontSize: 25,

        fontWeight: 'bold',

        textAlign: 'center',

        color: '#344953'

    }

})

# **CHƯƠNG IV: CHI TIẾT ỨNG DỤNG VÀ CÔNG VIỆC CỦA MỖI THÀNH VIÊN**

1. **Phạm Văn Nhân**

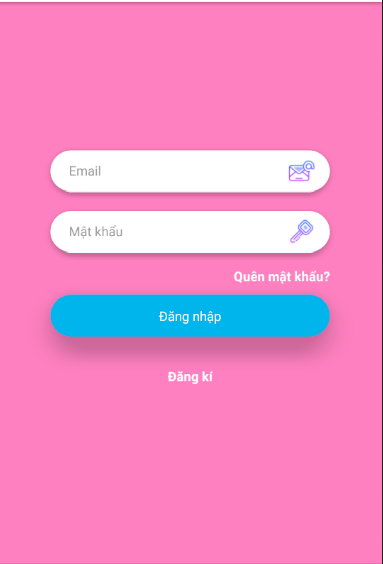
* Màn hình thêm sản phẩm



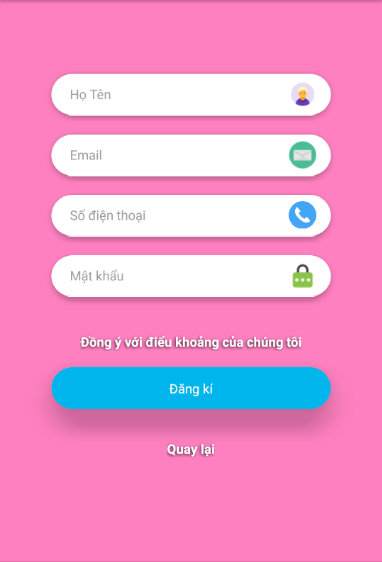
* Code
* /\* eslint-disable no-alert \*/
* import React, {Component} from 'react';
* import {
* StyleSheet,
* Text,
* View,
* TextInput,
* Button,
* TouchableOpacity,
* Image,
* Alert,
* } from 'react-native';
* export default class SignUp extends Component {
* state: {
* masp: string,
* title: string,
* image: string,
* price: string,
* };
* constructor(props) {
* super(props);
* this.state = {
* masp: '',
* title: '',
* image: '',
* price: '',
* };
* }
* insertSP() {
* fetch('http://10.0.2.2:8888/WebSV/Insert\_SanPham.php', {
* method: 'POST',
* header: {
* Accept: 'application/json',
* 'Content-Type': 'application/json',
* },
* body: JSON.stringify({
* masp: this.state.masp,
* title: this.state.title,
* image: this.state.image,
* price: this.state.price,
* }),
* })
* .then(response => response.json())
* .then(responseJsonFromServer => {
* alert(responseJsonFromServer);
* this.setState({ActivityIndicator\_Loading: false});
* })
* .catch(error => {
* console.log(error);
* this.setState({ActivityIndicator\_Loading: false});
* });
* }
* //onClickListener = viewId => {
* //Alert.alert('Alert', 'Button pressed ' + viewId);
* //};
* render() {
* return (
* <View style={styles.MainContainer}>
* <TextInput
* style={styles.TextInputStyleClass}
* placeholder="Nhập mã sản phẩm"
* underlineColorAndroid="transparent"
* onChangeText={masp => this.setState({masp})}
* />
* <TextInput
* style={styles.TextInputStyleClass}
* placeholder="Nhập tên sản phẩm"
* underlineColorAndroid="transparent"
* onChangeText={title => this.setState({title})}
* />
* <TextInput
* style={styles.TextInputStyleClass}
* placeholder="Nhập tên hình ảnh"
* underlineColorAndroid="transparent"
* onChangeText={image => this.setState({image})}
* />
* <TextInput
* style={styles.TextInputStyleClass}
* placeholder="Nhập giá sản phẩm"
* underlineColorAndroid="transparent"
* onChangeText={price => this.setState({price})}
* />
* <TouchableOpacity
* style={[styles.TouchableOpacityStyle]}
* onPress={() => this.insertSP()}>
* <Text style={styles.TextStyle}>Thêm sản phẩm</Text>
* </TouchableOpacity>
* {this.state.ActivityIndicator\_Loading ? (
* <ActivityIndicator
* color="#009688"
* size="large"
* style={styles.ActivityIndicatorStyle}
* />
* ) : null}
* </View>
* );
* }
* }
* const resizeMode = 'center';
* const styles = StyleSheet.create({
* container: {
* flex: 1,
* justifyContent: 'center',
* alignItems: 'center',
* backgroundColor: 'rgb(255,128,192)',
* },
* TextStyle: {
* color: '#fff',
* textAlign: 'center',
* fontSize: 18,
* },
* MainContainer: {
* flex: 1,
* justifyContent: 'center',
* alignItems: 'center',
* margin: 20,
* },
* TouchableOpacityStyle: {
* paddingTop: 10,
* paddingBottom: 10,
* backgroundColor: '#009688',
* marginBottom: 20,
* width: '90%',
* },
* ActivityIndicatorStyle: {
* position: 'absolute',
* left: 0,
* right: 0,
* top: 0,
* bottom: 0,
* alignItems: 'center',
* justifyContent: 'center',
* },
* TextInputStyleClass: {
* textAlign: 'center',
* height: 40,
* backgroundColor: '#fff',
* borderWidth: 1,
* borderColor: '#009688',
* borderRadius: 7,
* marginBottom: 10,
* width: '95%',
* },
* inputContainer: {
* borderBottomColor: '#F5FCFF',
* backgroundColor: '#FFFFFF',
* borderRadius: 30,
* borderBottomWidth: 1,
* width: 300,
* height: 45,
* marginBottom: 20,
* flexDirection: 'row',
* alignItems: 'center',
* shadowColor: '#808080',
* shadowOffset: {
* width: 0,
* height: 2,
* },
* shadowOpacity: 0.25,
* shadowRadius: 3.84,
* elevation: 5,
* },
* inputs: {
* height: 45,
* marginLeft: 16,
* borderBottomColor: '#FFFFFF',
* flex: 1,
* },
* inputIcon: {
* width: 30,
* height: 30,
* marginRight: 15,
* justifyContent: 'center',
* },
* buttonContainer: {
* height: 45,
* flexDirection: 'row',
* justifyContent: 'center',
* alignItems: 'center',
* marginBottom: 20,
* width: 300,
* borderRadius: 30,
* backgroundColor: 'transparent',
* },
* btnByRegister: {
* height: 15,
* flexDirection: 'row',
* justifyContent: 'center',
* alignItems: 'center',
* marginVertical: 20,
* width: 300,
* backgroundColor: 'transparent',
* },
* loginButton: {
* backgroundColor: '#00b5ec',
* shadowColor: '#808080',
* shadowOffset: {
* width: 0,
* height: 9,
* },
* shadowOpacity: 0.5,
* shadowRadius: 12.35,
* elevation: 19,
* },
* loginText: {
* color: 'white',
* },
* bgImage: {
* flex: 1,
* resizeMode,
* position: 'absolute',
* width: '100%',
* height: '100%',
* justifyContent: 'center',
* },
* btnText: {
* color: 'white',
* fontWeight: 'bold',
* textShadowColor: 'rgba(0, 0, 0, 0.75)',
* textShadowOffset: {width: -1, height: 1},
* textShadowRadius: 10,
* },
* textByRegister: {
* color: 'white',
* fontWeight: 'bold',
* textAlign: 'center',
* textShadowColor: 'rgba(0, 0, 0, 0.75)',
* textShadowOffset: {width: -1, height: 1},
* textShadowRadius: 10,
* },
* });

1. **Nguyễn Công Nhân**

* Màn hình đăng nhập



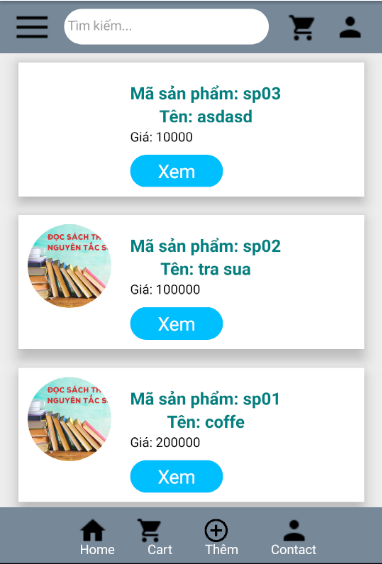
* Code
* import React, {Component} from 'react';
* import {
* StyleSheet,
* Text,
* View,
* TextInput,
* Button,
* TouchableOpacity,
* Image,
* Alert,
* } from 'react-native';
* export default class Login extends Component {
* static navigationOptions = {
* title: 'Welcome',
* };
* state: {password: string, email: string};
* constructor(props) {
* super(props);
* // eslint-disable-next-line no-undef
* this.state = {
* email: '',
* password: '',
* };
* }
* onClickListener = viewId => {
* Alert.alert('Alert', 'Button pressed ' + viewId);
* };
* render() {
* const {navigate} = this.props.navigation;
* return (
* <View style={styles.container}>
* <View style={styles.inputContainer}>
* <TextInput
* style={styles.inputs}
* placeholder="Email"
* keyboardType="email-address"
* underlineColorAndroid="transparent"
* onChangeText={email => this.setState({email})}
* />
* <Image
* style={styles.inputIcon}
* source={{uri: 'https://img.icons8.com/nolan/40/000000/email.png'}}
* />
* </View>
* <View style={styles.inputContainer}>
* <TextInput
* style={styles.inputs}
* placeholder="Mật khẩu"
* secureTextEntry={true}
* underlineColorAndroid="transparent"
* onChangeText={password => this.setState({password})}
* />
* <Image
* style={styles.inputIcon}
* source={{uri: 'https://img.icons8.com/nolan/40/000000/key.png'}}
* />
* </View>
* <TouchableOpacity
* style={styles.btnForgotPassword}
* onPress={() => this.login()}>
* <Text style={styles.btnText}>Quên mật khẩu?</Text>
* </TouchableOpacity>
* <TouchableOpacity
* style={[styles.buttonContainer, styles.loginButton]}
* onPress={() => this.login()}>
* <Text style={styles.loginText}>Đăng nhập</Text>
* </TouchableOpacity>
* <TouchableOpacity
* style={styles.buttonContainer}
* onPress={() => navigate('Register')}>
* <Text style={styles.btnText}>Đăng kí</Text>
* </TouchableOpacity>
* </View>
* );
* }
* login() {
* const {navigate} = this.props.navigation;
* fetch('http://10.0.2.2:8888/WebSV/Login.php', {
* method: 'POST',
* headers: {
* Accept: 'application/json',
* 'Content-Type': 'application/json',
* },
* body: JSON.stringify({
* username: this.state.email,
* password: this.state.password,
* }),
* })
* .then(response => response.json())
* .then(responseJson => {
* if (responseJson == 'true') {
* navigate('Home');
* } else {
* alert('Login Failed');
* }
* })
* .catch(error => {
* console.log(error);
* });
* }
* }
* const resizeMode = 'center';
* const styles = StyleSheet.create({
* container: {
* flex: 1,
* justifyContent: 'center',
* alignItems: 'center',
* backgroundColor: 'rgb(255,128,192)',
* },
* inputContainer: {
* borderBottomColor: '#F5FCFF',
* backgroundColor: '#FFFFFF',
* borderRadius: 30,
* borderBottomWidth: 1,
* width: 300,
* height: 45,
* marginBottom: 20,
* flexDirection: 'row',
* alignItems: 'center',
* shadowColor: '#808080',
* shadowOffset: {
* width: 0,
* height: 2,
* },
* shadowOpacity: 0.25,
* shadowRadius: 3.84,
* elevation: 5,
* },
* inputs: {
* height: 45,
* marginLeft: 16,
* borderBottomColor: '#FFFFFF',
* flex: 1,
* },
* inputIcon: {
* width: 30,
* height: 30,
* marginRight: 15,
* justifyContent: 'center',
* },
* buttonContainer: {
* height: 45,
* flexDirection: 'row',
* justifyContent: 'center',
* alignItems: 'center',
* marginBottom: 20,
* width: 300,
* borderRadius: 30,
* backgroundColor: 'transparent',
* },
* btnForgotPassword: {
* height: 15,
* flexDirection: 'row',
* justifyContent: 'flex-end',
* alignItems: 'flex-end',
* marginBottom: 10,
* width: 300,
* backgroundColor: 'transparent',
* },
* loginButton: {
* backgroundColor: '#00b5ec',
* shadowColor: '#808080',
* shadowOffset: {
* width: 0,
* height: 9,
* },
* shadowOpacity: 0.5,
* shadowRadius: 12.35,
* elevation: 19,
* },
* loginText: {
* color: 'white',
* },
* bgImage: {
* flex: 1,
* resizeMode,
* position: 'absolute',
* width: '100%',
* height: '100%',
* justifyContent: 'center',
* },
* btnText: {
* color: 'white',
* fontWeight: 'bold',
* },
* });
* Màn hình đăng kí



* Code
* import React, {Component} from 'react';
* import {
* StyleSheet,
* Text,
* View,
* TextInput,
* Button,
* TouchableOpacity,
* Image,
* Alert,
* } from 'react-native';
* export default class SignUp extends Component {
* state: {
* password: string,
* email: string,
* hoten: string,
* sdt: string,
* password1: string,
* };
* constructor(props) {
* super(props);
* this.state = {
* email: '',
* password: '',
* hoten: '',
* sdt: '',
* password1: '',
* };
* }
* register() {
* fetch('http://10.0.2.2:8888/WebSV/Register.php', {
* method: 'POST',
* header: {
* Accept: 'application/json',
* 'Content-Type': 'application/json',
* },
* body: JSON.stringify({
* email: this.state.email,
* password: this.state.password,
* hoten: this.state.hoten,
* sdt: this.state.sdt,
* //password1: this.state.password1,
* }),
* })
* .then(response => response.json())
* .then(responseJson => {
* if (responseJson == 'true') {
* alert('Đăng kí thành công');
* } else {
* alert('Đăng kí thất bại');
* }
* })
* .catch(error => {
* console.log(error);
* });
* }
* //onClickListener = viewId => {
* //Alert.alert('Alert', 'Button pressed ' + viewId);
* //};
* render() {
* const {navigate} = this.props.navigation;
* return (
* <View style={styles.container}>
* <View style={styles.inputContainer}>
* <TextInput
* style={styles.inputs}
* placeholder="Họ Tên"
* underlineColorAndroid="transparent"
* onChangeText={hoten => this.setState({hoten})}
* />
* <Image
* style={styles.inputIcon}
* source={{
* uri:
* 'https://img.icons8.com/color/40/000000/circled-user-male-skin-type-3.png',
* }}
* />
* </View>
* <View style={styles.inputContainer}>
* <TextInput
* style={styles.inputs}
* placeholder="Email"
* keyboardType="email-address"
* underlineColorAndroid="transparent"
* onChangeText={email => this.setState({email})}
* />
* <Image
* style={styles.inputIcon}
* source={{
* uri:
* 'https://img.icons8.com/flat\_round/40/000000/secured-letter.png',
* }}
* />
* </View>
* <View style={styles.inputContainer}>
* <TextInput
* style={styles.inputs}
* placeholder="Số điện thoại"
* keyboardType="email-address"
* underlineColorAndroid="transparent"
* onChangeText={sdt => this.setState({sdt})}
* />
* <Image
* style={styles.inputIcon}
* source={{
* uri:
* 'https://www.iconbunny.com/icons/media/catalog/product/1/6/168.9-phone-icon-iconbunny.jpg',
* }}
* />
* </View>
* <View style={styles.inputContainer}>
* <TextInput
* style={styles.inputs}
* placeholder="Mật khẩu"
* secureTextEntry={true}
* underlineColorAndroid="transparent"
* onChangeText={password => this.setState({password})}
* />
* <Image
* style={styles.inputIcon}
* source={{
* uri: 'https://img.icons8.com/color/40/000000/password.png',
* }}
* />
* </View>
* {/\* <View style={styles.inputContainer}>
* <TextInput
* style={styles.inputs}
* placeholder="Nhập lại mật khẩu"
* secureTextEntry={true}
* underlineColorAndroid="transparent"
* onChangeText={password1 => this.setState({password1})}
* />
* <Image
* style={styles.inputIcon}
* source={{
* uri: 'https://img.icons8.com/color/40/000000/password.png',
* }}
* />
* </View> \*/}
* <TouchableOpacity
* style={styles.btnByRegister}
* onPress={() => this.onClickListener('restore\_password')}>
* <Text style={styles.textByRegister}>
* Đồng ý với điểu khoảng của chúng tôi
* </Text>
* </TouchableOpacity>
* <TouchableOpacity
* style={[styles.buttonContainer, styles.loginButton]}
* onPress={() => this.register()}>
* <Text style={styles.loginText}>Đăng kí</Text>
* </TouchableOpacity>
* <TouchableOpacity
* style={styles.buttonContainer}
* onPress={() => navigate('Login')}>
* <Text style={styles.btnText}>Quay lại</Text>
* </TouchableOpacity>
* </View>
* );
* }
* }
* const resizeMode = 'center';
* const styles = StyleSheet.create({
* container: {
* flex: 1,
* justifyContent: 'center',
* alignItems: 'center',
* backgroundColor: 'rgb(255,128,192)',
* },
* inputContainer: {
* borderBottomColor: '#F5FCFF',
* backgroundColor: '#FFFFFF',
* borderRadius: 30,
* borderBottomWidth: 1,
* width: 300,
* height: 45,
* marginBottom: 20,
* flexDirection: 'row',
* alignItems: 'center',
* shadowColor: '#808080',
* shadowOffset: {
* width: 0,
* height: 2,
* },
* shadowOpacity: 0.25,
* shadowRadius: 3.84,
* elevation: 5,
* },
* inputs: {
* height: 45,
* marginLeft: 16,
* borderBottomColor: '#FFFFFF',
* flex: 1,
* },
* inputIcon: {
* width: 30,
* height: 30,
* marginRight: 15,
* justifyContent: 'center',
* },
* buttonContainer: {
* height: 45,
* flexDirection: 'row',
* justifyContent: 'center',
* alignItems: 'center',
* marginBottom: 20,
* width: 300,
* borderRadius: 30,
* backgroundColor: 'transparent',
* },
* btnByRegister: {
* height: 15,
* flexDirection: 'row',
* justifyContent: 'center',
* alignItems: 'center',
* marginVertical: 20,
* width: 300,
* backgroundColor: 'transparent',
* },
* loginButton: {
* backgroundColor: '#00b5ec',
* shadowColor: '#808080',
* shadowOffset: {
* width: 0,
* height: 9,
* },
* shadowOpacity: 0.5,
* shadowRadius: 12.35,
* elevation: 19,
* },
* loginText: {
* color: 'white',
* },
* bgImage: {
* flex: 1,
* resizeMode,
* position: 'absolute',
* width: '100%',
* height: '100%',
* justifyContent: 'center',
* },
* btnText: {
* color: 'white',
* fontWeight: 'bold',
* textShadowColor: 'rgba(0, 0, 0, 0.75)',
* textShadowOffset: {width: -1, height: 1},
* textShadowRadius: 10,
* },
* textByRegister: {
* color: 'white',
* fontWeight: 'bold',
* textAlign: 'center',
* textShadowColor: 'rgba(0, 0, 0, 0.75)',
* textShadowOffset: {width: -1, height: 1},
* textShadowRadius: 10,
* },
* });

1. **Nguyễn Văn An**

* Màn hình hiển thị danh sách sản phẩm



* Code
* import React, {Component} from 'react';
* import {
* StyleSheet,
* Text,
* TextInput,
* View,
* Image,
* TouchableOpacity,
* FlatList,
* Dimensions,
* Modal,
* ScrollView,
* AppRegistry,
* } from 'react-native';
* import {NavigationBar} from 'navigationbar-react-native';
* const ComponentLeft = () => {
* return  (
* <View style={{flex: 1, alignItems: 'flex-start'}}>
* <TouchableOpacity
* style={{justifyContent: 'center', flexDirection: 'row'}}>
* <Image
* source={require('./img/ic\_menu.png')}
* style={{
* resizeMode: 'contain',
* width: 45,
* height: 40,
* marginLeft: 20,
* alignSelf: 'center',
* }}
* />
* {/\* <Text style={{ color: 'white', }}>Back Home</Text> \*/}
* </TouchableOpacity>
* </View>
* );
* };
* const ComponentCenter = () => {
* return  (
* <View style={{flet: 1, flexDirection: 'row', padding: 20}}>
* <TextInput
* placeholder="Tìm kiếm..."
* style={{
* width: 220,
* height: 38,
* backgroundColor: 'white',
* borderRadius: 30,
* }}
* />
* <TouchableOpacity
* style={{justifyContent: 'center', flexDirection: 'row'}}>
* <View>
* <Image
* source={require('./img/ic\_cart.png')}
* style={{
* resizeMode: 'contain',
* width: 33,
* height: 40,
* marginLeft: 20,
* alignSelf: 'center',
* }}
* />
* </View>
* {/\* <Text style={{ color: 'white', }}>Back Home</Text> \*/}
* </TouchableOpacity>
* </View>
* );
* };
* const ComponentRight = () => {
* return  (
* <View style={{flex: 1, alignItems: 'flex-end'}}>
* <TouchableOpacity>
* <Image
* source={require('./img/ic\_user.png')}
* style={{
* resizeMode: 'contain',
* width: 70,
* height: 30,
* alignSelf: 'center',
* }}
* />
* </TouchableOpacity>
* </View>
* );
* };
* export default class Users extends Component {
* constructor(props) {
* super(props);
* this.state = {
* text: '',
* modalVisible: false,
* userSelected: [],
* data: [
* {
* id: 1,
* name: 'Trà sữa trân châu đường đen',
* price: '50.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* {
* id: 2,
* name: 'Trà sữa trân châu đường đen',
* price: '60.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* {
* id: 3,
* name: 'Trà sữa trân châu đường đen',
* price: '70.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* {
* id: 4,
* name: 'Trà sữa trân châu đường đen',
* price: '80.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* {
* id: 5,
* name: 'Trà sữa trân châu đường đen',
* price: '50.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* {
* id: 6,
* name: 'Trà sữa trân châu đường đen',
* price: '40.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* {
* id: 7,
* name: 'Trà sữa trân châu đường đen',
* price: '30.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* {
* id: 8,
* name: 'Trà sữa trân châu đường đen',
* price: '20.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* {
* id: 9,
* name: 'Trà sữa trân châu đường đen',
* price: '10.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* {
* id: 10,
* name: 'Trà sữa trân châu đường đen',
* price: '90.000 VND',
* image:
* 'https://www.habibi.com.vn/wp-content/uploads/2015/09/cafe-nau-den.jpg',
* about:
* 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.',
* },
* ],
* };
* }
* //set su kien khi nhan vao
* clickEventListener = item => {
* this.setState({userSelected: item}, () => {
* this.setModalVisible(true);
* });
* };
* setModalVisible(visible) {
* this.setState({modalVisible: visible});
* }
* componentDidMount() {
* return fetch('http:/10.0.2.2:8888/WebSV/Select\_SanPham.php')
* .then(response => response.json())
* .then(responseJson => {
* this.setState(
* {
* isLoading: false,
* dataSource: responseJson,
* },
* function() {},
* );
* })
* .catch(error => {
* console.error(error);
* });
* }
* render() {
* const {navigate} = this.props.navigation;
* return (
* <View style={styles.container}>
* <NavigationBar
* componentLeft={() => <ComponentLeft />}
* componentCenter={() => <ComponentCenter />}
* componentRight={() => <ComponentRight />}
* navigationBarStyle={{backgroundColor: '#778899'}}
* statusBarStyle={{
* barStyle: 'light-content',
* backgroundColor: '#215e79',
* }}
* />
* {/\* Danh sach thong tin do uong \*/}
* <FlatList
* style={styles.userList}
* columnWrapperStyle={styles.listContainer}
* data={this.state.dataSource}
* a
* renderItem={({item}) => {
* return (
* // set su kien khi nhan vao
* <TouchableOpacity
* style={styles.card}
* onPress={() => {
* this.clickEventListener(item);
* }}>
* <Image style={styles.image} source={{uri: item.image}} />
* <View style={styles.cardContent}>
* <Text style={styles.name}>Mã sản phẩm: {item.masp}</Text>
* <Text style={styles.name}>Tên: {item.title}</Text>
* <Text style={styles.price}>Giá: {item.price}</Text>
* {/\* Button theo doi \*/}
* <TouchableOpacity
* style={styles.followButton}
* onPress={() => this.clickEventListener(item)}>
* <Text style={styles.followButtonText}>Xem</Text>
* </TouchableOpacity>
* </View>
* </TouchableOpacity>
* );
* }}
* />
* <Modal
* animationType={'fade'}
* transparent={true}
* onRequestClose={() => this.setModalVisible(false)}
* visible={this.state.modalVisible}>
* <View style={styles.popupOverlay}>
* <View style={styles.popup}>
* <View style={styles.popupContent}>
* {/\* Hien thi len thong tin cua tung nguoi theo dang scrollview \*/}
* <ScrollView contentContainerStyle={styles.modalInfo}>
* <Image
* style={styles.image}
* source={{uri: this.state.userSelected.image}}
* />
* <Text style={styles.name}>
* {this.state.userSelected.name}
* </Text>
* <Text style={styles.price}>
* {this.state.userSelected.price}
* </Text>
* <Text style={styles.about}>
* {this.state.userSelected.about}
* </Text>
* </ScrollView>
* </View>
* {/\* Button close\*/}
* <View style={styles.popupButtons}>
* <TouchableOpacity
* onPress={() => {
* this.setModalVisible(false);
* }}
* style={styles.btnClose}>
* <Text style={styles.txtClose}>Close</Text>
* </TouchableOpacity>
* </View>
* </View>
* </View>
* </Modal>
* <View style={styles.body}>
* <View style={styles.item}>
* <TouchableOpacity
* onPress={() => this.props.navigation.navigate('Touch')}>
* <View style={styles.iconContent}>
* <Image
* style={styles.icon}
* source={require('./img/ic\_home.png')}
* />
* </View>
* <View>
* <Text style={styles.info}>Home</Text>
* </View>
* </TouchableOpacity>
* <TouchableOpacity
* onPress={() => this.props.navigation.navigate('Cart')}>
* <View style={styles.iconContent}>
* <Image
* style={styles.icon}
* source={require('./img/ic\_cart.png')}
* />
* </View>
* <View>
* <Text style={styles.info}>Cart</Text>
* </View>
* </TouchableOpacity>
* <TouchableOpacity onPress={() => navigate('InsertSanPham')}>
* <View style={styles.iconContent}>
* <Image
* style={styles.icon}
* source={require('./img/ic\_search.png')}
* />
* </View>
* <View>
* <Text style={styles.info}>Thêm</Text>
* </View>
* </TouchableOpacity>
* <TouchableOpacity>
* <View style={styles.iconContent}>
* <Image
* style={styles.icon}
* source={require('./img/ic\_user.png')}
* />
* </View>
* <View>
* <Text style={styles.info}>Contact</Text>
* </View>
* </TouchableOpacity>
* </View>
* </View>
* </View>
* );
* }
* }
* //set style cho tung doi tuong
* const styles = StyleSheet.create({
* inputIcon: {
* marginLeft: 15,
* justifyContent: 'center',
* },
* info: {
* fontSize: 14,
* marginLeft: 15,
* marginTop: 35,
* paddingLeft: 10,
* paddingRight: 10,
* color: '#FFFFFF',
* },
* body: {
* backgroundColor: '#778899',
* height: 60,
* alignItems: 'center',
* },
* item: {
* flexDirection: 'row',
* },
* icon: {
* padding: 5,
* marginRight: 19,
* paddingLeft: 9,
* paddingRight: 10,
* width: 30,
* height: 30,
* marginTop: 10,
* },
* infoContent: {
* flex: 1,
* alignItems: 'flex-start',
* },
* iconContent: {
* flex: 1,
* alignItems: 'flex-end',
* },
* title: {
* marginTop: 13,
* marginLeft: 110,
* height: 35,
* width: 200,
* textAlign: 'center',
* flexDirection: 'row',
* justifyContent: 'center',
* alignItems: 'center',
* borderRadius: 50,
* color: 'black',
* fontSize: 16,
* backgroundColor: '#CC33FF',
* },
* container: {
* flex: 1,
* backgroundColor: '#eeeeee',
* },
* header: {
* backgroundColor: '#00CED1',
* height: 200,
* },
* headerContent: {
* padding: 30,
* alignItems: 'center',
* flex: 1,
* },
* detailContent: {
* top: 80,
* height: 500,
* width: Dimensions.get('screen').width - 90,
* marginHorizontal: 30,
* flexDirection: 'row',
* position: 'absolute',
* backgroundColor: '#ffffff',
* },
* userList: {
* flex: 1,
* },
* cardContent: {
* marginLeft: 20,
* marginTop: 10,
* },
* image: {
* width: 90,
* height: 90,
* borderRadius: 45,
* },
* card: {
* shadowColor: '#00000021',
* shadowOffset: {
* width: 0,
* height: 6,
* },
* shadowOpacity: 0.37,
* shadowRadius: 7.49,
* elevation: 12,
* marginVertical: 10,
* marginHorizontal: 20,
* backgroundColor:'white',
* flexBasis: '46%',
* padding: 10,
* flexDirection: 'row',
* },
* name: {
* fontSize: 18,
* flex: 1,
* alignSelf: 'center',
* color:'#008080',
* fontWeight: 'bold',
* },
* position: {
* fontSize: 14,
* flex: 1,
* alignSelf: 'center',
* color: '#696969',
* },
* about: {
* marginHorizontal: 10,
* },
* followButton: {
* marginTop: 10,
* height: 35,
* width: 100,
* flexDirection: 'row',
* justifyContent: 'center',
* alignItems: 'center',
* borderRadius: 30,
* backgroundColor: '#00BFFF',
* },
* followButtonText: {
* color: '#FFFFFF',
* fontSize: 20,
* },
* /\*\*\*\*\*\*\*\*\*\*\*\* modals \*\*\*\*\*\*\*\*\*\*\*\*/
* popup: {
* backgroundColor: 'white',
* marginTop: 80,
* marginHorizontal: 20,
* borderRadius: 7,
* },
* popupOverlay: {
* backgroundColor: '#00000057',
* flex: 1,
* marginTop: 30,
* },
* popupContent: {
* //alignItems: 'center',
* margin: 5,
* height: 250,
* },
* popupHeader: {
* marginBottom: 45,
* },
* popupButtons: {
* marginTop: 15,
* flexDirection: 'row',
* borderTopWidth: 1,
* borderColor: '#eee',
* justifyContent: 'center',
* },
* popupButton: {
* flex: 1,
* marginVertical: 16,
* },
* btnClose: {
* height: 20,
* backgroundColor: '#20b2aa',
* padding: 20,
* },
* modalInfo: {
* alignItems: 'center',
* justifyContent: 'center',
* },
* });

# **CHƯƠNG V: KẾT LUẬN KIẾN NGHỊ**

Trong thời gian học tập ở trương và đặt biệt hơn là môn lập trình di động 3, với sự giúp đở tận tình của thầy Thái, nhóm chúng em đã nắm được nhiều kiến thức về React native, làm việc nhóm, khả năng lắng nghe và học hỏi,… và học được nhiều kiến thức mới trong quá trình làm việc.

Em xin chân thành cảm ơn thầy Thái đã giúp đở nhóm chúng em trong môn học này.