**TRƯỜNG CAO ĐẲNG CÔNG NGHỆ THỦ ĐỨC**



**KHOA CÔNG NGHỆ THÔNG TIN**

**🙜🕮🙞**

****

**BÁO CÁO MÔN HỌC**

**Lập trình di động 3**

**Đề tài: App bán Laptop**

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

**Nhóm sinh viên thực hiện**

* + - * **Nguyễn Hữu Phú 17211TT3116**
      * **Võ Xuân Hoàng 17211TT3684**
      * **Võ Văn Chiến 17211TT3360**

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

**Mục Lục**

[**Chương 1: Tổng Quan 3**](#_Toc23401907)

[**Chương 2: ES6 16**](#_Toc23401908)

[**1. Let (Nguyễn Hữu Phú) 16**](#_Toc23401909)

[**2. Var (Nguyễn Hữu Phú) 16**](#_Toc23401910)

[**3. Arrow Function: (Võ Xuân Hoàng) 17**](#_Toc23401911)

[**4. Object (Võ Văn Chiến) 20**](#_Toc23401912)

[**Chương 3: Component 26**](#_Toc23401913)

[**1. Props (Nguyễn Hữu Phú) 26**](#_Toc23401914)

[**2. State (Nguyễn Hữu Phú) 28**](#_Toc23401915)

[**3. Check Box (Nguyễn Hữu Phú) 30**](#_Toc23401916)

[**4. Style (Võ Xuân Hoàng) 32**](#_Toc23401917)

[**5. Layout with Flexbox (Võ Xuân Hoàng) 33**](#_Toc23401918)

[**6. Handling Text Input (Võ Xuân Hoàng) 34**](#_Toc23401919)

[**7. Handling Touches (Võ Văn Chiến) 35**](#_Toc23401920)

[**8. Using a ScrollView (Võ Văn Chiến) 38**](#_Toc23401921)

[**9. Progressbar (Võ Văn Chiến) 41**](#_Toc23401922)

[**10. ListView (Nguyễn Hữu Phú) 43**](#_Toc23401923)

[**11. Picker (Võ Xuân Hoàng) 44**](#_Toc23401924)

[**12. Switch Button (Võ Văn Chiến) 47**](#_Toc23401925)

[**Chương 4: Xây Dựng Ứng Dụng 72**](#_Toc23401926)

[**1. Màn hình Login (Võ Văn Chiến) 72**](#_Toc23401927)

[**2. Màn hình sản phẩm (Nguyễn Hữu Phú) 75**](#_Toc23401928)

[**3. Màn hình thêm sản phẩm (Võ Xuân Hoàng) 79**](#_Toc23401929)

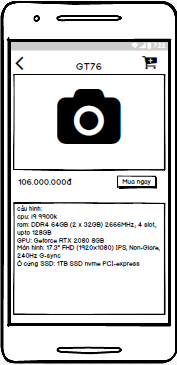
[**4. Màn hình sửa sản phẩm (Võ Xuân Hoàng) 82**](#_Toc23401930)

[**5. Màn hình xóa sản phẩm (Võ Văn Chiến) 86**](#_Toc23401931)

[**Chương 5: Kết Luận Và Kiến Nghị 88**](#_Toc23401932)

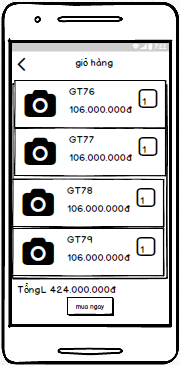
# Chương 1: Tổng Quan

1. **Trang chi tiết(Võ Xuân Hoàng)**

****

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Giao diện trang chi tiết** | **Giao diện người quản lý gồm 3 button, 1 image và 3 text:**   * **Button giỏ hàng** * **Button mua** * **Button back** * **Image sản phẩm** * **Textbox mô tả** * **Textbox tiêu đề** * **Textbox giá** | **N/A** | **N/A** |
| **Button giỏ hàng** | **Liên kết tới activity giỏ hang người dùng** | **Nhấn vào icon giỏ hàng** | **Một activity sẽ xuất hiện được dung để lưu sản phẩm cần mua của người dúng** |
| **Button mua ngay** | **Liên kết với trang giỏ hang của người dùng** | **Nhấn vào button mua ngay** | **Sản phẩm sau khi bấm mua ngay sẽ tự động thêm vào giỏ hang của người dung.** |
| **Image sản phẩm** | **Cho phép người dùng xem trước sản phẩm cần mua** | **Nhấn vào hình** | **Hình sẽ được phóng to ra nhằm mục đích cho người dùng xem trước** |
| **Text giá, text mô tả, text tiêu đề** | **Mô tả chi tiết sản phẩm cho người dùng một cách chân thật nhất** | **N/A** | **Hiện thông số của sản phẩm như tên, giá cấu hình sản phẩm.** |

1. **Trang giỏ hàng: (Võ Xuân Hoàng)**

****

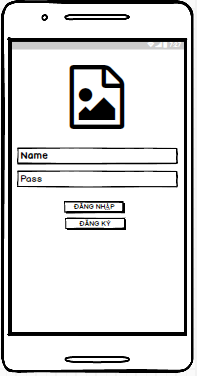
|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Giao diện giỏ hàng** | **Giao diện người quản lý gồm :**   1. **Listview chứa các mặt hàng** 2. **Item thông tin sản phẩm** 3. **Button mua ngay** | **N/A** | **N/A** |
| **Listview sản phẩm** | **Chứa các item mặt hàng** | **N/a** | **Một list chứa tất cả các sản phẩm cần mua của người dùn cho phép người dùng xem lại các sản phẩm mình đã chọn mua** |
| **Item mặt hàng** | **Các mặt hang trong listview** | **Nhấn vào mặt hang để xem** | **Một của sổ trang chi tiết của mặt hàng đó**  **S3 được bật lên cho phép người dùng xem lại thông số kĩ thuật của sản phẩm.** |
| **Button mua ngay** | **Button thực hiện giao dịch** | **Nhấn vào button mua** | **Khi người mua nhấn vào button này sẽ thực hiện giao dịch thanh toán** |

1. **Trang đăng kí(Võ Xuân Hoàng)**

****

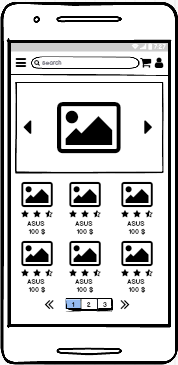
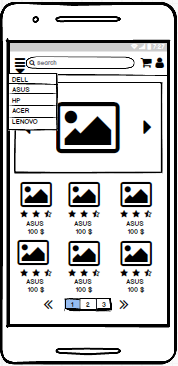
|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Giao diện đăng kí** | **Giao diện người quản lý gồm:**   1. **Textbox họ tên** 2. **Textbox email** 3. **Textbox mật khẩu vs nhập mật khẩu** 4. **Textbox số điên thoại** 5. **Textbox địa chỉ** 6. **Button tạo** | **Nhập thông tin vào textbox và nhấn button tạo** | **Cho phép người dùng tạo ra tài khoản với những thông tin họ đạ nhập.** |

1. **Trang đăng nhập (Nguyễn Hữu Phú)**

****

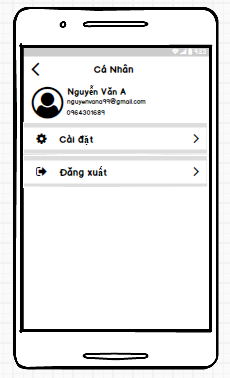
|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình đăng nhập** | **Màn hình đăng nhập gồm:**   * **ImageView Logo** * **RadioButton quản lý** * **RadioButton cửa hàng** * **EditText Name** * **EditText Pass** * **Button đăng nhập** * **Button đăng ký** | **N/A** | **N/A** |
| **RadioButton Quản lý** | **Người dùng chọn vào radio này để vào giao diện của quản lý cửa hàng** | **Chạm vào radio Quản lý** | **Hiển thị Radio khi người dùng chọn** |
| **RadioButton Cửa hàng** | **Người dùng chọn vào radio này để vào giao diện của cửa hàng** | **Chạm vào radio Cửa hàng** | **Hiển thị Radio khi người dùng chọn** |
| **EditText Name** | **Cho phép người dùng nhập thông tin name** | **Chạm vào EditText để nhập name** | **Hiển thị con trỏ chuột vào ô name cho phép người dùng nhập vào ô** |
| **EditText Pass** | **Cho phép người dùng nhập thông tin Pass (Thông tin đưa vào dưới dạng ẩn)** | **Chạm vào EditText để nhập pass** | **Hiển thị con trỏ chuột vào ô pass cho phép người dùng nhập vào ô** |
| **ImageView Logo** | **Hiển thị logo của ứng dụng** | **N/A** | **N/A** |
| **Button đăng nhập** | **Cho phép người dùng đăng nhập vào ứng dụng** | **Chạm vào nút đăng nhập** | 1. **Lấy dữ liệu từ EditText Name và EditText Pass** 2. **So sánh dữ liệu vừa lấy với dữ liệu trong Database** 3. **Nếu trùng khớp thì sẽ chuyển hướng đến màn hình “Thông tin cửa hàng”.**   **Nếu không trùng khớp thì sẽ hiển thị thông báo đăng nhập không thành công** |
| **Button đăng ký** | **Cho phép người dùng chuyển đến màn hình đăng ký tài khoản** | **Chạm vào button để đăng ký** | **Hiển thị màn hình đăng ký tài khoản** |

1. **Trang chủ(Nguyễn Hữu Phú)**

** **

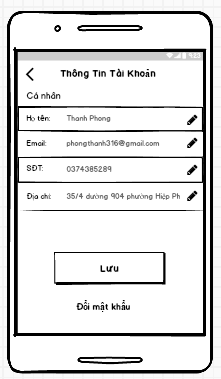
|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Màn hình đăng nhập** | **Màn hình đăng nhập gồm:**   * **ImageView Menu** * **ImageView Giỏ hàng** * **ImageView User** * **Image hiển thị sản phẩm** * **Recyclerview danh sách sản phẩm** * **EditText tìm kiếm** * **Button phân trang** | **N/A** | **N/A** |
| **ImageView Menu** | **Hiển thị icon menu của ứng dụng, mở ra các thương hiệu để người dùng chọn** | **Chạm vào icon menu** | **Hiển thị ra listView các thương hiệu laptop** |
| **ImageView Giỏ hàng** | **Hiển thị icon giỏ hàng của ứng dụng** | **Chạm vào icon giỏ hàng** | **Chuyển sang màn hình giỏ hàng** |
| **ImageView User** | **Hiển thị icon user của ứng dụng** | **Chạm vào icon user** | **Show thông tin của user, đăng xuất tài khoản** |
| **Image hiển thị sản phẩm** | **Hiển thị sản phẩm được chọn** | **Chạm vào sản phẩm dưới danh sách sản phẩm** | **Hiển thị sản phẩm theo danh sách sản phẩm** |
| **Recyclerview danh sách sản phẩm** | **Hiển thị danh sách sản phẩm** | **Chọn sản phẩm cần hiển thị** | **Load dữ liệu sản phẩm ra danh sách sản phẩm** |
| **EditText tìm kiếm** | **Tìm kiếm theo tên sản phẩm** | **Nhập sản phẩm cần tìm** | **Load sản phẩm ra theo tên sản phẩm cần tìm** |
| **Button phân trang** | **Chọn trang để qua** | **Chạm vào trang cần qua** | **Load sản phẩm theo trang** |

1. **Giao diện cá nhân :(Võ Văn Chiến)**

****

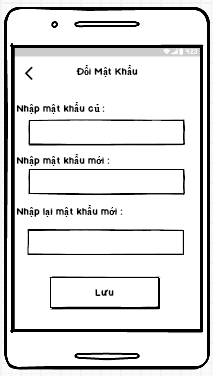
|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Giao diện cá nhân** | **Giao diện gồm có 1 image, 3 textview, 3 button :**  **- Image cá nhân**  **- Textview name**  **- Textview email**  **- Textview sdt**  **- Button cài đặt**  **- Button đăng xuất**  **- Button back** |  |  |
| **Button cài đặt** | **Sử dụng để chuyển qua màn hình cài đặt** | **Nhấn vào button cài đặt** | **Chuyển sang màn hình cài đặt** |
| **Button back** | **Sử dụng để chuyển về màn hình trước** | **Nhấn vào button back** | **Chuyển về màn hình trước** |
| **Button đăng suất** | **Sử dụng để chuyển về màn hình login** | **Nhấn vào button đăng suất** | **Chuyển về màn hình login** |

1. **Giao diện thông tin tài khoản:(Võ Văn Chiến)**

****

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Giao diện thông tin tài khoản** | **Giao diện gồm có 6 textview, 4 edittext, 7 button :**  **- Textview title**  **- Textview cá nhân**  **- Textview họ tên**  **- Textview email**  **- Textview sdt**  **- Textview địa chỉ**  **- Edittext họ tên**  **- Edittext email**  **- Edittext sdt**  **- Edittext địa chỉ**  **- Button họ tên**  **- Button email**  **- Button sdt**  **- Button địa chỉ**  **- Button lưu**  **- Button đổi mật khẩu**  **- Button back** |  |  |
| **Button họ tên** | **Sử dụng để nhập tên** | **Nhấ7n vào button họ tên** |  |
| **Button email** | **Sử dụng để nhập email** | **Nhấn vào button email** |  |
| **Button sdt** | **Sử dụng để nhập sdt** | **Nhấn vào button sdt** |  |
| **Button địa chỉ** | **Sử dụng để nhập địa chỉ** | **Nhấn vào button địa chỉ** |  |
| **Button lưu** | **Sử dụng để lưu thông tin** | **Nhấn vào button lưu** |  |
| **Button đổi mật khẩu** | **Sử dụng để đổi mật khẩu** | **Nhấn vào button đổi mật khẩu** | **Chuyển sang màn hình đổi mật kahu36** |
| **Button back** | **Sử dụng để trở về màn hình trước** | **Nhấn vào button back** | **Chuyển về màn hình trước** |

1. **Giao diện đổi mật khẩu:(Võ Văn Chiến)**

****

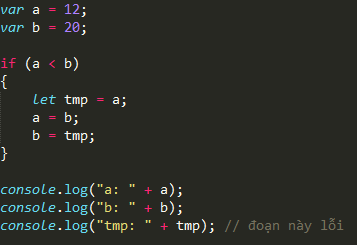
|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Action** | **Response** |
| **Giao diện đổi mật khẩu** | **Giao diện gồm có 4 textview, 3 textbox, 2 button :**  **- Textview title**  **- Textview mật khẩu cũ**  **- Textview mật khẩu mới**  **- Textview mật khẩu mới 2**  **- Textbox mật khẩu cũ**  **- Textbox mật khẩu mới**  **- Textbox mật khẩu mới 2**  **- Button lưu**  **- Button back** |  |  |
| **Textbox mật khẩu cũ** | **Sử dụng để nhập mật khẩu củ** | **Nhấn vào textbox mật khẩu củ** |  |
| **Textbox mật khẩu mới** | **Sử dụng để nhập mật khẩu mới** | **Nhấn vào textbox mật khẩu mới** |  |
| **Textbox mật khẩu mới 2** | **Sử dụng để nhập lại mật khẩu mới** | **Nhấn vào textbox mật khẩu mới 2** |  |
| **Button lưu** | **Sử dụng để lưu mật khẩu mới** | **Nhấn vào button lưu** |  |
| **Button back** | **Sử dụng để trở về màn hình trước** | **Nhấn vào button back** |  |

# Chương 2: ES6

1. **Let (Nguyễn Hữu Phú)**

* **let tạo ra một biến chỉ có thể truy cập được trong block bao quanh nó**

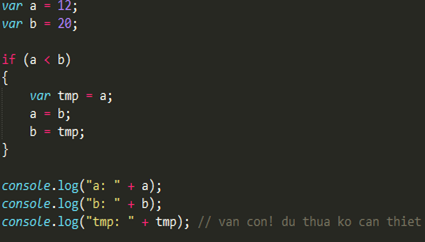
**Sử dụng let:**

****

1. **Var (Nguyễn Hữu Phú)**

* **var tạo ra một biến có phạm vi truy cập xuyên suốt function chứa nó.**

**Sử dụng var:**

****

**Const:**

**const dùng để khai báo một hằng số - là một giá trị không thay đổi được trong suốt quá trình chạy.**

**Ví dụ:**

**const A = 5;**

**A = 10; *// Lỗi Uncaught TypeError: Assignment to constant variable***

1. **Arrow Function: (Võ Xuân Hoàng)**

* **Arrow function - còn được gọi là "fat arrow", là cú pháp được mượn từ CoffeeScript (một ngôn ngữ chuyển tiếp), cú pháp này là cách ngắn gọn hơn dùng để viết function. Ở đây sử dụng kí tự =>, trông giống như một mũi tên "béo". Arrow function là một hàm vô danh và nó thay đổi cách this bind đến function. Arrow function làm code của ta trông ngắn gọn hơn, giúp đơn giản hóa function scoping cũng như từ khóa this. Arrow function hoạt động tương tự như Lambdas trong các ngôn ngữ khác như C # hay Python. Bằng cách sử dụng arrow function, chúng ta tránh được việc phải gõ từ khoá function, return và dấu ngoặc nhọn.**

**Cách dùng arrow function**

**Có khá nhiều cú pháp có thể dùng với arrow function.**[**EcmaScript.org**](http://wiki.ecmascript.org/doku.php?id=harmony:arrow_function_syntax)**,**[**MDN**](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Functions/Arrow_functions)**có liệt kê một danh sách đầy đủ các cú pháp có thể được dùng. Ở đây tôi sẽ đề cập đến những cú pháp phổ biến nhất.**

* **Trong trường hợp nhiều tham số**

**// (param1, param2, paramN) => expression**

**// ES5**

**var multiply = function(x, y) {**

**return x \* y;**

**};**

**// ES6**

**var multiply = (x, y) => { return x \* y };**

**Ví dụ trên cho cùng một kết quả, tuy nhiên cú pháp với arrow function tốn ít dòng mã hơn. Trong trường hợp chỉ có một biểu thức thì không cần tới dấu ngoặc nhọn: Ví dụ trên có thể viết lại như sau:**

**var multiply = (x, y) => x \* y ;**

**Trong trường hợp 1 tham số**

**Dấu ngoặc đơn là optional khi chỉ có một tham số**

**//ES5**

**var phraseSplitterEs5 = function phraseSplitter(phrase) {**

**return phrase.split(' ');**

**};**

**//ES6**

**var phraseSplitterEs6 = phrase => phrase.split(" ");**

**console.log(phraseSplitterEs6("ES6 Awesomeness")); // ["ES6", "Awesomeness"]**

**Trong trường hợp không có tham số**

**Dấu ngoặc đơn là bắt buộc khi không có tham số.**

**//ES5**

**var docLogEs5 = function docLog() {**

**console.log(document);**

**};**

**//ES6**

**var docLogEs6 = () => { console.log(document); }**

**docLogEs6(); // #document... <html> ….**

**Khi nào thì nên sử dụng arrow function**

**Một usecase phổ biến áp dung arrow function là thao tác mảng, thông thường là khi dùng map hoặc reduce. Có 1 mảng như sau:**

**var smartPhones = [**

**{ name: 'iphone', price: 649 },**

**{ name: 'Galaxy S6', price: 576 },**

**{ name: 'Galaxy Note 5', price: 489 }**

**];**

**Để lấy 1 mảng toàn tên các smart phones, ta thực hiện:**

**// ES5**

**console.log(smartPhones.map(function(smartPhone){**

**return smartPhone.price;**

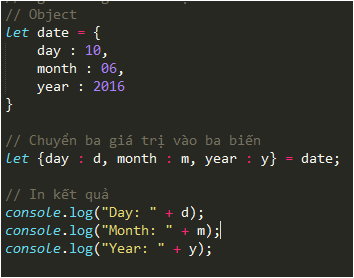
**}); // [649, 576, 489]**

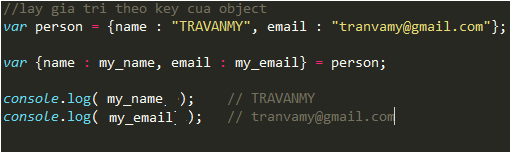
**Với arrow function, có thể viết rút gọn lại thành**

**// ES6**

**console.log(smartPhones.map(smartPhone => smartPhone.price)); // [649, 576, 489]**

1. **Object (Võ Văn Chiến)**

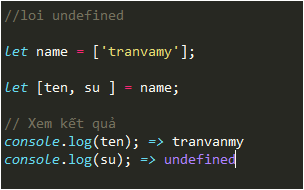
****

**-Lấy giá trị theo key Object**

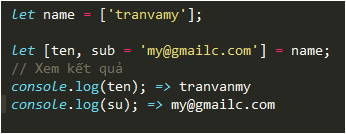
**Nhưng nếu bạn cố ý truy cập vào một key không tồn tại thì sẽ bị lỗi undefined.**

****

**Hay**

****

**Khi gặp những lỗi thế này thì các bạn chỉ cần gán luôn giá trị cho biến trong mảng là ok.**

****

**Trong ES6 đã hỗ trợ chúng ta khai báo một đối tượng theo chuẩn OOP, bằng cách sử dụng từ khóa class.**

**VD: Đối với VD trên chúng ta có thể chuyển sang dạng ES6 thành như sau:**

**class Employee {**

**setName (name) {**

**this.name = name;**

**}**

**getName () {**

**return this.name;**

**}**

**setAge (age) {**

**this.age = age;**

**}**

**getAge () {**

**return this.age;**

**}**

**};**

## **Bài tập es6 của các thành viên**

* Nguyễn Hữu Phú
* Code:

// Arrow function

var PhepTinh = (a, b) => {

    var result = a + b;

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

    var result1 = a - b;

    console.log("Phép Trừ: " + result1);

    var result2 = a \* b;

    console.log("Phép Nhân: " + result2);

    var result3 = a / b;

    console.log("Phép Chia: " + result3);

};

PhepTinh(5, 5);

//Collection Maps

let map = new Map([

    ["Name", "Nguyen huu Phu"],

    ["Email", "nhp@gmail.com"],

    ["Address", "HCM"],

]);

//kiem tra phan tu

console.log(map.has('domain')); // k co phan tu domain

console.log(map.has('Name')); // co phan tu Name

//Xóa phan tu

//map.clear();

console.log(map);

* Chạy console:

Phép Cộng: 10

Phép Trừ: 0

Phép Nhân: 25

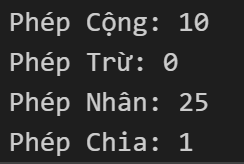
Phép Chia: 1

false

true

> Map(3) {}

* Võ Văn Chiến
* Code:
* class NumberObject{
* constructor(number) {
* this.number = number;
* }
* isOdd() {
* if (this.number % 2 !== 0)
* return true;
* return false;
* }
* divisibleByThree() {
* if (this.number % 3 === 0)
* return true;
* return false;
* }
* getNumber() {
* return this.number;
* }
* }
* let numbersArr = [
* new NumberObject(0),
* new NumberObject(1),
* new NumberObject(2),
* new NumberObject(3),
* new NumberObject(4),
* new NumberObject(5),
* new NumberObject(6),
* new NumberObject(7),
* new NumberObject(8),
* new NumberObject(9),
* ];
* let odds = [];
* let evens = [];
* let numbersDivisibleByThree = [];
* numbersArr.map(value => {
* if (value.isOdd()) {
* odds.push(value.getNumber());
* } else if (!value.isOdd()) {
* evens.push(value.getNumber());
* }
* if (value.getNumber() !== 0 && value.divisibleByThree()) {
* numbersDivisibleByThree.push(value.getNumber());
* }
* });
* console.log('-------Số lẻ-------');
* console.log(odds);
* console.log('-------Số chẵn-------');
* console.log(evens);
* console.log('-------Số chia hết cho 3-------');
* console.log(numbersDivisibleByThree);
* Chạy console:
* -------Số lẻ-------
* Array(5) [1, 3, 5, 7, 9]
* -------Số chẵn-------
* Array(5) [0, 2, 4, 6, 8]
* -------Số chia hết cho 3-------
* Array(3) [3, 6, 9]
* Võ Xuân Hoàng
* Code:
* // Arrow function
* var PhepTinh = (a, b) => {
* var result = a + b;
* console.log("Phép Cộng: " + result);
* var result1 = a - b;
* console.log("Phép Trừ: " + result1);
* var result2 = a \* b;
* console.log("Phép Nhân: " + result2);
* var result3 = a / b;
* console.log("Phép Chia: " + result3);
* };
* PhepTinh(5, 5);
* Chạy console:



# Chương 3: Component

1. **Props (Nguyễn Hữu Phú)**

* **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'**

**};**

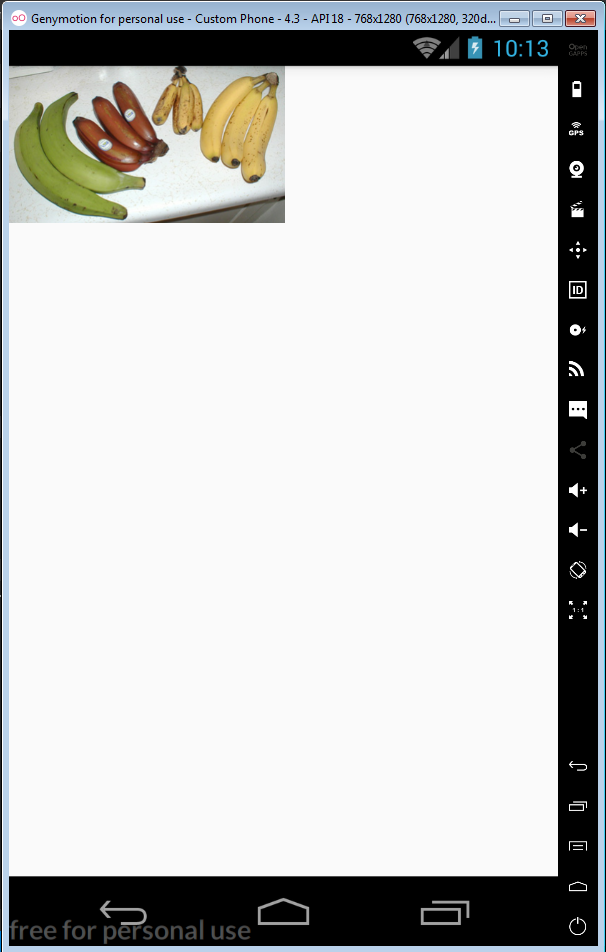
**return (**

**<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' />**

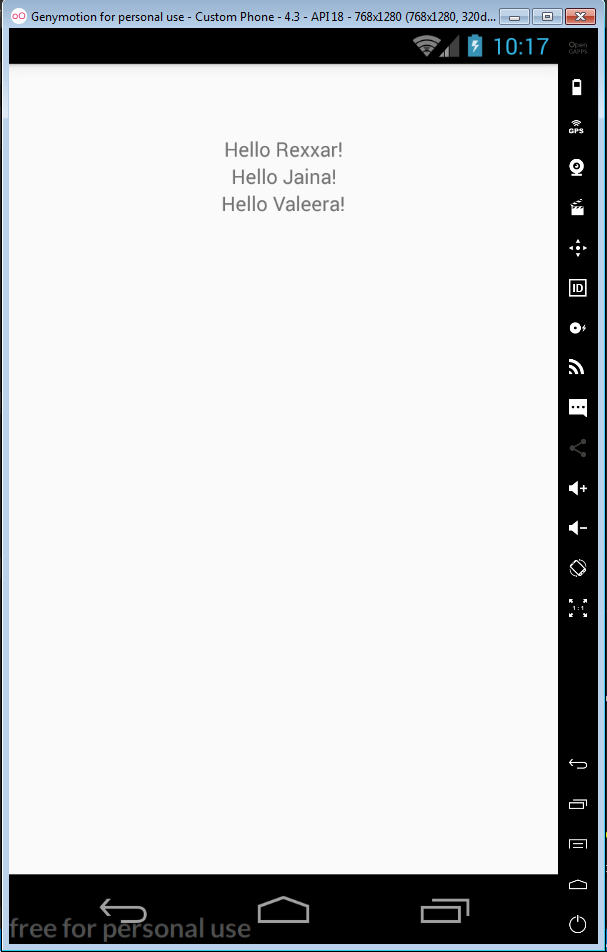
**<Greeting name='Valeera' />**

**</View>**

**);**

**}**

**}**

****

1. **State (Nguyễn Hữu Phú)**

* **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' />**

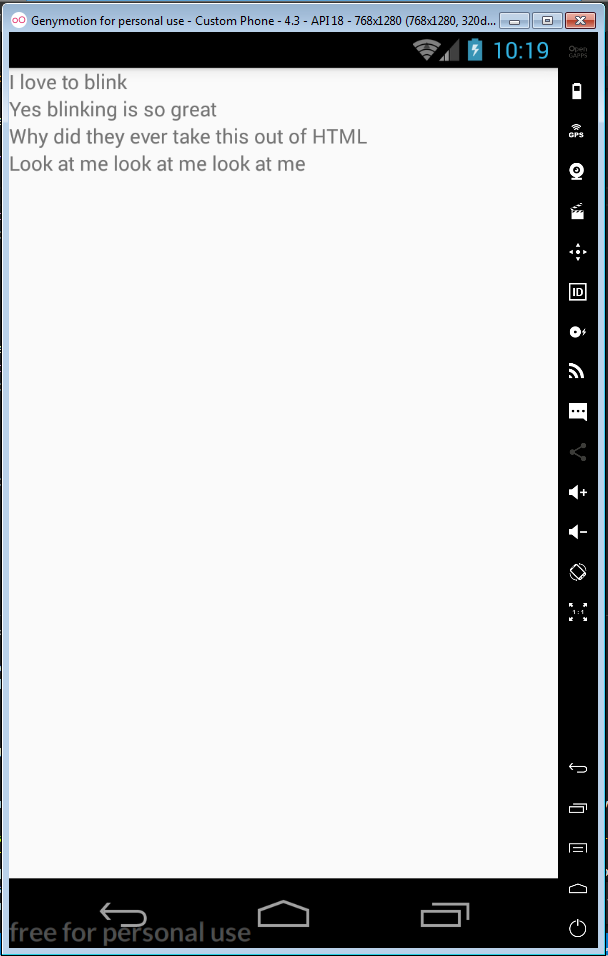
**<Blink text='Look at me look at me look at me' />**

**</View>**

**);**

**}**

**}**

****

1. **Check Box (Nguyễn Hữu Phú)**

**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>**

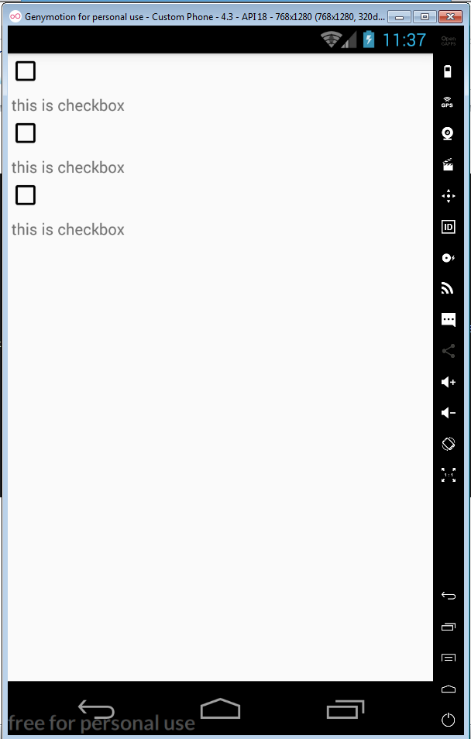
**</View >**

**)})**

**);**

**}**

**}**

****

1. **Style (Võ Xuân Hoàng)**

* **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>**

**);**

**}**

**}**

****

1. **Layout with Flexbox (Võ Xuân Hoàng)**

* **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>**

**);**

**}**

**};**

****

1. **Handling Text Input (Võ Xuân Hoàng)**

* **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(' ')}**

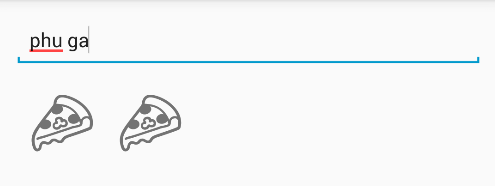
**</Text>**

**</View>**

**);**

**}**

**}**

****

1. **Handling Touches (Võ Văn Chiến)**

**- 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: {**

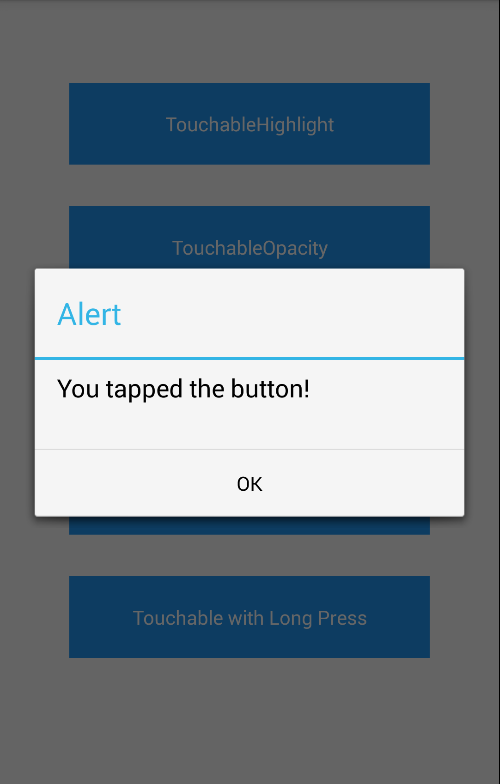
**textAlign: 'center',**

**padding: 20,**

**color: 'white'**

**}**

**});**

****

1. **Using a ScrollView (Võ Văn Chiến)**

**- 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}} />**

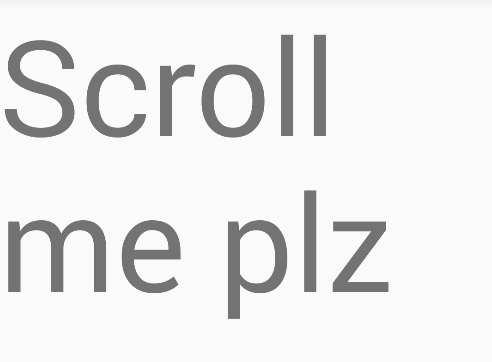
**<Text style={{fontSize:80}}>React Native</Text>**

**</ScrollView>**

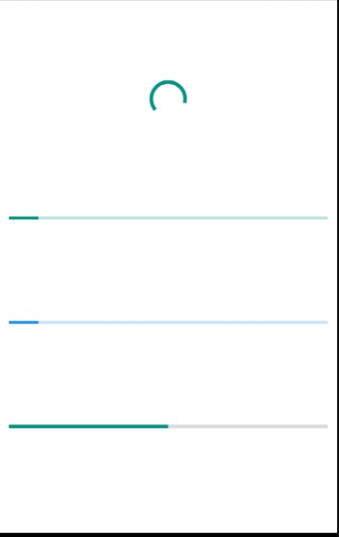
**);**

**}**

**}**

****

1. **Progressbar (Võ Văn Chiến)**

* **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,**

**},**

**});**

1. **ListView (Nguyễn Hữu Phú)**

**- 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>}**

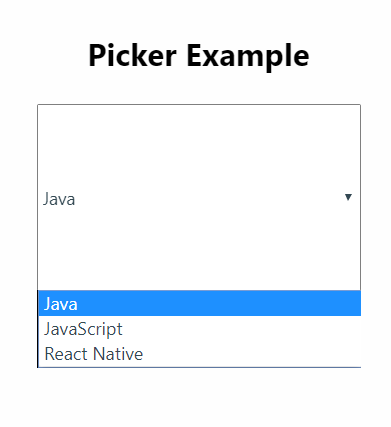
**/>**

**);**

**}**

**}**

1. **Picker (Võ Xuân Hoàng)**

****

**- 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%",**

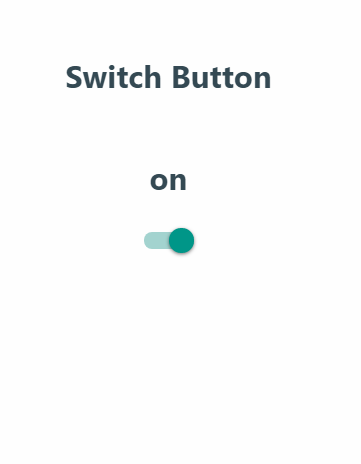
**color: '#344953',**

**justifyContent: 'center',**

**}**

**})**

## **12. Switch Button (Võ Văn Chiến)**

**- 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',**

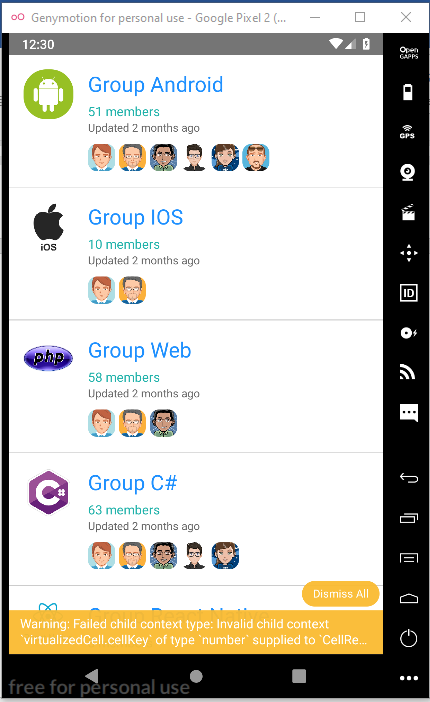
**color: '#344953'**

**}**

**})**

**Bài tập Exe1 và Exe3 của các thành viên**

* Nguyễn Hữu Phú
* Code:
* export default class Groups extends Component {
* constructor(props) {
* super(props);
* this.state = {
* data:[
* {
* id:3,
* image: "http://seoiclick.com//content/uploads/1420854296\_os\_android.png",
* name:"Group Android",
* countMembers:51,
* members:[
* "https://bootdey.com/img/Content/avatar/avatar6.png",
* "https://bootdey.com/img/Content/avatar/avatar1.png",
* "https://bootdey.com/img/Content/avatar/avatar2.png",
* "https://bootdey.com/img/Content/avatar/avatar7.png",
* "https://bootdey.com/img/Content/avatar/avatar3.png",
* "https://bootdey.com/img/Content/avatar/avatar4.png"
* ]
* },
* {
* id:2,
* image: "data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAOEAAADhCAMAAAAJbSJIAAAAhFBMVEX///8mJiYiIiIfHx8cHBw4ODjj4+MaGho8PDz6+vooKCgYGBj19fUAAADt7e0WFhbb29sxMTHU1NQPDw9PT09xcXHi4uLDw8OampphYWELCwukpKRBQU gCBog+uvYPUzQgADt8H6Cpbz3XIahL492gIrwyFYUvy35UdkjjUxCEQNz1un7rwQjIc19d ",
* name:"Group IOS",
* countMembers:10,
* members:[
* "https://bootdey.com/img/Content/avatar/avatar6.png",
* "https://bootdey.com/img/Content/avatar/avatar1.png",
* ]
* },
* {
* id:4,
* image: "https://cdn2.iconfinder.com/data/icons/crystalproject/crystal\_project\_256x256/apps/php.png",
* name:"Group Web",
* countMembers:58,
* members:[
* "https://bootdey.com/img/Content/avatar/avatar6.png",
* "https://bootdey.com/img/Content/avatar/avatar1.png",
* "https://bootdey.com/img/Content/avatar/avatar2.png"
* ]
* },
* {
* id:5,
* image: "http://dccoder.com/wp-content/uploads/2018/07/c.png",
* name:"Group C#",
* countMembers:63,
* members:[
* "https://bootdey.com/img/Content/avatar/avatar6.png",
* "https://bootdey.com/img/Content/avatar/avatar1.png",
* "https://bootdey.com/img/Content/avatar/avatar2.png",
* "https://bootdey.com/img/Content/avatar/avatar7.png",
* "https://bootdey.com/img/Content/avatar/avatar3.png",
* ]
* },
* {
* id:1,
* image: "http://www.bizzcode.com/img/react.png",
* name:"Group React Native",
* countMembers:3,
* members:[
* "https://bootdey.com/img/Content/avatar/avatar6.png",
* "https://bootdey.com/img/Content/avatar/avatar1.png",
* "https://bootdey.com/img/Content/avatar/avatar2.png"
* ]
* },
* {
* id:6,
* image: "data:image/png;base64NK1oG2FkA88vWTS6cxrHv3maoUqUsgW12Ym3SS5CgifXtIrjTHem8pIsw7vcxMCU4Lg/Fc6EwBeYZip65DzgWyLf0khh7mN8yYw/a /4VK8prFB9OLmd8odGBz7l/dKXvvQlvvof8OfLzul/Od4AAAAASUVORK5CYII=",
* name:"Group C++",
* countMembers:12,
* members:[
* "https://bootdey.com/img/Content/avatar/avatar6.png",
* "https://bootdey.com/img/Content/avatar/avatar1.png",
* "https://bootdey.com/img/Content/avatar/avatar2.png"
* ]
* },
* {
* id:7,
* image: "https://miro.medium.com/fit/c/256/256/1\*FFHsUMGrt-Ece6KP6TRhHA.png",
* name:"Group Python",
* countMembers:234,
* members:[
* "https://bootdey.com/img/Content/avatar/avatar6.png",
* "https://bootdey.com/img/Content/avatar/avatar1.png",
* "https://bootdey.com/img/Content/avatar/avatar2.png"
* ]
* },
* ]
* }
* }
* renderGroupMembers = (group) => {
* if(group.members) {
* return (
* <View style={styles.groupMembersContent}>
* {group.members.map((prop, key) => {
* return (
* <Image key={key} style={styles.memberImage}  source={{uri:prop}}/>
* );
* })}
* </View>
* );
* }
* return null;
* }
* render() {
* return (
* <FlatList
* style={styles.root}
* data={this.state.data}
* extraData={this.state}
* ItemSeparatorComponent={() => {
* return (
* <View style={styles.separator}/>
* )
* }}
* keyExtractor={(item)=>{
* return item.id;
* }}
* renderItem={(item) => {
* const Group = item.item;
* let mainContentStyle;
* if(Group.attachment) {
* mainContentStyle = styles.mainContent;
* }
* return(
* <View style={styles.container}>
* <Image source={{uri:Group.image}} style={styles.avatar}/>
* <View style={styles.content}>
* <View style={mainContentStyle}>
* <View style={styles.text}>
* <Text style={styles.groupName}>{Group.name}</Text>
* </View>
* <Text style={styles.countMembers}>
* {Group.countMembers} members
* </Text>
* <Text style={styles.timeAgo}>
* Updated 2 months ago
* </Text>
* {this.renderGroupMembers(Group)}
* </View>
* </View>
* </View>
* );
* }}/>
* );
* }
* }
* Giao diện trên giả lập:



* Võ Xuân Hoàng
* Code:

constructor(props) {

    super(props);

    this.state = {

      msg: '',

      messages: [

        {id:1, sent: true,  msg: 'Lorem ipsum dolor',   image:'https://www.bootdey.com/img/Content/avatar/avatar1.png'},

        {id:2, sent: true,  msg: 'sit amet, consectetuer',   image:'https://www.bootdey.com/img/Content/avatar/avatar1.png'},

        {id:3, sent: false, msg: 'adipiscing elit. Aenean ', image:'https://www.bootdey.com/img/Content/avatar/avatar6.png'},

        {id:4, sent: true,  msg: 'commodo ligula eget dolor.',   image:'https://www.bootdey.com/img/Content/avatar/avatar1.png'},

        {id:5, sent: false, msg: 'Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes', image:'https://www.bootdey.com/img/Content/avatar/avatar6.png'},

        {id:6, sent: true,  msg: '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',   image:'https://www.bootdey.com/img/Content/avatar/avatar1.png'},

        {id:7, sent: false, msg: 'rhoncus ut, imperdiet', image:'https://www.bootdey.com/img/Content/avatar/avatar6.png'},

        {id:8, sent: false, msg: 'a, venenatis vitae', image:'https://www.bootdey.com/img/Content/avatar/avatar6.png'},

      ]

    };

    this.send = this.send.bind(this);

    this.reply = this.reply.bind(this);

    this.renderItem   = this.\_renderItem.bind(this);

  }

  reply() {

    var messages = this.state.messages;

    messages.push({

      id:Math.floor((Math.random() \* 99999999999999999) + 1),

      sent: false,

      msg: this.state.msg,

      image:'https://www.bootdey.com/img/Content/avatar/avatar6.png'

    });

    this.setState({msg:'', messages:messages});

  }

  send() {

    if (this.state.msg.length > 0) {

      var messages = this.state.messages;

      messages.push({

        id:Math.floor((Math.random() \* 99999999999999999) + 1),

        sent: true,

        msg: this.state.msg,

        image:'https://www.bootdey.com/img/Content/avatar/avatar1.png'

      });

      this.setState({messages:messages});

      setTimeout(() => {

        this.reply();

      }, 2000);

    }

  }

  \_renderItem = ({item}) => {

    if (item.sent === false) {

      return (

        <View style={styles.eachMsg}>

          <Image source={{ uri: item.image}} style={styles.userPic} />

          <View style={styles.msgBlock}>

            <Text style={styles.msgTxt}>{item.msg}</Text>

          </View>

        </View>

      );

    } else{

      return (

        <View style={styles.rightMsg} >

          <View style={styles.rightBlock} >

            <Text style={styles.rightTxt}>{item.msg}</Text>

          </View>

          <Image source={{uri: item.image}} style={styles.userPic} />

        </View>

      );

    }

  };

  render() {

    return (

      <View style={{ flex: 1 }}>

          <KeyboardAvoidingView behavior="padding" style={styles.keyboard}>

            <FlatList

              style={styles.list}

              extraData={this.state}

              data={this.state.messages}

              keyExtractor = {(item) => {

                return item.id;

              }}

              renderItem={this.renderItem}/>

            <View style={styles.input}>

              <TextInput

                style={{flex: 1 }}

                value={this.state.msg}

                placeholderTextColor = "#696969"

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

                blurOnSubmit={false}

                onSubmitEditing={() => this.send()}

                placeholder="Type a message"

                returnKeyType="send"/>

            </View>

          </KeyboardAvoidingView>

      </View>

    );

  }

* Giao diện trên giả lập:



* Võ Văn Chiến
* Code:

export default class Album extends Component {

  constructor(props) {

    super(props);

    this.state = {

      data: [

        { id: 0, title: "Product 1", count: 8, image: "https://lorempixel.com/400/200/nature/6/" },

        { id: 1, title: "Product 2", count: 8, image: "https://lorempixel.com/400/200/nature/5/" },

        { id: 2, title: "Product 3", count: 8, image: "https://lorempixel.com/400/200/nature/4/" },

        { id: 3, title: "Product 4", count: 8, image: "https://lorempixel.com/400/200/nature/6/" },

        { id: 4, title: "Product 5", count: 8, image: "https://lorempixel.com/400/200/sports/1/" },

        { id: 5, title: "Product 6", count: 8, image: "https://lorempixel.com/400/200/nature/8/" },

        { id: 6, title: "Product 7", count: 8, image: "https://lorempixel.com/400/200/nature/1/" },

        { id: 7, title: "Product 8", count: 3, image: "https://lorempixel.com/400/200/nature/3/" },

        { id: 8, title: "Product 9", count: 8, image: "https://lorempixel.com/400/200/nature/4/" },

        { id: 9, title: "Product 10", count: 8, image: "https://lorempixel.com/400/200/nature/5/" },

      ]

    };

  }

  next() {

    fetch('https://jsonplaceholder.typicode.com/photos', {

      method: 'POST',

      headers: {

        Accept: 'application/json',

        'Content-Type': 'application/json',

      },

    })

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

      .then((responseJson) => {

        this.setState({data: responseJson})

        if (responseJson == "true") {

          alert("Successfully");

        } else {

          alert("Failed");

        }

      })

      .catch((error) => {

        console.log(error);

      })

  }

  componentDidMount() {

   this.next();

  }

  addProductToCart = () => {

    Alert.alert('Success', 'The product has been added to your cart')

  }

  render() {

    return (

      <View style={styles.container}>

        <FlatList style={styles.list}

          contentContainerStyle={styles.listContainer}

          data={this.state.data}

          horizontal={false}

          numColumns={2}

          keyExtractor={(item) => {

            return item.id;

          }}

          ItemSeparatorComponent={() => {

            return (

              <View style={styles.separator} />

            )

          }}

          renderItem={(post) => {

            const item = post.item;

            return (

              <View style={styles.card}>

                <View style={styles.imageContainer}>

                  <TouchableHighlight onPress={() => this.props.navigation.navigate('Galleries')}>

                    <Image style={styles.cardImage} source={{ uri: item.url }} />

                  </TouchableHighlight>

                </View>

                <View style={styles.cardContent}>

                  <Text style={styles.title}>{item.title}</Text>

                  <Text style={styles.count}>({item.albumID})</Text>

                </View>

              </View>

            )

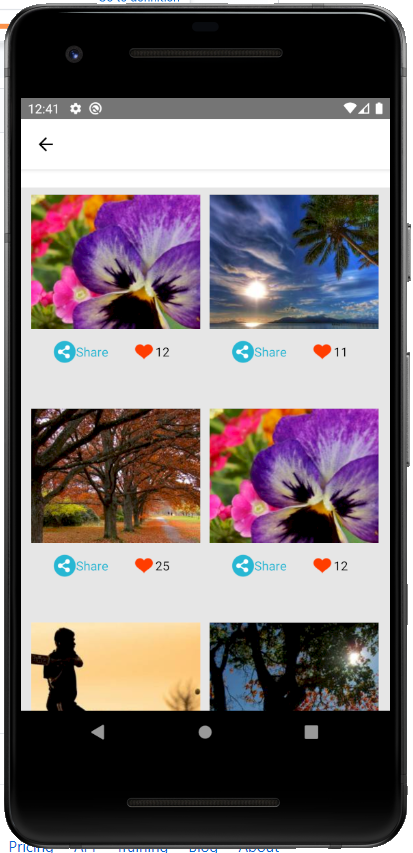
          }} />

      </View>

    );

  }

* **Chạy ứng dụng:**

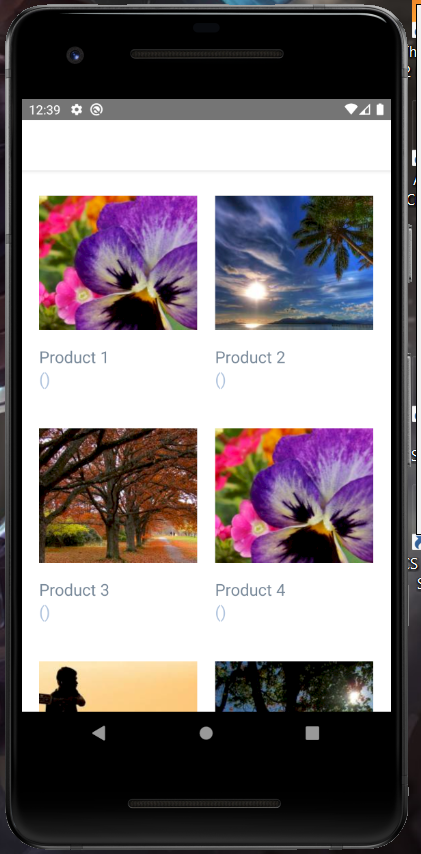
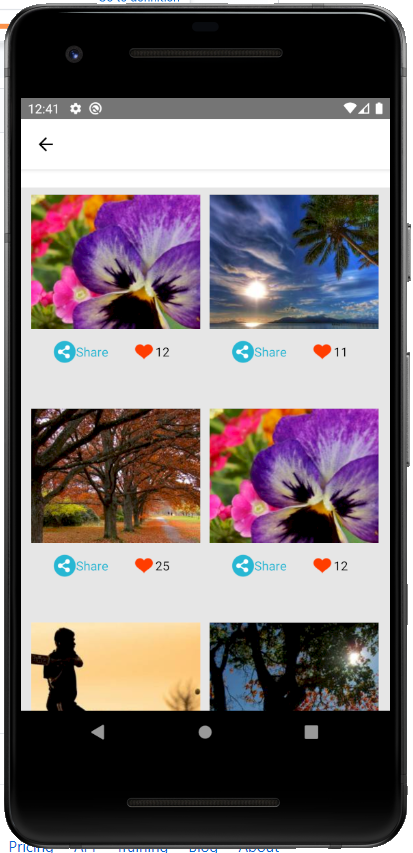


**Bài tập Exe4 của các thành viên**

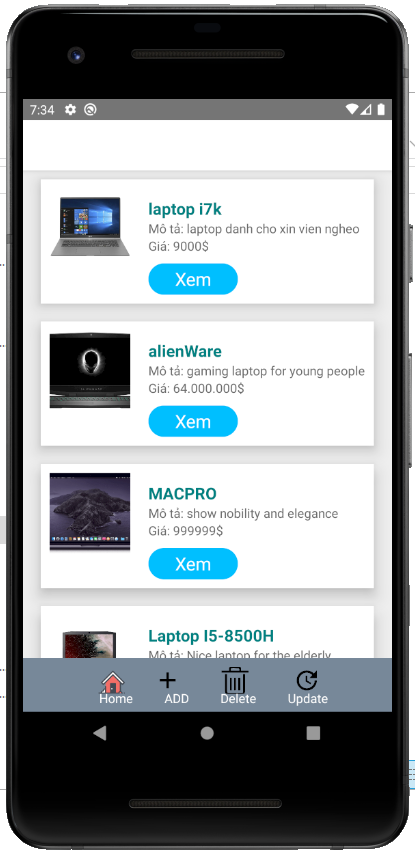
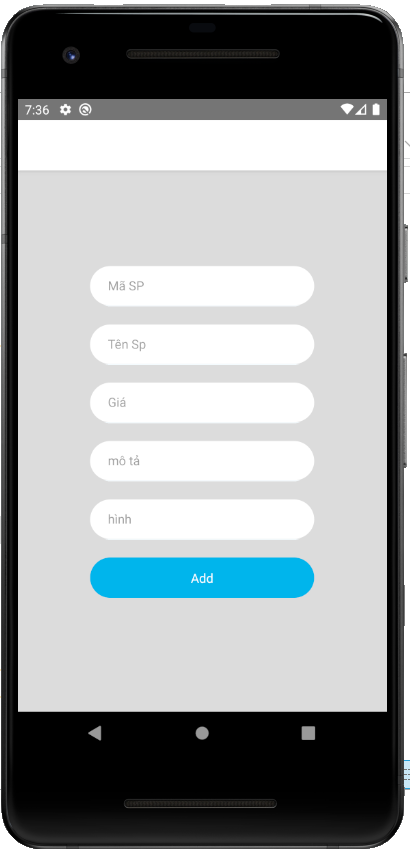
* Nguyễn Hữu Phú
* Code:
* export default class SanPham extends Component{
* constructor(props){
* super(props);
* this.state={
* }
* }
* render() {
* return (
* <View style={styles.product}>
* <View style={styles.containerImg}>
* <View>
* <Image style={styles.img} source={{uri: this.props.img}} />
* </View>
* </View>
* <View style={styles.containerContent}>
* <Text> Mã sản phẩm : {this.props.maSP} </Text>
* <Text> Tên sản phẩm : {this.props.tenSP} </Text>
* <Text> Mô tả : {this.props.des} </Text>
* <Text> Giá : {this.props.gia} </Text>
* </View>
* </View>
* );
* }
* }
* export default class Fetch extends React.Component {
* constructor(props){
* super(props);
* this.state ={ isLoading: true}
* }
* componentDidMount(){
* return fetch('http://192.168.1.4/php/Select\_Sanpham.php')
* .then((response) => response.json())
* .then((responseJson) => {
* this.setState({
* isLoading: false,
* dataSource: responseJson,
* }, function(){
* });
* })
* .catch((error) =>{
* console.error(error);
* });
* }
* render(){
* if(this.state.isLoading){
* return(
* <View style={{flex: 1, padding: 20}}>
* <ActivityIndicator/>
* </View>
* )
* }
* return(
* <View style={{flex: 1, paddingTop:20}}>
* <FlatList
* data={this.state.dataSource}
* renderItem={({item}) => <SanPham maSP={item.masp} tenSP={item.tensp} gia={item.gia} des={item.des} img={item.img}/>}
* />
* </View>
* );
* }
* }
* Giao diện



* Võ Văn Chiến
* Code:
* Album.js
* export default class Album extends Component {
* constructor(props) {
* super(props);
* this.state = {
* data: [
* { id: 0, title: "Product 1", count: 8, image: "https://lorempixel.com/400/200/nature/6/" },
* { id: 1, title: "Product 2", count: 8, image: "https://lorempixel.com/400/200/nature/5/" },
* { id: 2, title: "Product 3", count: 8, image: "https://lorempixel.com/400/200/nature/4/" },
* { id: 3, title: "Product 4", count: 8, image: "https://lorempixel.com/400/200/nature/6/" },
* { id: 4, title: "Product 5", count: 8, image: "https://lorempixel.com/400/200/sports/1/" },
* { id: 5, title: "Product 6", count: 8, image: "https://lorempixel.com/400/200/nature/8/" },
* { id: 6, title: "Product 7", count: 8, image: "https://lorempixel.com/400/200/nature/1/" },
* { id: 7, title: "Product 8", count: 3, image: "https://lorempixel.com/400/200/nature/3/" },
* { id: 8, title: "Product 9", count: 8, image: "https://lorempixel.com/400/200/nature/4/" },
* { id: 9, title: "Product 10", count: 8, image: "https://lorempixel.com/400/200/nature/5/" },
* ]
* };
* }
* next() {
* fetch('https://jsonplaceholder.typicode.com/photos', {
* method: 'POST',
* headers: {
* Accept: 'application/json',
* 'Content-Type': 'application/json',
* },
* })
* .then((response) => response.json())
* .then((responseJson) => {
* this.setState({data: responseJson})
* if (responseJson == "true") {
* alert("Successfully");
* } else {
* alert("Failed");
* }
* })
* .catch((error) => {
* console.log(error);
* })
* }
* componentDidMount() {
* this.next();
* }
* addProductToCart = () => {
* Alert.alert('Success', 'The product has been added to your cart')
* }
* render() {
* return (
* <View style={styles.container}>
* <FlatList style={styles.list}
* contentContainerStyle={styles.listContainer}
* data={this.state.data}
* horizontal={false}
* numColumns={2}
* keyExtractor={(item) => {
* return item.id;
* }}
* ItemSeparatorComponent={() => {
* return (
* <View style={styles.separator} />
* )
* }}
* renderItem={(post) => {
* const item = post.item;
* return (
* <View style={styles.card}>
* <View style={styles.imageContainer}>
* <TouchableHighlight onPress={() => this.props.navigation.navigate('Galleries')}>
* <Image style={styles.cardImage} source={{ uri: item.url }} />
* </TouchableHighlight>
* </View>
* <View style={styles.cardContent}>
* <Text style={styles.title}>{item.title}</Text>
* <Text style={styles.count}>({item.albumID})</Text>
* </View>
* </View>
* )
* }} />
* </View>
* );
* }
* }
* Galleries.js
* export default class Galleries extends Component {
* constructor(props) {
* super(props);
* this.state = {
* data: [
* {id:1, likes:12, image:"https://lorempixel.com/400/200/nature/6/"},
* {id:2, likes:11, image:"https://lorempixel.com/400/200/nature/5/"} ,
* {id:3, likes:25, image:"https://lorempixel.com/400/200/nature/4/"},
* {id:4, likes:12, image:"https://lorempixel.com/400/200/nature/6/"},
* {id:5, likes:10, image:"https://lorempixel.com/400/200/sports/1/"},
* {id:6, likes:12, image:"https://lorempixel.com/400/200/nature/8/"},
* {id:8, likes:45, image:"https://lorempixel.com/400/200/nature/3/"},
* ],
* };
* }
* addProductToCart = () => {
* Alert.alert('Success', 'The product has been added to your cart')
* }
* render() {
* return (
* <View style={styles.container}>
* <FlatList style={styles.list}
* contentContainerStyle={styles.listContainer}
* data={this.state.data}
* horizontal={false}
* numColumns={2}
* keyExtractor= {(item) => {
* return item.id;
* }}
* ItemSeparatorComponent={() => {
* return (
* <View style={styles.separator}/>
* )
* }}
* renderItem={(post) => {
* const item = post.item;
* return (
* <View style={styles.card}>
* <Image style={styles.cardImage} source={{uri:item.image}}/>
* <View style={styles.cardFooter}>
* <View style={styles.socialBarContainer}>
* <View style={styles.socialBarSection}>
* <TouchableOpacity style={styles.socialBarButton} onPress={() => this.addProductToCart()}>
* <Image style={styles.icon} source={{uri: 'https://png.icons8.com/flat\_round/50/000000/share.png'}}/>
* <Text style={[styles.socialBarLabel, styles.share]}>Share</Text>
* </TouchableOpacity>
* </View>
* <View style={styles.socialBarSection}>
* <TouchableOpacity style={styles.socialBarButton}>
* <Image style={styles.icon} source={{uri: 'https://png.icons8.com/color/50/000000/hearts.png'}}/>
* <Text style={styles.socialBarLabel}>{item.likes}</Text>
* </TouchableOpacity>
* </View>
* </View>
* </View>
* </View>
* )
* }}/>
* </View>
* );
* }
* }
* Giao diện

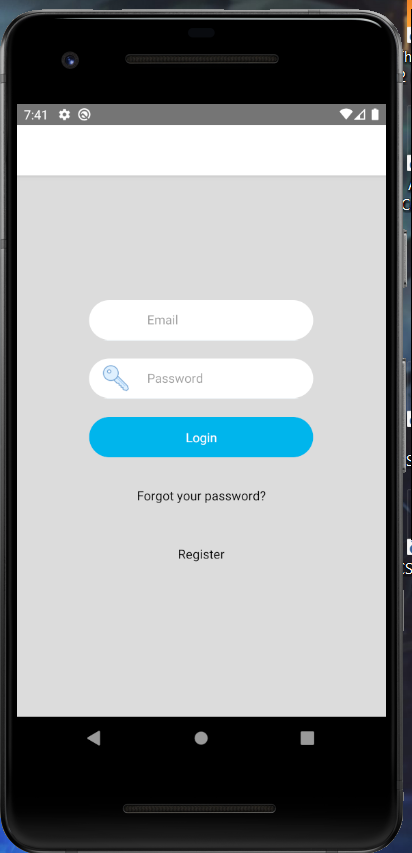
* Võ Xuân Hoàng
* Code:
* componentDidMount(){
* return fetch('http://172.168.83.73:81/Webservice/select-sanpham.php')
* .then((response) => response.json())
* .then((responseJson) => {
* this.setState({
* isLoading: false,
* dataSource: responseJson,
* }, function(){
* });
* })
* .catch((error) =>{
* console.error(error);
* });
* }
* setModalVisible(visible) {
* this.setState({modalVisible: visible});
* }
* render() {
* const {navigate} = this.props.navigation;
* return (
* <View style={styles.container}>
* {/\* Danh sach thong tin nguoi dung \*/}
* <FlatList
* style={styles.userList}
* columnWrapperStyle={styles.listContainer}
* data={this.state.dataSource}
* renderItem={({item}) => {
* return (
* // set su kien khi nhan vao
* <TouchableOpacity style={styles.card} onPress={() => {this.clickEventListener(item)}}>
* <Image style={styles.image} source={{uri: item.img}}/>
* <View style={styles.cardContent}>
* <Text style={styles.name}>{item.tensp}</Text>
* <Text style={styles.position}>Mô tả: {item.des}</Text>
* <Text style={styles.position}>Giá: {item.gia}</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.position}>{this.state.userSelected.position}</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={() => navigate('Users')}>
* <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={() => navigate('Insert')}>
* <View style={styles.iconContent}>
* <Image style={styles.icon}
* source={require('./img/ic\_add.png')}/>
* </View>
* <View >
* <Text style={styles.info}>ADD</Text>
* </View>
* </TouchableOpacity>
* <TouchableOpacity  onPress={() => navigate('Delete')}>
* <View style={styles.iconContent}>
* <Image style={styles.icon}
* source={require('./img/ic\_delete.png')}/>
* </View>
* <View >
* <Text style={styles.info}>Delete</Text>
* </View>
* </TouchableOpacity>
* <TouchableOpacity onPress={() => navigate('Update')}>
* <View style={styles.iconContent}>
* <Image style={styles.icon}
* source={require('./img/ic\_update.png')}/>
* </View>
* <View >
* <Text style={styles.info}>Update</Text>
* </View>
* </TouchableOpacity>
* </View>
* </View>
* </View>
* );
* }
* }
* Update\_Data\_Into\_MySQL = () =>
* {
* this.setState({ ActivityIndicator\_Loading : true }, () =>
* {
* fetch('http://10.0.2.2:80/webservice/update-sanpham.php',
* {
* method: 'POST',
* headers:
* {
* 'Accept': 'application/json',
* 'Content-Type': 'application/json',
* },
* body: JSON.stringify(
* {
* masp : this.state.masp,
* tensp : this.state.tensp,
* gia : this.state.gia,
* des : this.state.des,
* img : this.state.img
* })
* }).then((response) => response.json()).then((responseJsonFromServer) =>
* {
* alert(responseJsonFromServer);
* this.setState({ ActivityIndicator\_Loading : false });
* }).catch((error) =>
* {
* console.error(error);
* this.setState({ ActivityIndicator\_Loading : false});
* });
* });
* }
* render() {
* return (
* <View style={styles.container}>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Mã SP"
* keyboardType="default"
* underlineColorAndroid='transparent'
* onChangeText={(masp) => this.setState({masp})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Tên Sp"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(tensp) => this.setState({tensp})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Giá"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(gia) => this.setState({gia})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="mô tả"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(des) => this.setState({des})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="hình"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(img) => this.setState({img})}/>
* </View>
* <TouchableHighlight style={[styles.buttonContainer, styles.loginButton]} onPress={() => this.Update\_Data\_Into\_MySQL()}>
* <Text style={styles.loginText}>Add</Text>
* </TouchableHighlight>
* </View>
* );
* }
* }
* Giao diện:

# Chương 4: Xây Dựng Ứng Dụng

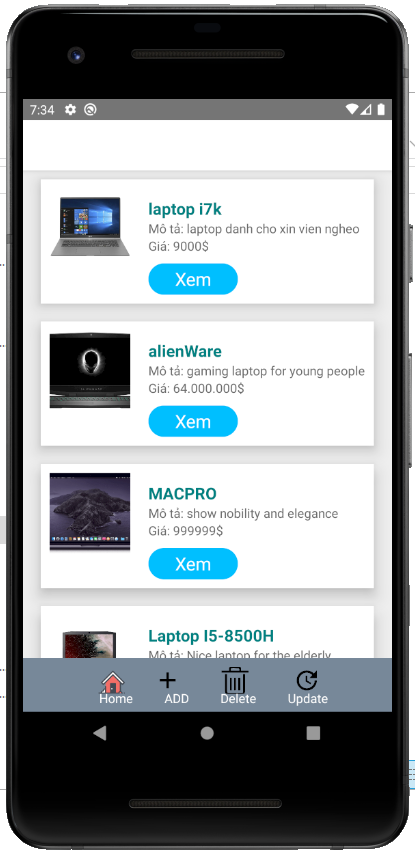
1. **Màn hình Login (Võ Văn Chiến)**

* **Code**
* login() {
* fetch('http://172.168.83.73:81/webservice/login.php', {
* method: 'POST',
* headers: {
* Accept: 'application/json',
* 'Content-Type': 'application/json',
* },
* body: JSON.stringify({
* username: this.props.username,
* password: this.props.password,
* }),
* })
* .then((response) => response.text())
* .then((responseJson) => {
* if (responseJson == "true") {
* alert("Successfully Login");
* } else {
* alert("Login Failed");
* }
* })
* .catch((error) => {
* alert(error);
* console.log(error);
* })
* }
* render() {
* return (
* <View style={styles.container}>
* <View style={styles.inputContainer}>
* <Image style={styles.inputIcon} source={{uri: 'https://png.icons8.com/message/ultraviolet/50/3498db'}}/>
* <TextInput style={styles.inputs}
* placeholder="Email"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(username) => this.setState({username})}/>
* </View>
* <View style={styles.inputContainer}>
* <Image style={styles.inputIcon} source={{uri: 'https://png.icons8.com/key-2/ultraviolet/50/3498db'}}/>
* <TextInput style={styles.inputs}
* placeholder="Password"
* secureTextEntry={true}
* underlineColorAndroid='transparent'
* onChangeText={(password) => this.setState({password})}/>
* </View>
* <TouchableHighlight style={[styles.buttonContainer, styles.loginButton]}
* onPress={() => this.login()}>
* <Text style={styles.loginText}>Login</Text>
* </TouchableHighlight>
* <TouchableHighlight style={styles.buttonContainer} onPress={() => this.onClickListener('restore\_password')}>
* <Text>Forgot your password?</Text>
* </TouchableHighlight>
* <TouchableHighlight style={styles.buttonContainer} onPress={() => this.onClickListener('register')}>
* <Text>Register</Text>
* </TouchableHighlight>
* </View>
* );
* }
* }
* **Giao diện:**



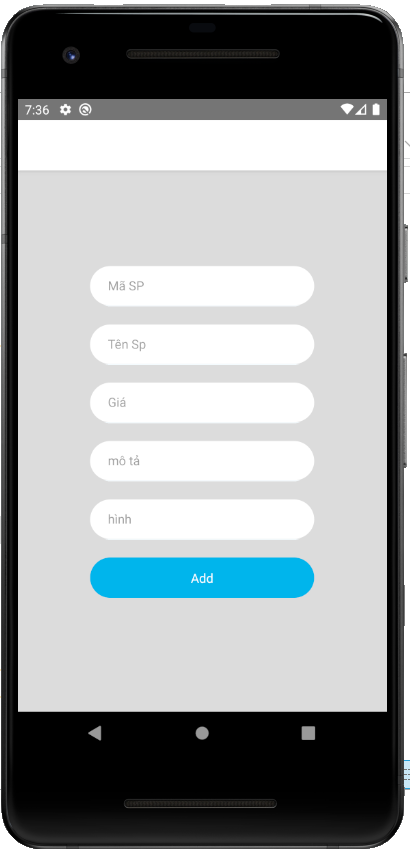
1. **Màn hình sản phẩm (Nguyễn Hữu Phú)**

* **Code:**
* componentDidMount(){
* return fetch('http://172.168.83.73:81/Webservice/select-sanpham.php')
* .then((response) => response.json())
* .then((responseJson) => {
* this.setState({
* isLoading: false,
* dataSource: responseJson,
* }, function(){
* });
* })
* .catch((error) =>{
* console.error(error);
* });
* }
* setModalVisible(visible) {
* this.setState({modalVisible: visible});
* }
* render() {
* const {navigate} = this.props.navigation;
* return (
* <View style={styles.container}>
* {/\* Danh sach thong tin nguoi dung \*/}
* <FlatList
* style={styles.userList}
* columnWrapperStyle={styles.listContainer}
* data={this.state.dataSource}
* renderItem={({item}) => {
* return (
* // set su kien khi nhan vao
* <TouchableOpacity style={styles.card} onPress={() => {this.clickEventListener(item)}}>
* <Image style={styles.image} source={{uri: item.img}}/>
* <View style={styles.cardContent}>
* <Text style={styles.name}>{item.tensp}</Text>
* <Text style={styles.position}>Mô tả: {item.des}</Text>
* <Text style={styles.position}>Giá: {item.gia}</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.position}>{this.state.userSelected.position}</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={() => navigate('Users')}>
* <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={() => navigate('Insert')}>
* <View style={styles.iconContent}>
* <Image style={styles.icon}
* source={require('./img/ic\_add.png')}/>
* </View>
* <View >
* <Text style={styles.info}>ADD</Text>
* </View>
* </TouchableOpacity>
* <TouchableOpacity  onPress={() => navigate('Delete')}>
* <View style={styles.iconContent}>
* <Image style={styles.icon}
* source={require('./img/ic\_delete.png')}/>
* </View>
* <View >
* <Text style={styles.info}>Delete</Text>
* </View>
* </TouchableOpacity>
* <TouchableOpacity onPress={() => navigate('Update')}>
* <View style={styles.iconContent}>
* <Image style={styles.icon}
* source={require('./img/ic\_update.png')}/>
* </View>
* <View >
* <Text style={styles.info}>Update</Text>
* </View>
* </TouchableOpacity>
* </View>
* </View>
* </View>
* );
* }
* }
* **Giao diện:**



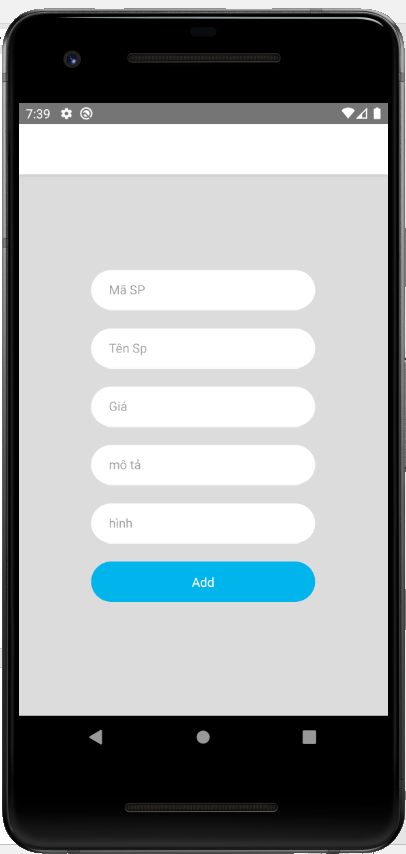
1. **Màn hình thêm sản phẩm (Võ Xuân Hoàng)**

* Insert\_Data\_Into\_MySQL = () =>
* {
* this.setState({ ActivityIndicator\_Loading : true }, () =>
* {
* fetch('http://172.168.83.73:81/webservice/insert-sanPham.php',
* {
* method: 'POST',
* headers:
* {
* 'Accept': 'application/json',
* 'Content-Type': 'application/json',
* },
* body: JSON.stringify(
* {
* masp : this.state.masp,
* tensp : this.state.tensp,
* gia : this.state.gia,
* des : this.state.des,
* img : this.state.img
* })
* }).then((response) => response.json()).then((responseJsonFromServer) =>
* {
* alert(responseJsonFromServer);
* this.setState({ ActivityIndicator\_Loading : false });
* }).catch((error) =>
* {
* console.error(error);
* this.setState({ ActivityIndicator\_Loading : false});
* });
* });
* }
* render() {
* return (
* <View style={styles.container}>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Mã SP"
* keyboardType="default"
* underlineColorAndroid='transparent'
* onChangeText={(masp) => this.setState({masp})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Tên Sp"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(tensp) => this.setState({tensp})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Giá"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(gia) => this.setState({gia})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="mô tả"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(des) => this.setState({des})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="hình"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(img) => this.setState({img})}/>
* </View>
* <TouchableHighlight style={[styles.buttonContainer, styles.loginButton]} onPress={() => this.Insert\_Data\_Into\_MySQL()}>
* <Text style={styles.loginText}>Add</Text>
* </TouchableHighlight>
* </View>
* );
* }
* }
* **Giao diện**



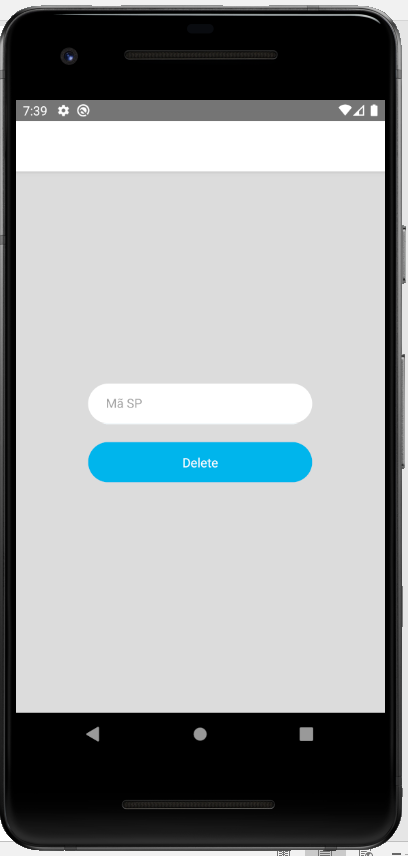
1. **Màn hình sửa sản phẩm (Võ Xuân Hoàng)**

* **Code:**
* Update\_Data\_Into\_MySQL = () =>
* {
* this.setState({ ActivityIndicator\_Loading : true }, () =>
* {
* fetch('http://10.0.2.2:80/webservice/update-sanpham.php',
* {
* method: 'POST',
* headers:
* {
* 'Accept': 'application/json',
* 'Content-Type': 'application/json',
* },
* body: JSON.stringify(
* {
* masp : this.state.masp,
* tensp : this.state.tensp,
* gia : this.state.gia,
* des : this.state.des,
* img : this.state.img
* })
* }).then((response) => response.json()).then((responseJsonFromServer) =>
* {
* alert(responseJsonFromServer);
* this.setState({ ActivityIndicator\_Loading : false });
* }).catch((error) =>
* {
* console.error(error);
* this.setState({ ActivityIndicator\_Loading : false});
* });
* });
* }
* render() {
* return (
* <View style={styles.container}>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Mã SP"
* keyboardType="default"
* underlineColorAndroid='transparent'
* onChangeText={(masp) => this.setState({masp})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Tên Sp"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(tensp) => this.setState({tensp})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Giá"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(gia) => this.setState({gia})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="mô tả"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(des) => this.setState({des})}/>
* </View>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="hình"
* keyboardType="email-address"
* underlineColorAndroid='transparent'
* onChangeText={(img) => this.setState({img})}/>
* </View>
* <TouchableHighlight style={[styles.buttonContainer, styles.loginButton]} onPress={() => this.Update\_Data\_Into\_MySQL()}>
* <Text style={styles.loginText}>Add</Text>
* </TouchableHighlight>
* </View>
* );
* }
* }
* Giao diện:



1. **Màn hình xóa sản phẩm (Võ Văn Chiến)**

* **Code:**
* Delete\_Data\_Into\_MySQL = () =>
* {
* this.setState({ ActivityIndicator\_Loading : true }, () =>
* {
* fetch('http://10.0.2.2:80/webservice/delete-sanpham.php',
* {
* method: 'POST',
* headers:
* {
* 'Accept': 'application/json',
* 'Content-Type': 'application/json',
* },
* body: JSON.stringify(
* {
* masp : this.state.masp,
* })
* }).then((response) => response.json()).then((responseJsonFromServer) =>
* {
* alert(responseJsonFromServer);
* this.setState({ ActivityIndicator\_Loading : false });
* }).catch((error) =>
* {
* console.error(error);
* this.setState({ ActivityIndicator\_Loading : false});
* });
* });
* }
* render() {
* return (
* <View style={styles.container}>
* <View style={styles.inputContainer}>
* <TextInput style={styles.inputs}
* placeholder="Mã SP"
* keyboardType="default"
* underlineColorAndroid='transparent'
* onChangeText={(masp) => this.setState({masp})}/>
* </View>
* <TouchableHighlight style={[styles.buttonContainer, styles.loginButton]} onPress={() => this.Delete\_Data\_Into\_MySQL()}>
* <Text style={styles.loginText}>Delete</Text>
* </TouchableHighlight>
* </View>
* );
* }
* }
* **Giao diện:**



# Chương 5: Kết Luận Và Kiến Nghị

- Trong thời gian học tập ở trường và đặt biệt hơn là môn lập trình di động 3, với sự giúp đỡ 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 đỡ nhóm chúng em trong môn học này.

**Tài liệu tham khảo**

<https://facebook.github.io/react-native/>

<https://viblo.asia/p/component-state-va-props-trong-react-native-Eb85oM34Z2G>

<https://viblo.asia/p/custom-components-trong-react-native-3Q75waB2ZWb>