|  |
| --- |
|  |
| |  |  | | --- | --- | |  |  | |  | |

**BÁO CÁO ĐỒ ÁN**

|  |
| --- |
| **Môn : Lập trình di động 3** |
| **Thành viên nhóm 16:**   * **Cao Sơn Hiệp** * **Phạm Văn Thịnh** * **Nguyễn Hùng Thắng**   **Tên đồ án : Ứng dụng xem thực đơn món ăn** |

**THÁNG 10/2019**

MỤC LỤC

[**Chương 1. Giới thiệu** 1](#_Toc22755875)

[**1.1** **Giới thiệu ứng dụng xem thực đơn món ăn:** 1](#_Toc22755876)

[**Chương 2. Tìm hiểu ES6** 2](#_Toc22755877)

[**2.1 Cao Sơn Hiệp** 2](#_Toc22755878)

[**2.2 Thịnh** 5](#_Toc22755879)

[**2.3 Thắng** 5](#_Toc22755880)

[**Chương 3 : Tìm hiểu Component** 6](#_Toc22755881)

[**3.1 Cao Sơn Hiệp** 6](#_Toc22755882)

[**3.2 Thịnh** 8](#_Toc22755883)

[**3.3 Thắng** 8](#_Toc22755884)

[**Chương 4 : Xây dựng ứng dụng** 9](#_Toc22755885)

[**4.1 Màn hình đăng nhập: cho phép người dùng đăng nhập và đăng ký (Nguyễn Hùng Thắng).** 9](#_Toc22755886)

[**4.2 Màn hình Trang chủ và Tìm Kiếm: Hiển thị danh sách món ăn và tìm kiếm (Phạm Văn Thịnh).** 9](#_Toc22755887)

[**4.3 Màn hình Chi tiết món ăn : Hiển thị thông tin món ăn (Cao Sơn Hiệp).** 10](#_Toc22755888)

[**Chương 5 : Kết Luận** 11](#_Toc22755889)

[**CÁC TÀI LIỆU THAM KHẢO** 11](#_Toc22755890)

# **Chương 1. Giới thiệu**

## **Giới thiệu ứng dụng xem thực đơn món ăn:**

* Là ứng dụng nấu ăn miêu tả từng bước rõ ràng công thức nấu ăn và có giao diện đơn giản dễ sử dụng. Ứng dụng này hội tụ rất nhiều công thức nấu ăn với những hướng dẫn chi tiết bằng hình ảnh sinh động giúp bạn dễ dàng nắm bắt được cách nấu các món ăn một cách nhanh chóng.
* Màn hình chính của ứng dụng:



# **Chương 2. Tìm hiểu ES6**

## **2.1 Cao Sơn Hiệp**

- Giới thiệu: tìm hiểu class trong es6

- Ví dụ: Xuất thông tin sinh viên: mã sinh viên, tên sinh viên, điểm từng môn, tính điểm trung bình cộng, xếp loại sinh viên.

/\*  Tên: Cao Sơn Hiệp

    Masv: 15211TT0925

    Bài tập: Xuất thông tin sinh viên trong es6

\*/

// Lớp cha 'Nguoi' có câu lệnh constructor để khởi tạo đối tượng

class Nguoi {

    constructor(ten) {

      this.\_ten = ten;

    }

    get ten() {

      return this.\_ten;

    }

    set ten(newTen) {

      this.\_ten = newTen;

    }

    xuatten() {

        console.log(this.\_ten + '.');

    }

}

// Truyền tham số vào lớp 'Nguoi'

let nguoi = new Nguoi('Cao Sơn Hiệp');

//nguoi.xuatten();

// Tạo một 'sinhvien' kế thừa lớp 'Nguoi'

class sinhvien extends Nguoi {

    constructor(ten, masv, diem1, diem2, diem3) {

      super(ten);

      this.\_masv = masv;

      this.\_diem1 = diem1;

      this.\_diem2 = diem2;

      this.\_diem3 = diem3;

    }

    get masv() {

      return this.\_masv;

    }

    set masv(newmasv) {

      this.\_masv = newmasv;

    }

    get diem1() {

        return this.\_diem1;

      }

    set diem1(newdiem1) {

        this.\_diem1 = newdiem1;

    }

    get diem2() {

        return this.\_diem2;

      }

    set diem2(newdiem2) {

        this.\_diem2 = newdiem2;

    }

    get diem3() {

        return this.\_diem3;

      }

    set diem3(newdiem3) {

        this.\_diem1 = newdiem3;

    }

    thongtin() {

      console.log(this.\_ten + ' -- ' + this.\_masv + ' -- ' + this.\_diem1

      + ' -- ' + this.\_diem2 + ' -- ' + this.\_diem3);

    }

    diemtb(){

        console.log(`${(this.diem1 + this.diem2 + this.diem3)/3}`);

    }

    maxdiem(){

      let numbers = [this.diem1, this.diem2, this.diem3]

      console.log(Math.max(...numbers))

    }

    mindiem(){

      let numbers = [this.diem1, this.diem2, this.diem3]

      console.log(Math.min(...numbers))

    }

    xeploai() {

      if ((this.diem1 + this.diem2 + this.diem3)/3 >= 5) {

        console.log('Đạt');

        console.log('--> Sinh viên ' + nguoi.ten + ' đủ điều kiện lên lớp.');

      } else {

        console.log('Không Đạt');

        console.log('--> Sinh viên ' + nguoi.ten + ' không đủ điều kiện lên lớp.');

      }

    }

}

// Truyền tham số vào lớp 'sinhvien'

let sv = new sinhvien('Cao Sơn Hiệp', 'sv01', 7, 10, 3);

sv.thongtin();

console.log('-----Xuất thông tin-----');

console.log('Mã :' + sv.masv);

console.log('Tên :' + nguoi.ten);

console.log('Điểm Toán : ' + sv.diem1);

console.log('Điểm Lý : ' + sv.diem2);

console.log('Điểm Hóa : ' + sv.diem3);

console.log('Điểm trung bình cộng : ') + sv.diemtb();

console.log('Điểm cao nhất : ') + sv.maxdiem();

console.log('Điểm thấp nhất : ') + sv.mindiem();

console.log('Xếp loại : ') + sv.xeploai();

Kết quả:



## **2.2 Phạm Văn Thịnh**

Tạo class Xe và Xe Máy, khởi tạo đối tượng XeMay và tính giá bán trong ES6

//Phạm Văn Thịnh

//15211TT2290

//ES6

//Class

class Xe{

    constructor(nhaSanXuat, namSanXuat){

        this.nhaSanXuat = nhaSanXuat

        this.namSanXuat = namSanXuat

    }

    xuat(){

        return this.nhaSanXuat + " " + this.namSanXuat + " Loại xe: " + this.loaiXeMay + " Chi phí: " + this.chiPhi + " Giá: " + this.tinhGiaBan();

    }

}

class XeMay extends Xe{

    constructor(nhaSanXuat, namSanXuat, loaixe, chiphi){

        super(nhaSanXuat,namSanXuat);

        this.loaiXeMay = loaixe

        this.chiPhi = chiphi

    }

    tinhGiaBan(){

        return this.chiPhi\*2;

    }

}

//Object

xemay1 = new XeMay("Honda",2000, "Yamaha",100);

xemay2 = new XeMay("Tokota",2000, "Yoake",150);

console.log(xemay1.xuat());

console.log(xemay2.xuat());

Run:



# **Chương 3 : Tìm hiểu Component**

## **3.1 Cao Sơn Hiệp**

- Giới thiệu: tìm hiểu thư viện “native-base”, “react-native-vector-icons/FontAwesome” trong Component

- ví dụ: màn hình xem chi tiết sản phẩm

import React, { Component } from 'react';

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

import {

Container, Header, Content, Card, CardItem, Text,

Body, Left, Right, Button, Thumbnail, DeckSwiper

} from "native-base";

import Icon from 'react-native-vector-icons/FontAwesome';

import { Carousel } from '@ant-design/react-native';

export default class ChiTietSP extends Component {

render() {

return (

<View>

<Text style={{ marginTop: 2 }}>Chi tiết</Text>

<Container>

<Content>

<Card>

<CardItem style={styles.container}>

<Left>

<Thumbnail source={{

uri:

'http://mevaobep.com/wp-content/uploads/2015/11/mon-long-ga-xao-bong-thien-ly-bo-duong-cho-ca-nha-01-575x335.jpg'

}} />

<Body>

<Text>Mề gà xào hoa thiên lý</Text>

<Text note>Tác giả: ...</Text>

</Body>

</Left>

</CardItem>

<CardItem cardBody style={styles.container}>

<Image source={{

uri:

'http://mevaobep.com/wp-content/uploads/2015/11/mon-long-ga-xao-bong-thien-ly-bo-duong-cho-ca-nha-01-575x335.jpg'

}} style={{ height: 100, width: null, flex: 1 }} />

</CardItem>

<CardItem style={styles.container}>

<Left>

<Button transparent>

<Icon active name="thumbs-up" />

<Text>1 Likes</Text>

</Button>

</Left>

<Right>

<Button transparent>

<Icon active name="comment" />

<Text>4 Text</Text>

</Button>

</Right>

</CardItem>

<View

style={{

paddingLeft: 20,

paddingRight: 20,

backgroundColor: 'lightgray',

}}>

<Text>-Mề gà giòn sần sật với hoa thiên lý mềm cà ngon là món ăn rất bổ dưỡng.</Text>

<Text>-Nguyên liệu: </Text>

<Text>+ Hoa thiên lý: 200gr </Text>

<Text>+ Mề và tim gà: 300gr </Text>

<Text>+ Tỏi: 3-4 tép </Text>

<Text>+ Hành khô: 1 củ nhỏ </Text>

<Text>+ Hành lá, hạt tiêu, dầu hào, gia vị, dầu ăn, hạt nêm, mắm. </Text>

<Text> </Text>

<Text> </Text>

<Text> </Text>

<Text> </Text>

<Text> </Text>

<Text> </Text>

</View>

</Card>

</Content>

</Container>

</View>

)

}

}

const styles = StyleSheet.create({

container: {

backgroundColor: 'lightgray',

padding: 10

},

});

Kết quả:



## **3.2 Phạm Văn Thịnh**

- ví dụ: màn hình danh sách sản phẩm

Component: FlatList, ActivityIndicator, Text, View

StyleSheet, View, Image, Text, TouchableOpacity, ToastAndroid

import React, { Component } from "react";

import { StyleSheet, View, Image, Text, TouchableOpacity, ToastAndroid } from "react-native";

export default class SanPham extends Component {

  constructor(props) {

    super(props)

  }

  render() {

    return (

      <View style={styles.container}>

        <View

          style={[

            styles.stack,

            {

              marginTop: 10.81,

              marginLeft: 0.86,

              height: 89.99,

              width: 360.86

            }

          ]}

        >

          <TouchableOpacity style={styles.items} onPress={()=> ToastAndroid.show(this.props.ten, ToastAndroid.SHORT)}>

            <View style={styles.items}>

              <Text style={styles.tieude}>{this.props.ten}</Text>

              <Text style={styles.gioithieu}>{this.props.mota}</Text>

              <Text style={styles.gia}>{this.props.gia}</Text>

            </View>

            <Image

              source={{ uri: this.props.hinh }}

              resizeMode="contain"

              style={styles.image3}

            />

          </TouchableOpacity>

        </View>

      </View>

    );

  }

}

const styles = StyleSheet.create({

  container: {

    flex: 1

  },

  materialHeader2: {

    width: 375,

    height: 56,

    marginLeft: -2

  },

  image2: {

    width: 273,

    height: 161,

    marginTop: 13,

    marginLeft: 32

  },

  text: {

    color: "rgba(155,155,155,1)",

    fontSize: 20,

    marginTop: 24.08,

    marginLeft: 32

  },

  stack: {

    position: "relative"

  },

  items: {

    top: 0.19,

    left: 1.14,

    width: 360,

    height: 90,

    backgroundColor: "rgba(255,255,255,1)",

    position: "absolute",

    borderColor: "rgba(255,255,255,1)",

    borderWidth: 1,

    borderRadius: 5

  },

  tieude: {

    color: "rgba(197,81,81,1)",

    fontSize: 16,

    marginTop: 5,

    marginLeft: 125.28

  },

  gioithieu: {

    color: "rgba(197,81,81,1)",

    fontSize: 16,

    marginTop: 5,

    marginLeft: 125.28

  },

  gia: {

    color: "rgba(197,81,81,1)",

    fontSize: 16,

    marginTop: 5,

    marginLeft: 125.28

  },

  image3: {

    top: 0.19,

    left: 0.14,

    width: 101,

    height: 90,

    position: "absolute"

  },

  materialIconTextButtonsFooter: {

    width: 375,

    height: 56,

    marginTop: 123.4,

    marginLeft: -9

  }

});

Màn hình fetch dữ liệu :

import React from 'react';

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

import SanPham from './SanPham';

export default class FetchExample extends React.Component {

//http://www.json-generator.com/api/json/get/cqyWkibRwy?indent=2

//http://www.json-generator.com/api/json/get/cfVyXkLmeW?indent=2

  constructor(props){

    super(props);

    this.state ={ isLoading: true}

  }

  componentDidMount(){

    return fetch('http://www.json-generator.com/api/json/get/cfVyXkLmeW?indent=2')

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

      .then((responseJson) => {

        this.setState({

          isLoading: false,

          dataSource: responseJson.drinks,

        }, 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 ten={item.ten} mota={item.mota} gia={item.gia} hinh={item.hinh}/>}

          keyExtractor={(item, index) => index}

        />

      </View>

    );

  }

}

## **3.3 Nguyễn Hùng Thắng**

- ví dụ: màn hình đăng nhập

React.Component

Để sử dụng phải cần extends React.Component.

Nó khác ở React.PureComponent một chổ duy nhất là shouldComponentUpdate.

/\* eslint-disable no-alert \*/

import React, {Component} from 'react';

import {

StyleSheet,

Text,

View,

TextInput,

TouchableOpacity,

TouchableWithoutFeedback,

Keyboard,

Image,

} from 'react-native';

export default class Login extends Component {

constructor(props) {

super(props);

this.state = {

UserName: '',

UserEmail: '',

UserPassword: '',

};

}

render() {

return (

<TouchableWithoutFeedback onPress={Keyboard.dismiss}>

<View style={styles.container}>

<View style={styles.up}>

<Image

source={require('../Contents/Image/Logo.png')}

// eslint-disable-next-line react-native/no-inline-styles

style={{height: 200, width: 250, resizeMode: 'stretch'}}

/>

<Text style={styles.title}>Quản Lý Nhà Hàng</Text>

</View>

<View style={styles.down}>

<View style={styles.textInputContainer}>

<TextInput

style={styles.textInput}

textContentType="emailAddress"

keyboardType="email-address"

placeholder="Tên Tài Khoản "

value={this.state.UserEmail}

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

/>

</View>

<View style={styles.textInputContainer}>

<TextInput

style={styles.textInput}

placeholder="Nhập Mật Khẩu"

secureTextEntry={true}

value={this.state.UserPassword}

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

/>

</View>

<TouchableOpacity

style={styles.loginButton}

onPress={this.UserLoginFunction}>

<Text style={styles.loginButtonTitle}>LOGIN</Text>

</TouchableOpacity>

</View>

</View>

</TouchableWithoutFeedback>

);

}

}

// eslint-disable-next-line no-undef

UserLoginFunction = () => {

const {UserEmail} = this.state;

const {UserPassword} = this.state;

fetch('http://localhost:88/quanlyquanan/database/Login.php', {

method: 'POST',

headers: {

Accept: 'application/json',

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

},

body: JSON.stringify({

email: UserEmail,

password: UserPassword,

}),

})

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

.then(responseJson => {

// If server response message same as Data Matched

if (responseJson === 'yes') {

//Then open Profile activity and send user email to profile activity.

alert('Successfully Login');

} else {

alert(responseJson);

}

})

.catch(error => {

console.error(error);

});

};

const styles = StyleSheet.create({

container: {

flex: 1,

flexDirection: 'column',

justifyContent: 'center',

alignItems: 'stretch',

backgroundColor: '#a3bded',

},

up: {

flex: 5,

flexDirection: 'column',

justifyContent: 'center',

alignItems: 'center',

},

down: {

flex: 5, //50% of column

flexDirection: 'column',

justifyContent: 'flex-start',

alignItems: 'center',

},

title: {

color: 'rgb(240,48,7)',

textAlign: 'center',

width: 400,

fontSize: 30,

},

textInputContainer: {

paddingHorizontal: 10,

borderRadius: 6,

marginBottom: 20,

backgroundColor: 'rgb(255, 255, 255)',

},

textInput: {

width: 280,

height: 45,

},

loginButton: {

width: 300,

height: 45,

borderRadius: 6,

justifyContent: 'center',

alignItems: 'center',

backgroundColor: 'rgb(221, 97, 97)',

},

loginButtonTitle: {

fontSize: 18,

color: 'white',

},

facebookButton: {

width: 300,

height: 45,

borderRadius: 6,

justifyContent: 'center',

},

line: {

height: 1,

flex: 2,

backgroundColor: 'black',

},

textOR: {

flex: 1,

textAlign: 'center',

},

divider: {

flexDirection: 'row',

height: 40,

width: 298,

justifyContent: 'center',

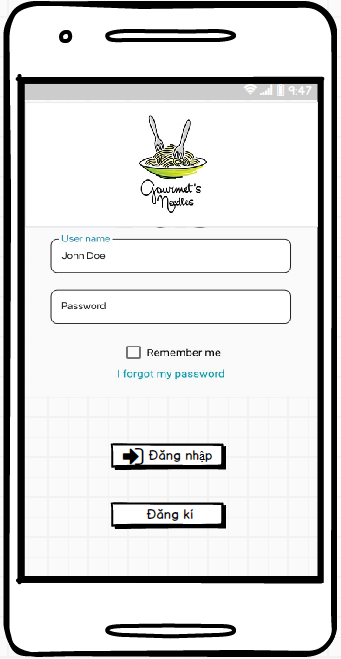
alignItems: 'center',

},

});

# **Chương 4 : Xây dựng ứng dụng**

## **4.1 Màn hình đăng nhập: cho phép người dùng đăng nhập và đăng ký (Nguyễn Hùng Thắng).**



Chức năng:

|  |  |  |
| --- | --- | --- |
| Tên chức năng | Chức năng | Mô tả |
| Logo | None | Hiển thị logo ứng dụng |
| Text User name | Click | Nhập tên đăng nhập |
| Text Pass | Click | Nhập mật khẩu |
| Radiobutton | Click | Chọn ghi nhớ tài khoản |
| Button đăng nhập | Click | Đăng nhập tài khoản |
| Button Đăng kí | Click | Chuyển sang màn hình đăng ký |

Login.js

/\* eslint-disable no-alert \*/

import React, {Component} from 'react';

import {

StyleSheet,

Text,

View,

TextInput,

TouchableOpacity,

TouchableWithoutFeedback,

Keyboard,

Image,

} from 'react-native';

export default class Login extends Component {

constructor(props) {

super(props);

this.state = {

UserName: '',

UserEmail: '',

UserPassword: '',

};

}

render() {

return (

<TouchableWithoutFeedback onPress={Keyboard.dismiss}>

<View style={styles.container}>

<View style={styles.up}>

<Image

source={require('../Contents/Image/Logo.png')}

// eslint-disable-next-line react-native/no-inline-styles

style={{height: 200, width: 250, resizeMode: 'stretch'}}

/>

<Text style={styles.title}>Quản Lý Nhà Hàng</Text>

</View>

<View style={styles.down}>

<View style={styles.textInputContainer}>

<TextInput

style={styles.textInput}

textContentType="emailAddress"

keyboardType="email-address"

placeholder="Tên Tài Khoản "

value={this.state.UserEmail}

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

/>

</View>

<View style={styles.textInputContainer}>

<TextInput

style={styles.textInput}

placeholder="Nhập Mật Khẩu"

secureTextEntry={true}

value={this.state.UserPassword}

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

/>

</View>

<TouchableOpacity

style={styles.loginButton}

onPress={this.UserLoginFunction}>

<Text style={styles.loginButtonTitle}>LOGIN</Text>

</TouchableOpacity>

</View>

</View>

</TouchableWithoutFeedback>

);

}

}

// eslint-disable-next-line no-undef

UserLoginFunction = () => {

const {UserEmail} = this.state;

const {UserPassword} = this.state;

fetch('http://localhost:88/quanlyquanan/database/Login.php', {

method: 'POST',

headers: {

Accept: 'application/json',

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

},

body: JSON.stringify({

email: UserEmail,

password: UserPassword,

}),

})

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

.then(responseJson => {

// If server response message same as Data Matched

if (responseJson === 'yes') {

//Then open Profile activity and send user email to profile activity.

alert('Successfully Login');

} else {

alert(responseJson);

}

})

.catch(error => {

console.error(error);

});

};

const styles = StyleSheet.create({

container: {

flex: 1,

flexDirection: 'column',

justifyContent: 'center',

alignItems: 'stretch',

backgroundColor: '#a3bded',

},

up: {

flex: 5,

flexDirection: 'column',

justifyContent: 'center',

alignItems: 'center',

},

down: {

flex: 5, //50% of column

flexDirection: 'column',

justifyContent: 'flex-start',

alignItems: 'center',

},

title: {

color: 'rgb(240,48,7)',

textAlign: 'center',

width: 400,

fontSize: 30,

},

textInputContainer: {

paddingHorizontal: 10,

borderRadius: 6,

marginBottom: 20,

backgroundColor: 'rgb(255, 255, 255)',

},

textInput: {

width: 280,

height: 45,

},

loginButton: {

width: 300,

height: 45,

borderRadius: 6,

justifyContent: 'center',

alignItems: 'center',

backgroundColor: 'rgb(221, 97, 97)',

},

loginButtonTitle: {

fontSize: 18,

color: 'white',

},

facebookButton: {

width: 300,

height: 45,

borderRadius: 6,

justifyContent: 'center',

},

line: {

height: 1,

flex: 2,

backgroundColor: 'black',

},

textOR: {

flex: 1,

textAlign: 'center',

},

divider: {

flexDirection: 'row',

height: 40,

width: 298,

justifyContent: 'center',

alignItems: 'center',

},

});

## **4.2 Màn hình Trang chủ và Tìm Kiếm: Hiển thị danh sách món ăn và tìm kiếm (Phạm Văn Thịnh).**



Chức năng:

|  |  |  |
| --- | --- | --- |
| Tên chức năng | Chức năng | Mô tả |
| Seach bar | Click | Nhập thông tin tìm kiếm |
| Button tìm kiếm | Click | Chuyển sang màn hình kết quả tìm kiếm |
| Images | None | Hiển thị hình ảnh món ăn |
| ListViewItem | Click | Chuyển sang màn hình chi tiêt món ăn |
| Text | None | Hiển thị tên món ăn, mô tả món ăn |
| Button Home | Click | Chuyển sang màn hình trang chủ |
| Button User | Click | Chuyển sang màn hình tài khoản |
| Button Thông báo | Click | Chuyển sang màn hình thông báo |

SanPham.js

import React, { Component } from "react";

import { StyleSheet, View, Image, Text, TouchableOpacity, ToastAndroid } from "react-native";

export default class SanPham extends Component {

  constructor(props) {

    super(props)

  }

  render() {

    return (

      <View style={styles.container}>

        <View

          style={[

            styles.stack,

            {

              marginTop: 10.81,

              marginLeft: 0.86,

              height: 89.99,

              width: 360.86

            }

          ]}

        >

          <TouchableOpacity style={styles.items} onPress={()=> ToastAndroid.show(this.props.ten, ToastAndroid.SHORT)}>

            <View style={styles.items}>

              <Text style={styles.tieude}>{this.props.ten}</Text>

              <Text style={styles.gioithieu}>{this.props.mota}</Text>

              <Text style={styles.gia}>{this.props.gia}</Text>

            </View>

            <Image

              source={{ uri: this.props.hinh }}

              resizeMode="contain"

              style={styles.image3}

            />

          </TouchableOpacity>

        </View>

      </View>

    );

  }

}

const styles = StyleSheet.create({

  container: {

    flex: 1

  },

  materialHeader2: {

    width: 375,

    height: 56,

    marginLeft: -2

  },

  image2: {

    width: 273,

    height: 161,

    marginTop: 13,

    marginLeft: 32

  },

  text: {

    color: "rgba(155,155,155,1)",

    fontSize: 20,

    marginTop: 24.08,

    marginLeft: 32

  },

  stack: {

    position: "relative"

  },

  items: {

    top: 0.19,

    left: 1.14,

    width: 360,

    height: 90,

    backgroundColor: "rgba(255,255,255,1)",

    position: "absolute",

    borderColor: "rgba(255,255,255,1)",

    borderWidth: 1,

    borderRadius: 5

  },

  tieude: {

    color: "rgba(197,81,81,1)",

    fontSize: 16,

    marginTop: 5,

    marginLeft: 125.28

  },

  gioithieu: {

    color: "rgba(197,81,81,1)",

    fontSize: 16,

    marginTop: 5,

    marginLeft: 125.28

  },

  gia: {

    color: "rgba(197,81,81,1)",

    fontSize: 16,

    marginTop: 5,

    marginLeft: 125.28

  },

  image3: {

    top: 0.19,

    left: 0.14,

    width: 101,

    height: 90,

    position: "absolute"

  },

  materialIconTextButtonsFooter: {

    width: 375,

    height: 56,

    marginTop: 123.4,

    marginLeft: -9

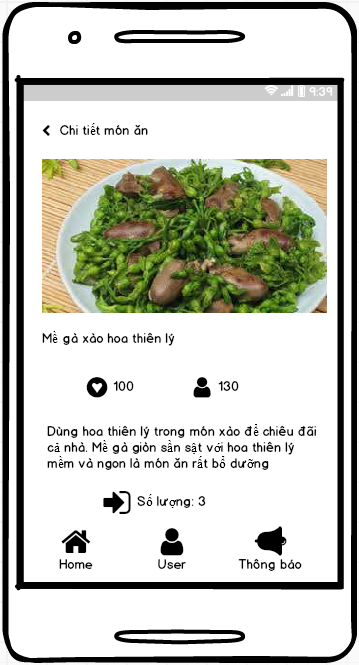
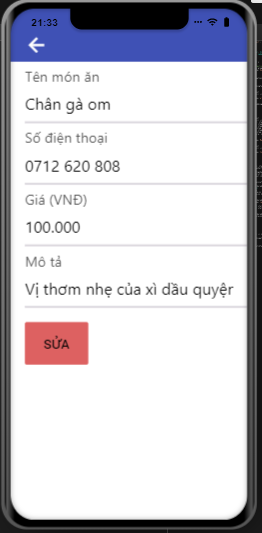
  }

});

## **4.3 Màn hình Chi tiết món ăn : Hiển thị thông tin thêm, xóa, sửa món ăn (Cao Sơn Hiệp).**

Chức năng:

|  |  |  |
| --- | --- | --- |
| Tên chức năng | Chức năng | Mô tả |
| Images | None | Hiển thị hình ảnh món ăn |
| Text | None | Hiển thị tên món ăn, mô tả món ăn |
| Button Like | Click | Đánh giá thích món ăn |
| Button View | Click | Lượt xem món ăn |
| Button Home | Click | Chuyển sang màn hình trang chủ |
| Button User | Click | Chuyển sang màn hình tài khoản |
| Button Thông báo | Click | Chuyển sang màn hình thông báo |



DetailsScreen.js

import React, { Component } from 'react';

import {

View, Alert

} from 'react-native'

import {

Container, Content, Button, Header, Icon, Title,

Body, Text, Left, Card, CardItem, Label, Right, Thumbnail

} from 'native-base';

import { connect } from 'react-redux'

class DetailsScreen extends Component {

deleteContact = () => {

//console.log(this.props.navigation.getParam('item').login.uuid)

this.props.deleteContact(this.props.navigation.getParam('item').login.uuid)

this.props.navigation.navigate('HomeScreen')

}

render() {

const item = this.props.navigation.getParam('item')

return (

<Container>

<Header style={{ paddingTop: 20 }}>

<Left>

<Button

transparent

onPress={() => this.props.navigation.navigate('HomeScreen')}

>

<Icon name='arrow-back' />

</Button>

</Left>

<Body style={{ paddingLeft: -20 }}>

<Title>

</Title>

</Body>

<Right>

<Button

transparent

onPress={() => this.props.navigation.navigate('addContactScreen', {

first: item.name.first,

last: item.name.last,

mota: item.mota,

gia: item.gia,

cell: item.cell,

key: item.login.uuid,

picture: item.picture.large

})}

>

<Icon type='MaterialIcons' name='edit' />

</Button>

<Button

transparent

onPress={() => this.deleteContact()}

>

<Icon type='Ionicons' name='md-trash' />

</Button>

</Right>

</Header>

<Content>

<Thumbnail

large

source={{ uri: item.picture.large }}

style={{ alignSelf: 'center' }}

/>

<Text style={{ alignSelf: 'center' }}>

{item.name.first.charAt(0).toUpperCase() + item.name.first.slice(1) + ' '}

{item.name.last.charAt(0).toUpperCase() + item.name.last.slice(1)}

</Text>

<Card

style={{ marginLeft: 5, marginRight: 5 }}

>

<CardItem bordered>

<Thumbnail

large

source={{ uri: item.picture.large }}

style={{ height: 150, width: 210, alignSelf: 'center' }}

/>

</CardItem>

<CardItem bordered>

<Icon name='call' style={{ fontSize: 20, color: 'grey', marginLeft: 0 }} />

<View>

<Text style={{ fontSize: 13 }}>{item.cell}</Text>

<Text style={{ fontSize: 13 }}>Gía: {item.gia} VNĐ</Text>

</View>

</CardItem>

<CardItem bordered>

<Text style={{ fontSize: 13, textAlign: 'center', }}>Mô tả: {item.mota}</Text>

</CardItem>

</Card>

</Content>

</Container>

);

}

}

const mapDispatchToProps = (dispatch) => {

return {

deleteContact: (product) => dispatch({ type: 'DELETE\_CONTACT', payload: product })

}

}

export default connect(null, mapDispatchToProps)(DetailsScreen)

addContactScreen.js

import React, { Component } from 'react';

import { Textarea, Toast, Container, Content, Form, Item, Input, Label, Button, Text, Root, Header, Left, Right, Icon, Title, Body } from 'native-base';

import { connect } from 'react-redux'

class addContactScreen extends Component {

state = {

bearbeiten: false,

first: this.props.navigation.getParam('first'),

last: this.props.navigation.getParam('last'),

cell: this.props.navigation.getParam('cell'),

gia: this.props.navigation.getParam('gia'),

mota: this.props.navigation.getParam('mota')

}

addContact = () => {

if (this.state.last != null && this.state.last != '') {

this.props.addContact({

"name": {

"first": this.state.first == null ? '' : this.state.first,

"last": this.state.last

},

"mota": this.state.mota,

"gia": this.state.gia,

"cell": this.state.cell,

"picture": {

"large": "https://giadinh.tv/wp-content/uploads/2016/12/7-ngay-giam-can-cap-toc-bang-trung-ga-luoc-4-e1482440277743.jpg"

},

"login": {

"uuid": this.props.state.contactKey

},

})

this.props.updateContactKey()

this.props.sortContacts()

this.props.navigation.navigate('HomeScreen')

} else {

Toast.show({

text: 'Tên không để trống!',

buttonText: 'Okay',

type: 'danger'

})

}

}

editContact = () => {

this.props.editContact({

"name": {

"first": this.state.first,

"last": this.state.last

},

"mota": this.state.mota,

"gia": this.state.gia,

"cell": this.state.cell,

"picture": {

"large": this.props.navigation.getParam('picture')

},

"login": {

"uuid": this.props.navigation.getParam('key')

},

})

this.props.sortContacts()

this.props.navigation.navigate('HomeScreen')

}

render() {

const first = this.props.navigation.getParam('first')

const last = this.props.navigation.getParam('last')

const cell = this.props.navigation.getParam('cell')

const gia = this.props.navigation.getParam('gia')

const mota = this.props.navigation.getParam('mota')

return (

<Root>

<Container>

<Content>

<Header style={{ paddingTop: 20 }}>

<Left>

<Button

transparent

onPress={() => this.props.navigation.navigate('HomeScreen')}

>

<Icon name='arrow-back' />

</Button>

</Left>

<Body>

<Title></Title>

</Body>

</Header>

<Form>

<Item stackedLabel>

<Label>Tên món ăn</Label>

<Input

defaultValue={last}

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

/>

</Item>

<Item stackedLabel>

<Label>Số điện thoại</Label>

<Input

defaultValue={cell}

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

/>

</Item>

<Item stackedLabel>

<Label>Giá (VNĐ)</Label>

<Input

defaultValue={gia}

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

/>

</Item>

<Item stackedLabel>

<Label>Mô tả</Label>

<Input

defaultValue={mota}

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

/>

</Item>

<Button

light

style={{ margin: 15, backgroundColor: 'rgb(221, 97, 97)' }}

onPress={() => last == null ? this.addContact() : this.editContact()}

>

<Text> {last == null ? 'THÊM' : 'SỬA'} </Text>

</Button>

</Form>

</Content>

</Container>

</Root>

);

}

}

const mapStateToProps = (state) => {

return {

state: state

}

}

const mapDispatchToProps = (dispatch) => {

return {

addContact: (product) => dispatch({ type: 'ADD\_CONTACT', payload: product }),

editContact: (product) => dispatch({ type: 'EDIT\_CONTACT', payload: product }),

updateContactKey: (product) => dispatch({ type: 'UPDATE\_CONTACT\_KEY', payload: product }),

sortContacts: (product) => dispatch({ type: 'SORT\_CONTACTS', payload: product }),

}

}

export default connect(mapStateToProps, mapDispatchToProps)(addContactScreen)

# **Chương 5 : Kết Luận**

Qua thời gian tìm hiểu React Native đã ứng dụng vào app quản lý danh sách các mặt hàng cần mua, thêm, sửa, xoá và lưu dữ liệu.Biết được khái niệm cơ bản trong React Native: components, props, state. Layout các Component với Flexbox, StyleSheet,...Các cú pháp trong javascript ES6: arrow functions, async, await, promise,...đã được lồng vào các ví dụ cụ thể.

# 

# **CÁC TÀI LIỆU THAM KHẢO**

* Tài liệu trên mạng :
* <https://quantrimang.com/bai-tap-javascript-co-giai-159572>
* <https://freetuts.net/hoc-javascript/bai-tap-javascript>
* <https://viblo.asia/p/bat-dau-voi-react-native-maGK7jQD5j2>