**TRƯỜNG CAO ĐẲNG CÔNG NGHỆ THỦ ĐỨC**

**KHOA CÔNG NGHỆ THÔNG TIN**

**🙞🕮🙜**

****

**TÀI LIỆU ĐẶC TẢ PHẦN MỀM**

**Lập trình Di Động 3**

**App Manga Reader Online**

**Giảng viên hướng dẫn: (Thầy) Trương Bá Thái**

**Nhóm thực hiện: Nhóm 3**

1. **Ngô Quốc Cường**
2. **Lê Quốc Việt**

***TP. Hồ Chí Minh, Ngày 19 tháng 9 năm 2019***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Họ và tên** | **Công việc được giao** | **Đóng góp** | **Chữ ký** |
| 1 | Ngô Quốc Cường | * Màn hình 1,2 |  |  |
| 2 | Lê Quốc Việt | * Màn hình 3 + Tìm kiếm |  |  |

**LỊCH SỬ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ngày** | **Phiên bản** | **Mô tả** | **Tác giả** | **Người sửa đổi** |
| **19/09/2019** | **1.0** | **Viết bản SRS** | **Ngô Quốc Cường** | **Ngô Quốc Cường** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**MỤC LỤC**

[CHƯƠNG 1. TỔNG QUAN,GIỚI THIỆU VỀ ỨNG DỤNG 1](#_Toc22203830)

[1.1. Tổng quan về ứng dụng 1](#_Toc22203831)

[1.2. Tẩm hoạt động của ứng dụng 1](#_Toc22203832)

[1.3. Feature/Component #1: Màn hình chính/ trang chủ: 1](#_Toc22203833)

[1.3.1. Giao diện người dùng 1](#_Toc22203834)

[1.3.2. Yêu cầu chức năng 1](#_Toc22203835)

[1.4. Feature/Component #2: Màn hình chi tiết từng truyện 1](#_Toc22203836)

[1.4.1. Giao diện người dùng 1](#_Toc22203837)

[1.4.2. Yêu cầu chức năng 1](#_Toc22203838)

[1.5. Feature/Component #3: Màn hình chương truyện 1](#_Toc22203839)

[1.5.1. Giao diện người dùng 1](#_Toc22203840)

[1.5.2. Yêu cầu chức năng 1](#_Toc22203841)

[CHƯƠNG 2. TÌM HIỂU VỀ ES6 1](#_Toc22203842)

[CHƯƠNG 3. CÁC COMPONENT 1](#_Toc22203843)

1. TỔNG QUAN,GIỚI THIỆU VỀ ỨNG DỤNG
   1. Tổng quan về ứng dụng

* App hỗ trợ việt đọc truyện tranh online của giới trẻ
  1. Tẩm hoạt động của ứng dụng

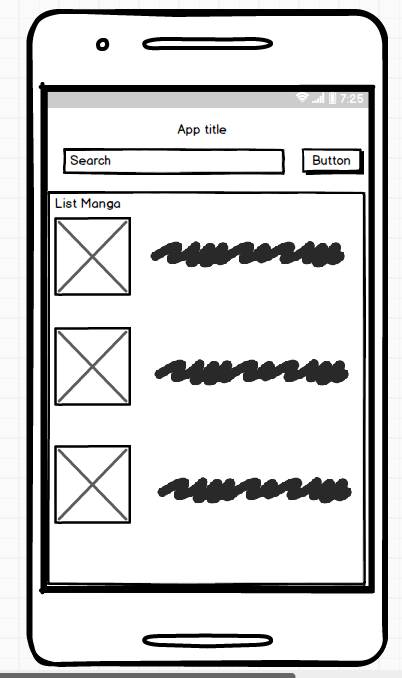
Android OS: Hỗ trợ các thiết bị có Android 26 , API 23 trở lên

* Android test devices: Oppo,Xiaomi

Android screen resolutions: 720x1520

Màn hình chỉ hỗ trợ chạy theo hàng dọc

* 1. Feature/Component #1: Màn hình chính/ trang chủ:
     1. Giao diện người dùng

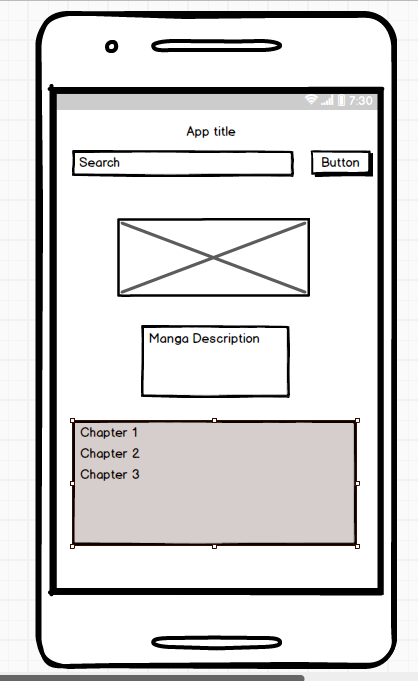


Hình 1 : Giao diện màn hình chính/trang chủ

* + 1. Yêu cầu chức năng

|  |  |  |  |
| --- | --- | --- | --- |
| **Đối tượng** | **Mô tả** | **Thao tác** | **Phản hồi** |
| **Màn hình gồm có** | * Text AppTitle * Button Search * TextInput Search * FlatList Truyen | N/A | N/A |
| **Button Search** | Tìm kiếm truyện | Nhấp vào button | Hiện ra danh sách truyện cần tìm |
| **TextInput** | Nhập truyện cần tìm kiếm | Nhấp vào TextInput | N/A |
| **FlatList Truyen** | Gồm các itiem hiển thị danh sách các truyện hiện có | Nhấp vào itiem muốn xem | Chuyển sang màn hình**“Chi tiết từng truyện”** |

* 1. Feature/Component #2: Màn hình chi tiết từng truyện
     1. Giao diện người dùng

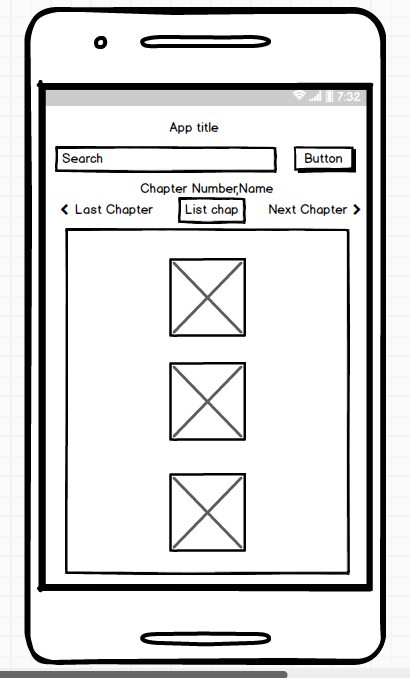


Hình 2 : Màn hình chi tiết từng truyện

* + 1. Yêu cầu chức năng

|  |  |  |  |
| --- | --- | --- | --- |
| **Đối tượng** | **Mô tả** | **Thao tác** | **Phản hồi** |
| Màn hình chọn nhân vật gồm có | * Text AppTitle * Button Search * TextInput Search * FlatList ChuongTruyen * Image Truyen * Text Mota | N/A | N/A |
| **Button Search** | Tìm kiếm truyện | Nhấp vào button | Hiện ra danh sách truyện cần tìm |
| **TextInput Search** | Nhập truyện cần tìm kiếm | Nhấp vào TextInput | N/A |
| **Image Truyen** | Hiển thị giao diện bìa của truyện | N/A | N/A |
| **Text Mota** | Mô tả sơ lược về truyện | N/A | N/A |
| **FlastList ChuongTruyen** | Chứa các itiem hiển thị danh sách các chương truyện hiện có của 1 truyện | Nhấp vào itiem muốn xem | Chuyển sang màn hình**“Chương truyện”** |

* 1. Feature/Component #3: Màn hình chương truyện
     1. Giao diện người dùng



Hình 3 : Giao diện màn hình hướng dẫn

* + 1. Yêu cầu chức năng

|  |  |  |  |
| --- | --- | --- | --- |
| **Đối tượng** | **Mô tả** | **Thao tác** | **Phản hồi** |
| **Màn hình hướng dẫn chơi gồm có** | Text App Title  TextInput Search  Button Search  Text Chap Title  Button Back  Button Next  ComboBox ChuongTruyen  FlatList AnhTruyen | N/A | N/A |
| **Button Search** | Tìm kiếm truyện | Nhấp vào button | Hiện ra danh sách truyện cần tìm |
| **TextInput Search** | Nhập truyện cần tìm kiếm | Nhấp vào TextInput | N/A |
| **Button Next** | Cho phép người dùng chuyển chương kế tiếp | Nhấp vào button | Chuyển sang chương tiếp theo |
| **Button Back** | Cho phép người dùng chuyển chương kế trước đó | Nhấp vào button | Chuyển sang chương trước đó |
| **ComboBox ChuongTruyen** | Cho phép người dùng chọn chương truyện tùy ý | Nhấp vào combo box rồi chọn chương tùy ý | Chuyển sang chương truyện đã chọn |
| **FlatList AnhTruyen** | Gồm các item hiển thị nội dung chương truyện | N/A | N/A |

CHƯƠNG 2. TÌM HIỂU VỀ ES6

Code demo ES6 của Lê Quốc Việt

var fib = [];

var x = 20;

fib[0] = 0;

fib[1] = 1;

for (let i = 2; i <= x; i++) {

fib[i] = fib[i - 2] + fib[i - 1];

}

for (i = 1; i < fib.length ; i++) {

console.log(fib[i]);

}

Code demo ES6 của Ngô Quốc Cường

var arraySanPham = [

{masp: 'sp001' , tensp : 'LapTop '},

{masp: 'sp002' , tensp : 'Tivi '},

{masp: 'sp003' , tensp : 'Tủ Lạnh'},

];

Console.log(arraySanPham.map(arraySanPham => arraySanPham.tensp));

CHƯƠNG 3. CÁC COMPONENT

Code demo của Lê Quốc Việt

*danhsachSV.js*

import React, { Component } from 'react';

import {FlatList, StyleSheet, Text, View, TextInput, Button, Alert } from 'react-native';

import SinhVien from './SinhVien';

var arraySinhVien = [];

// class user {

// constructor(name, phone){

// this.name = name;

// this.phone = phone;

// }

// }

// class user {

// constructor (name, phone)

// {

// this.name = name;

// this.phone = phone;

// }

// setName (name){

// this.name = name;

// }

// getName () {

// return this.name;

// }

// setPhone (phone) {

// this.phone = phone;

// }

// getPhone (){

// return this.phone;

// }

// };

export default class DemoFlatlist extends Component {

constructor(props) {

super(props);

this.state = {name: ""};

this.state = {phone: ""};

this.state = {reload: null};

}

reloadFlat = (keyname) =>{

this.setState((prevState) => {

return {

reload: keyname

};

});

}

varTest(text) {

let x = ""

if (x== text || text == null) {

return true;

}

return false

}

render() {

return (

<View style={styles.container}>

<Text style={{fontSize: 20, marginLeft:130}}>Quản lý sinh viên</Text>

<TextInput

style={{height: 50, marginLeft: 20, marginRight: 20}}

placeholder="Name"

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

value={this.state.name}

/>

<TextInput

style={{height: 50, marginLeft: 20, marginRight: 20}}

placeholder="Phone"

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

value={this.state.phone}

/>

<Button

style={{height: 30, margin: 20}}

title="Save"

onPress={()=>{

if(this.varTest(this.state.phone) == true || this.varTest(this.state.name) == true){

alert("Vui lòng điền đầy đủ thông tin");

return;

}

const newSV = {name: this.state.name , phone : this.state.phone}

arraySinhVien.push(newSV)

//alert("Hello "+ this.state.name);

this.reloadFlat(this.state.name);

}}

/>

<Text style={{fontSize: 16, marginLeft:130}}>Danh sách sinh viên</Text>

<FlatList

data={arraySinhVien}

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

<SinhVien name = {item.name} phone= {item.phone} parentFlatlist={this}></SinhVien>

}

/>

</View>

);

}

}

const styles = StyleSheet.create({

container: {

flex: 1,

paddingTop: 22

},

item: {

padding: 10,

fontSize: 18,

height: 44,

},

})

*SinhVien.js*

import React,{Component} from 'react';

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

var icon;

export default class SanPham extends Component{

render(){

return (

<View style={styles.product}>

<View style={styles.containerContent}>

<Text> Họ và tên : {this.props.name} </Text>

<Text> Số điện thoại : {this.props.phone} </Text>

</View>

</View>

);

}

}

const styles = StyleSheet.create({

product: {

justifyContent: 'center',

//alignItems: 'center',

backgroundColor: 'lightgreen',

width: 410,

height: 70,

marginBottom : 10

},

containerContent: {

marginTop:-20,

marginLeft: 20

},

});

Code demo các component của Ngô Quốc Cường

*DBConfig.php*

<?php

//Define your host here.

$HostName = "localhost";

//Define your database name here.

$DatabaseName = "mangareader";

//Define your database username here.

$HostUser = "root";

//Define your database password here.

$HostPass = "";

?>

*Select\_manga.php*

<?php

include 'DBConfig.php';

$conn = new mysqli($HostName, $HostUser, $HostPass, $DatabaseName);

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$sql = "SELECT \* FROM manga";

$result = $conn->query($sql);

if ($result->num\_rows >0) {

while($row[] = $result->fetch\_assoc()) {

$tempt = $row;

$json = json\_encode($tempt);

}

} else {

echo "No Results Found.";

}

echo $json;

$conn->close();

?>

*Homepage.js*

import React, { Component } from 'react';

import {

StyleSheet, Text, View, Image,FlatList,

TouchableWithoutFeedback, StatusBar,

TextInput, SafeAreaView, Keyboard, TouchableOpacity,

KeyboardAvoidingView, Button, ToolbarAndroid

} from 'react-native';

import Manga from './itemManga';

export default class homepage extends Component {

// static navigationOptions = {

// header: null

// };

constructor(props) {

super(props);

this.state = {

keyword: "",

isLoading: true

};

}

componentDidMount(){

return fetch('http://10.0.3.2/Select\_Manga.php')

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

.then((responseJson) => {

this.setState({

isLoading: false,

dataSource: responseJson,

}, function(){

});

})

.catch((error) =>{

console.error(error);

});

}

render() {

return (

<View style={styles.container}>

<Text style={{fontSize: 20, marginLeft:130}}>Manga Home</Text>

<TextInput

style={styles.textInput}

placeholder="Name"

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

value={this.state.keyword}

/>

<TouchableOpacity

style = { styles.TouchableOpacityStyle }

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

<Text>Search</Text>

</TouchableOpacity>

<FlatList style={{marginTop:10}}

data={this.state.dataSource}

renderItem={({item}) => <Manga tenMG={item.manga\_name} desMG={item.manga\_des} img={item.image}/>}

/>

</View>

)

}

goSearch()

{

alert("go to search");

}

}

const styles = StyleSheet.create({

container:{

flex: 1,

padding: 20,

},

textInput:{

height: 40,

marginLeft: 15,

marginRight: 15,

marginTop: 10,

borderColor: '#009688',

borderWidth: 1,

borderRadius: 20

},

TouchableOpacityStyle:

{

marginLeft:310,

marginTop:-40,

width: '10%'

},

})

*Itemmanga.js*

import React, { Component } from 'react';

import {

TouchableOpacity,

StyleSheet,

Text,

View,

Image,

} from 'react-native';

export default class itemManga extends Component{

constructor(props){

super(props);

this.state={}

}

render() {

return (

<View style={styles.manga}>

<View style={styles.containerImg}>

<View>

<Image style={styles.img} source={{uri: this.props.img}} />

</View>

</View>

<View style={styles.containerTenMG}>

<Text style={{fontWeight: 'bold'}}> {this.props.tenMG} </Text>

</View>

<View style={styles.containerDesMG}>

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

</View>

</View>

);

}

}

const styles = StyleSheet.create({

manga: {

justifyContent: 'center',

//alignItems: 'center',

backgroundColor: '#DDDDDD',

width: 300,

height: 110,

marginBottom : 10,

borderRadius: 4,

borderWidth: 0.5,

},

idMG:{

marginTop: 300,

},

img: {

marginTop: -50,

width: 70,

height:90,

},

containerImg: {

width: 100,

height: 150,

marginLeft: 10

},

containerTenMG: {

marginTop: -200,

marginLeft: 90,

},

containerDesMG: {

marginTop:0,

marginLeft: 90

},

});

CHƯƠNG 4. PROJECT

CHƯƠNG 5. BÁO CÁO KIẾN NGHỊ