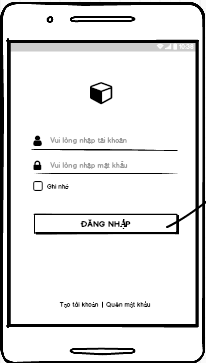
**BÁO CÁO ĐỒ ÁN**

**Chương 1: Tổng quan**

*Đề tài: Quản lí nhân sự*

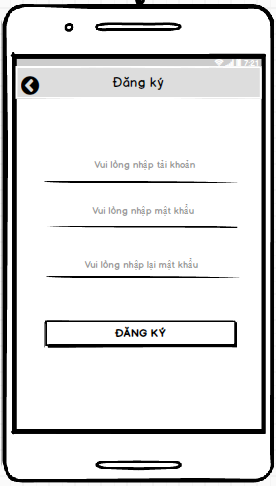
1. **Màn hình đăng nhập (Võ Cao Thanh Phong)**
   1. Giao diện



* 1. Yêu cầu chức năng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stt | Tên | Mô tả | Hoạt động | Trả về |
| 1 | Màn hình đăng nhập | Gồm:   * Logo app * 2 edit text(User và pass word) * Button đăng nhập * Button tạo tài khoản * Button quên mật khẩu * Checkbox | N/A | N/A |
|  | Edit text “user ” | Nhập tài khoản người dùng | Chọn và nhập | N/A |
|  | Edit text “password ” | Nhập mật khẩu người dùng | Chọn và nhập | N/A |
|  | Button đăng nhập | Kiểm tra thông tin đăng nhập | Click chọn | Thông tin đăng nhập đúng thì đăng nhập thành công ngược lại thông báo lỗi sai tài khoản hoặc mật khẩu |
|  | Checkbox ghi nhớ | Lưu tài khoản | Chọn |  |

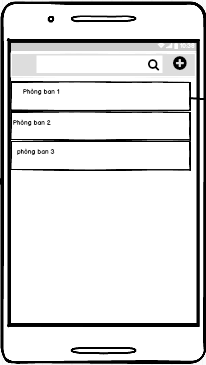
1. **Màn hình tạo tài khoản (Võ Cao Thanh Phong)**

a. Giao diện

b.Yêu cầu chức năng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stt | Tên | Mô tả | Hoạt động | Trả về |
| 1 | Màn hình đăng nhập | Gồm:   * 3 edit text(nhập tài khoản,nhập mật khẩu,xác nhận lại mật khẩu) * Button đăng ký | N/A | N/A |
|  | Edit text “tài khoản ” | Nhập tài khoản người dùng | Chọn và nhập | N/A |
|  | Edit text “mật khẩu ” | Nhập mật khẩu người dùng | Chọn và nhập | N/A |
|  | Edit text “xác nhận lại mật khẩu” | Nhập lại mật khẩu | Chọn và nhập | N/A |
|  | Button đăng ký | Tạo tài khoản mới | Chọn button | N/A |

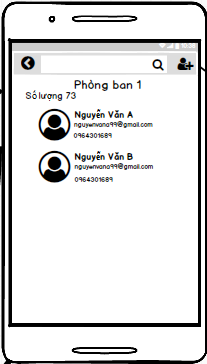
1. **Màn hình chính(Phòng ban) (Lê Thanh Liêm)**
   1. Giao diện



Yêu cầu chức năng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stt | Tên | Mô tả | Hoạt động | Trả về |
| 1 | Màn hình phòng ban | Gồm:   * List view phòng ban * Thanh tìm kiếm * Button thêm phòng ban | N/A | N/A |
| 2 | Listview | Hiện danh sách các phòng ban | Nhấn vào các item có trong listview | Hiển thị Listview nhân viên trong phòng ban đó |
| 3 | Thanh tìm kiếm | Tìm kiếm tên phòng ban | Tìm kiếm theo tên phòng | Trả về tên phòng |
| 4 | Button thêm phòng ban | Thêm dữ liệu phòng ban | Thêm dữ liệu | Tạo thêm item trong listview rồi trả về giá trị |

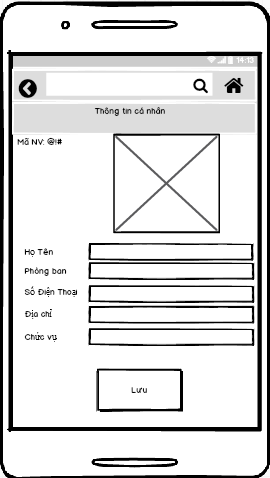
1. **Màn hình nhân viên trong phòng ban (Lê Thanh Liêm)**
2. Giao diện



1. Yêu cầu chức năng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stt | Tên | Mô tả | Hoạt động | Trả về |
| 1 | Màn hình nhân viên trong phòng ban | Gồm:   * List view hiên nhân viên trong phòng ban tương ứng * Thanh tìm kiếm * Text view tên phòng ban * Text view số lượng nhân viên * Button thêm nhân viên * Button trở về | N/A | N/A |
| 2 | Button trở về | Trở về màn hình trước | Trở về | N/A |
| 3 | Item danh sách nhân viên | Hiện nhân viên | Hiện lên danh sách nhân viên | Khi nhấn vào sẽ hiện lên chi tiết nhân viên |
| 4 | Thanh tìm kiếm | Tìm kiếm nhân viên | Tìm kiếm theo tên | Trả về nhân viên có từ khóa gần hoặc đúng nhất |
| 5 | Button Thêm nhân viên | Thêm nhân viên | Thêm nhân viên | Thêm một nhân viên mới vào dữ liệu |
| 6 | Số lượng nhân viên | Cho biết số lượng nhân viên trong phòng ban | Hiện số lựợng | Hiện số lượng được đếm trong listview |

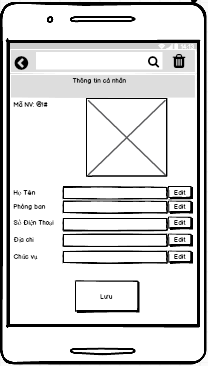
1. **Màn hình thêm nhân viên (Đào Duy Phương)**
2. Giao diện



1. Yêu cầu chức năng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stt | Tên | Mô tả | Hoạt động | Trả về |
| 1 | Màn hình thêm nhân viên | Gồm:   * Các Text view * Có Image(Hình ảnh nhân viên) * Các Edit text * Button lưu | N/A | N/A |
| 2 | Các text view | Hiển thị các mục thông tin nhân viên | N/A | N/A |
| 3 | Image View | Thêm hình ảnh nhân viên | Click chọn | Hiện màn hình thêm hình |
| 4 | Button lưu | Lưu thông tin nhân viên vào danh sách | Click chọn | Màn hình nhân viên của phòng ban đó |

1. **Màn hình thông tin nhân viên (Đào Duy Phương)**
2. Giao diện



1. Yêu cầu chức năng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stt | Tên | Mô tả | Hoạt động | Trả về |
| 1 | Màn hình thông tin nhân viên | Gồm:   * Các Text view * Có Image(Hình ảnh nhân viên) * Các Edit text * Button lưu thêm phòng ban * Button xóa | N/A | N/A |
| 2 | Listview | Hiển thị các mục thông tin nhân viên | N/A | N/A |
| 3 | Thanh tìm kiếm | Thêm hình ảnh nhân viên | Click chọn | Hiện màn hình thêm hình |
| 4 | Button lưu | Lưu thông tin nhân viên vào danh sách | Click chọn | Màn hình nhân viên của phòng ban đó |
| 5 | Button xóa | Xóa nhân viên | Click chọn | Nhân viên bị xóa khỏi dữ liệu |

**CHƯƠNG 2: ES6**

1. ***Biến Let, Var, Conts trong ES6 (Lê Thanh Liêm)***

* Biến let: từ khóa let dùng để khởi tạo một biến nhưng biến đó chỉ có tác dụng bên trong khối đang khai báo

if (a < b)

{

    let tmp = a;

    //biến let chỉ tồn tại trong phạm vi này

}

* Biến var: Dùng chung trong toàn bộ chương trình

var a = 12;

var b = 20;

* Conts: Hằng

1. ***Arrow function (Võ Cao Thanh Phong)***

Arrow function : dùng để viết một function một cách ngắn ngọn,sử dụng kí tự ‘=>’

* Cú pháp căn bản**:**

var functionName = (var1, var2) => {

    // Nội dung function

};

* Ví dụ:

var hello = (name, message) => {

    console.log("Chào " + name + ", bạn là " + message);

};

hello('Cường', 'admin freetuts.net');

1. ***Kiểu DL Object (Đào Duy Phương)***

Đối tượng là một khái niệm trừu tượng thể hiện cho một đối tượng cụ thể mà chúng ta có thể tự tạo một đối tượng theo ý của mình dựa vào yêu cầu của ứng dụng.

**Cách 1**: Sử dụng từ khóa new Object().

var Students= new Object();

**` Cách 2**: Sử dụng từ khóa {}.

var Students={};

**Chương 3: Component**

1. **Tổng Quan**

Trong ReactJS, mỗi đoạn code sẽ được phân chia thành những Component không lệ thuộc lẫn nhau và có thể tái sử dụng khi cần thiết.

***Props(Võ Cao Thanh Phong)***

Các component nhận props từ component cha. Chúng ta không được thay đổi giá trị của props trong các component này mà chỉ được phép đọc giá trị ra thôi. Trong React thì dữ liệu sẽ đi theo một chiều, có nghĩa là từ component cha => các component con.

Ví dụ:

// Parent

export default class ScreenOne extends React.Component {

render () {

return (

<View>

<Heading message={'Custom Heading for Screen One'}/>

</View>

)

}

}

// Child component

export default class Heading extends React.Component {

render () {

return (

<View>

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

</View>

)

}

}

Heading.propTypes = {

message: PropTypes.string

}

Heading.defaultProps = {

message: 'Heading One'

}

***State(Võ Cao Thanh Phong)***

State thì hoạt động khác với props. state là dữ liệu nội bộ của một Component, trong khi props là dữ liệu được truyền cho Component. Chính vì vậy chúng ta hoàn toàn có thể thay đổi state

Ví dụ:

class Form extends React.Component {

constructor (props) {

super(props)

this.state = {

input: ''

}

}

handleChangeInput = (text) => {

this.setState({ input: text })

}

render () {

const { input } = this.state

return (

<View>

<TextInput style={{height: 40, borderColor: 'gray', borderWidth: 1}}

onChangeText={this.handleChangeInput}

value={input}

/>

</View>

)

}

}

***Style trong React Native(Đào Duy Phương)***

Tất cả các thành phần cơ bản sẽ sử dụng là một tham số có tên là style. Những tham số style và giá trị của chúng tương tự với cách hoạt động của CSS trên môi trường Web, chỉ có vài khác biệt, ví dụ như backgroundColor thay vì sử dụng background-color trên CSS.

Ví dụ:

const styles = StyleSheet.create({

bigblue: {

color: 'blue',

fontWeight: 'bold',

fontSize: 30,

},

red: {

color: 'red',

},

});

***Height & Width(Đào Duy Phương)***

Chiều rộng và dài của các thành phần sẽ xác định kích thước của các thành phần giao diện trên màn hình.

Ví dụ:

class FixedDimensionsBasics extends Component {

render() {

return (

<View>

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

<View style={{width: 100, height: 100, backgroundColor: 'skyblue'}} />

<View style={{width: 150, height: 150, backgroundColor: 'steelblue'}} />

</View>

);

}

};

***FlexBox(Lê Thanh Liêm)***

Một thành phần có thể chỉ định rõ bố cục cho các view con của nó bằng cách sử dụng thuật toán của flexbox. Flexbox được thiết kế để cung cấp giao diện tương đồng trên những màn hình có kích thước khác nhau.

Ví dụ:

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. **Các component cơ bản**
   * Text Component
   * TextInput Component
   * Button Component
   * CheckBox
   * Radio Button
2. **Các component nâng cao**
   * ListView
   * Picker
   * Switch Button
   * Slider
   * DrawerLayout
   * ViewPager
   * WebView

**Chương 4: xây dựng ứng dụng**

**Chương 5: kết luận và kiến nghị**

Tài liệu tham khảo

***\*Cách xử lý mảng trong ES6 (Võ Cao Thanh Phong)***

***\*Cách sử dụng Map trong ES6 (Lê Thanh Liêm)***

***\*Cách sử dụng Set trong ES6 (Đào Duy Phương)***