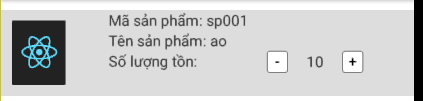
Tạo Project

React-native init DemoProject

React-native run-android

react-native log-android

ProductItem.js:



import React, {Component} from 'react';

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

export default class Product extends Component {

constructor(props) {

super(props)

this.state = {

quantity: props.quantity ? parseInt(props.quantity) : 0

}

}

\_onLowQuantitiy = () => {

this.setState({

quantity: this.state.quantity - 1

})

}

\_onGrowquantity = () => {

this.setState({

quantity: this.state.quantity + 1

})

}

render() {

return (

<View style={{ flex: 0,flexDirection: 'row', backgroundColor: '#ddd', borderColor: '#f9f9f9', marginBottom: 5 }}>

<View style={{ width: 100, height: 80, borderBottomColor: '#ddd', }}>

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

</View>

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

<Text>Mã sản phẩm: {this.props.code}</Text>

<Text>Tên sản phẩm: {this.props.name}</Text>

<View style={{ flex: 0, flexDirection: 'row' }}>

<Text style={{ width: 100 }}>Số lượng tồn:</Text>

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

<TouchableOpacity style={{ width: 20, height: 20 }} title='-' onPress={this.\_onLowQuantitiy}>

<Text style={{ backgroundColor: 'white', borderColor: 'gray', borderWidth: 1, borderRadius: 3, textAlign: 'center', color: 'black', }}>-</Text>

</TouchableOpacity>

<Text style={{ width: 50, alignItems: 'center', textAlign: 'center' }}>{this.state.quantity}</Text>

<TouchableOpacity style={{ width: 20, height: 20 }} title='-' onPress={this.\_onGrowquantity}>

<Text style={{ backgroundColor: 'white', borderColor: 'gray', borderWidth: 1, borderRadius: 3, textAlign: 'center', color: 'black', }}>+</Text>

</TouchableOpacity>

</View>

</View>

</View>

</View>

);

}

}

const styles = StyleSheet.create({

container: {

flex: 1,

},

Price: {

color: '#333333',

},

img: {

width: 50,

height:60,

marginLeft: 10,

marginTop: 10,

},

Image: {

marginRight: 5,

marginBottom: 5,

width: 70,

height: 70,

resizeMode: 'contain'

},

});

ProductList.js



import React, {Component} from 'react';

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

import ProductItem from './ProductItem';

import {host} from './app.json';

export default class ProductList extends React.Component {

static navigationOptions = {

title: 'Products',

};

constructor(props){

super(props);

this.state ={ isLoading: true}

}

componentDidMount(){

return fetch(host+'/LTDD3WS/ProductGet.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:10}}>

<FlatList

data={this.state.dataSource}

renderItem={({item}) => <ProductItem code={item.code} name={item.name} quantity={item.quantity} image={item.image}/>}

/>

<Button style={{paddingTop:10}}

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

title="Add New Product"

color="#841584"

accessibilityLabel="Learn more about this purple button"

/>

</View>

);

}

}

Index.js

/\*\* @format \*/

import {AppRegistry} from 'react-native';

import {name as appName} from './app.json';

import App from './App';

// import App from './ProductList';

AppRegistry.registerComponent(appName, () => App);

App.js

import React from 'react';

import { createStackNavigator, createAppContainer } from 'react-navigation'; // Version can be specified in package.json

import ProductList from './ProductList';

import ProductAdd from './ProductAdd';

const RootStack = createStackNavigator(

{

ProductList: ProductList,

ProductAdd: ProductAdd,

},

{

initialRouteName: 'ProductList',

}

);

const AppContainer = createAppContainer(RootStack);

export default class App extends React.Component {

render() {

return <AppContainer />;

}

}