# Department of Computing

**CS-213: Advanced Programming**

**Class: BSCS 7AB**

# Lab 12: React Native Login Application

**Date: 12 December, 2019**

**Time: 10:00-01:00pm & 02:00-05:00pm**

# Instructor: Dr. Sidra Sultana

**Lab Engineer: Ms. Ayesha Asif**

# 

# Lab 12: React Native Login Application

Ershad Hussain (237885)

BSCS-7A

**Lab Task**

Create a login Application in react native

|  |
| --- |
| Solution |
| Task Code:  App.js  import React, { Component } from 'react';  import {  StyleSheet,  View,  StatusBar  } from 'react-native';  import Routes from './Routes.js';  export default class App extends Component{  render() {  return (  <View style={styles.container}>  <StatusBar  backgroundColor="#002f6c"  barStyle="light-content"  />  <Routes/>  </View>  );  }  }  const styles = StyleSheet.create({  container: {  flex: 1,  }  });  Form.js  import React, { Component } from 'react';  import { StyleSheet, Text, View, TextInput, TouchableOpacity, AsyncStorage, Keyboard } from 'react-native';  export default class Form extends Component {  constructor(props)  {  super(props);  this.state=  {  email:'',  password: ''  }  }  saveData =async()=>{  const {email,password} = this.state;  let loginDetails={  email: email,  password: password  }  if(this.props.type !== 'Login'){  AsyncStorage.setItem('loginDetails', JSON.stringify(loginDetails));  Keyboard.dismiss();  alert("You successfully registered. Email: " + email + ' password: ' + password);  this.login();  }  else if(this.props.type == 'Login'){  try{  let loginDetails = await AsyncStorage.getItem('loginDetails');  let ld = JSON.parse(loginDetails);  if (ld.email != null && ld.password != null) {  if (ld.email == email && ld.password == password)  alert('Logged In!');  }  else  {  alert('User does not exist!');  }  }  //}  catch(error)  {  alert(error);  }  }  }  showData = async()=>{  let loginDetails = await AsyncStorage.getItem('loginDetails');  let ld = JSON.parse(loginDetails);  alert('email: '+ ld.email + ' ' + 'password: ' + ld.password);  }  render() {  return(  <View style={styles.container}>  <TextInput style={styles.inputBox}  onChangeText={(email) => this.setState({email})}  underlineColorAndroid='black'  placeholder="Email"  placeholderTextColor = "black"  selectionColor="black"  keyboardType="email-address"  onSubmitEditing={()=> this.password.focus()}/>  <Text>{'\n'}</Text>  <TextInput style={styles.inputBox}  onChangeText={(password) => this.setState({password})}  underlineColorAndroid='black'  placeholder="Password"  secureTextEntry={true}  placeholderTextColor = "black"  ref={(input) => this.password = input}  />  <Text>{'\n'}</Text>  <Text>{'\n'}</Text>  <TouchableOpacity style={styles.button}>  <Text style={styles.buttonText} onPress={this.saveData}>{this.props.type}</Text>  </TouchableOpacity>  </View>  )  }  }  const styles = StyleSheet.create({  container: {  justifyContent: 'center',  alignItems: 'center',  },  inputBox: {  width: 300,  backgroundColor: '#fff',  paddingHorizontal: 20,  fontSize: 12,  color: '#000',  marginVertical: 10  },  button: {  width: 300,  backgroundColor: '#000',  marginVertical: 10,  paddingVertical: 12  },  buttonText: {  fontSize: 18,  fontWeight: '500',  color: 'white',  textAlign: 'center'  }  });  Routes.js  import React, { Component } from 'react';  import {Router, Stack, Scene} from 'react-native-router-flux';  import Login from './login';  import Signup from './signup';  export default class Routes extends Component {  render() {  return (  <Router hideNavBar={false}  navigationBarStyle={styles}  titleStyle={styles}  >  <Stack key="root">  <Scene key="login" component={Login} title="Login"/>  <Scene key="signup" component={Signup} title="Sign up"/>  </Stack>  </Router>  )  }  }  const styles = {  backgroundColor: 'white',  color: '#000',  }  Login.js  import React, { Component } from 'react';  import { StyleSheet, Text, View, TextInput, TouchableOpacity, AsyncStorage, Keyboard } from 'react-native';  import {Actions} from 'react-native-router-flux';  import Form from './Form';  export default class Login extends Component {  signup() {  Actions.signup()  }  render() {  return(  <View style={styles.container}>  <Text>{'\n'}</Text>  <Text>{'\n'}</Text>  <Form type="Login"/>  <View style={styles.signupTextCont}>  <Text style={styles.signupText}>Don't have an account </Text>  <TouchableOpacity onPress={this.signup}><Text style={styles.signupButton}>Signup</Text></TouchableOpacity>  </View>  </View>  )  }  }  const styles = StyleSheet.create({  container: {  flex: 1,  //justifyContent: 'center',  alignItems: 'center',  //backgroundColor: 'gray',  },  signupTextCont: {  flexGrow: 1,  //justifyContent: 'center',  alignItems: 'flex-end',  paddingVertical: 16,  flexDirection: 'row',  },  signupText: {  color: '#000',  fontSize:20,  },  signupButton: {  color: 'blue',  fontSize:20,  fontWeight: '600',  }  });  Signup.js  import React, { Component } from 'react';  import {  StyleSheet,  Text,  View,  TextInput,  TouchableOpacity,  AsyncStorage,  Keyboard,  } from 'react-native';  import { Actions } from 'react-native-router-flux';  import Form from './Form';  export default class Signup extends Component {  goBack() {  Actions.pop();  }  render() {  return (  <View style={styles.container}>  <Text>{'\n'}</Text>  <Text>{'\n'}</Text>  <Form type="Signup" />  <View style={styles.signupTextCont}>  <Text style={styles.signupText}>Already have an account? </Text>  <TouchableOpacity onPress={this.goBack}>  <Text style={styles.signupButton}>Sign in</Text>  </TouchableOpacity>  </View>  </View>  );  }  }  const styles = StyleSheet.create({  container: {  flex: 1,  //justifyContent: 'center',  alignItems: 'center',  backgroundColor: 'gray',  },  signupTextCont: {  flexGrow: 1,  //justifyContent: 'center',  alignItems: 'flex-end',  paddingVertical: 16,  flexDirection: 'row',  },  signupText: {  color: '#000',  fontSize: 20,  },  signupButton: {  color: '#000',  fontSize: 20,  },  });  Task Output Screenshot: |

### Deliverable

Compile a single word document by filling in the solution part and submit this Word file on LMS. This lab grading policy is as follows: The lab is graded between 0 to 10 marks. The submitted solution can get a maximum of 5 marks. At the end of each lab or in the next lab, there will be a viva/quiz related to the tasks. You must show the implementation of the tasks in the designing tool, along with your complete Word document to get your work graded. You must also submit this Word document on the LMS. In case of any problems with submissions on LMS, submit your Lab assignments by emailing it to Ms. Ayesha Asif: [ayesha.asif@seecs.edu.pk](mailto:ayesha.asif@seecs.edu.pk).