SMU MedTech
Mediterranean
Institute of Technology

ABET

CS325 - Mobile Software Development

Lab 4: Dating App – Part 3

Pressable, Images and Hooks

Sahar Sarraj – Prof. Bochra Khiari

2024-2025

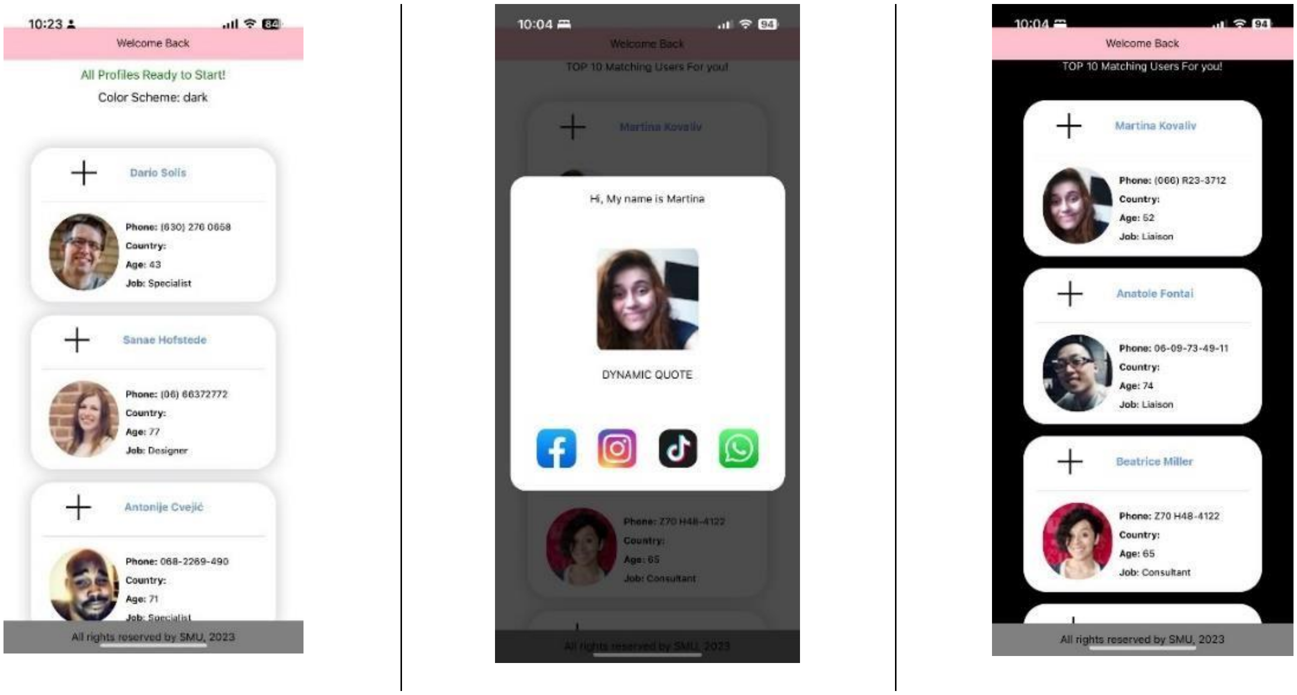
Lab introduction

The goal of Lab 4 is to continue working on the dating app and to focus on new components such as pressable, image, and modal. Additionally, you will delve into the utilization of the useColorScheme hook, integrate the NavigationContainer, and apply styling through theme-related props.

Goals

1. Modifying the user component by adding a **pressable component** and a **modal** that opens the user details: it includes the user's full name and picture as well as images to display social media accounts.
2. Add and use **useColorScheme** hook in the AllProfilesMain component to get to detect the theme selected in device settings.
3. Add **NavigationContainer** and use the prop theme to create dark/light theme around the app.

The screens should look like this:



10:23

Welcome Back

All Profiles Ready to Start!

Color Scheme: dark

Dario Solis

Phone: (630) 276 0658
Country:
Age: 43
Job: Specialist

Sanae Hofstede

Phone: (06) 66372772
Country:
Age: 77
Job: Designer

Antonije Cvejić

Phone: 068-2269-490
Country:
Age: 71
Job: Specialist

All rights reserved by SMU, 2023

10:04

Welcome Back

TOP 10 Matching Users For you!

Martina Kovaliv

Hi, My name is Martina

DYNAMIC QUOTE

Phone: Z70 H48-4122
Country:
Age: 65
Job: Consultant

All rights reserved by SMU, 2023

10:04

Welcome Back

TOP 10 Matching Users For you!

Martina Kovaliv

Phone: (066) R23-3712
Country:
Age: 52
Job: Liaison

Anatole Fontai

Phone: 06-09-73-49-11
Country:
Age: 74
Job: Liaison

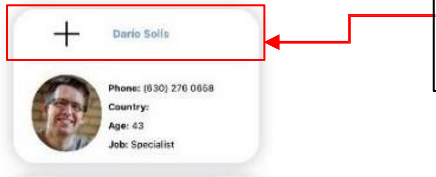
Beatrice Miller

Phone: Z70 H48-4122
Country:
Age: 65
Job: Consultant

All rights reserved by SMU, 2023

Figure 1: Screenshots from the lab goal

Part 1: Modify the user component


Dario Solís

Phone: (630) 276 0658
Country:
Age: 43
Job: Specialist

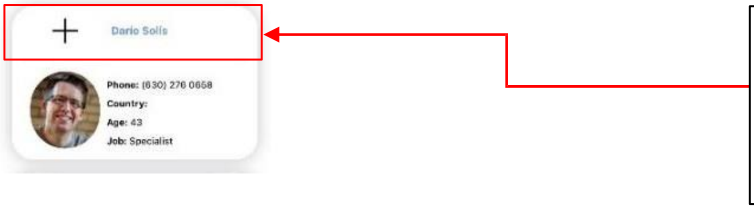
A Pressable component with a child Image

As we notice in the picture from figure 1, the change made is that we added a plus button in the card. To reach the same results follow the following instructions:

1. Open the User component where we defined the card component.
2. Add Pressable and Image from the following import:

```
import { View, Text, Pressable, Image } from "react-native";
```

3. Add a Pressable component inside the <Card.Title> Component, before the first name and last name.


Dario Solís

Phone: (630) 276 0658
Country:
Age: 43
Job: Specialist

```
<Card.Title>
    <View>
          <Pressable>
                <Image/>
          </Pressable>
          <Text/>
      </View>
<Card.Title>
```

4. Download the “plus-sign” image from Moodle and add it under the folder assets.
5. Create an Image component as a child of the Pressable component.

```
source={require("../assets/plus-sign.png")}
```

6. Let’s add some inline styling to the Image component:

```
width: 40, height: 40,
```

7. Wrap the firstName and lastName under a Text component if they are not already.

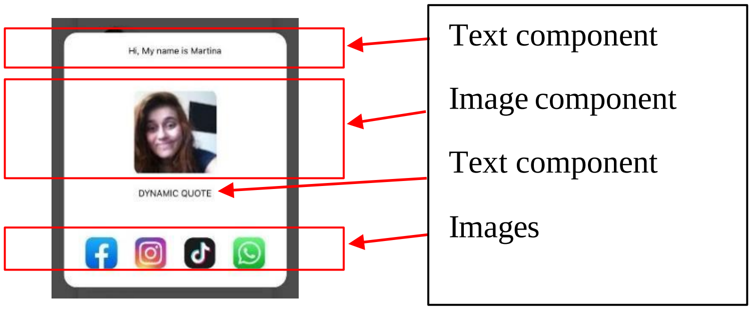
8. Center both the Image and the Text component by adding style in the parent View component.
9. Add the onPress property, which should take the function toggleModal, to the Pressable component. Console log a text or number to observe its functionality.
10. Add Modal library by following this link: https://www.npmjs.com/package/react-native-modal

```
npm i react-native-modal
```

11. Let's incorporate the required code from the example in the Modal library documentation from the “A complete example” section. Add the <Modal> before or after the <Card> Component.
12. Hide the modal by pressing outside of its content by using the onBackdropPress function of the Modal component.

```
onBackdropPress={() => setModalVisible(false)}
```

13. Inside the modal display a Text component, Image, another Text and a list of images displayed in a row direction.



Part 2: Add and use useColorScheme hook in the AllProfilesMain component

1. Let’s add a Text component inside the **AllProfilesMain.js** file.

2. Add the necessary imports to use useColorScheme.

```
import { useColorScheme} from "react-native";
```

3. Initialize the useColorScheme hook.

```
 const colorScheme = useColorScheme();
```

4. Let’s change the text color for the Text component: when the colorScheme is dark, set the text color to green else if its light then set it to red. (feel free to customize this!)

```
 style={[

      styles.title,

      colorScheme === "light" ? { color: "red" } : { color: "green" },

  ]}
```

Add the NavigationContainer and import the Dark/Light themes in your App.js

1. Install the required packages in your React Native project:

```
npm install @react-navigation/native-stack
```

2. Add the necessary imports:

```
import { NavigationContainer } from "@react-navigation/native";
import { createNativeStackNavigator } from "@react-navigation/native-stack";

import { DefaultTheme, DarkTheme } from "@react-navigation/native";
```

3. Create an instance of the stack navigator:

```
const Stack = createNativeStackNavigator();
```

4. Import and call the useColorScheme hook to retrieve the user's preferred color scheme (light or dark)

5. Define two Theme objects:
    a. LightTheme:
        i. Inherits properties from DefaultTheme
        ii. Sets dark to true to indicate a light theme.
        iii. Set the property text to black and property background to white
    b. DarkTheme:
        i. Inherits properties from DarkTheme
        ii. Sets dark to false to indicate a dark theme.
        iii. Set the property text to white and property background to black
6. Use the hook to conditionally apply themes inside the NavigationContainer in App.js:

```
theme={colorScheme === "light" ? LightTheme : CustomDarkTheme}
```

Assignment:

1. Add quotes inside the modal defined in the user component. Use the following API https://api.quotable.io/random?tags=love
2. Add a switch button that allows you to toggle between Dark and Light mode in all the screens.
3. Set a background image to each screen in the Dating App.