

COMP 306 – Fall 2018 – Assignment 05
Due: Monday, 8 Oct, beginning of class

Problem: This assignment is an extension of assignment 4. We still want to create an address book app. In reality, the address book would be stored in a database on a server, but we're going to ignore this part. Instead, we'll just create an app that allows me to enter the items below for a person. Store all the information in the app state and refresh that state in a separate class from the exported class. In this assignment we'll add a list so that we can store multiple people and we will also implement a 2 screen app.

You must store the information for each person in the app *state* using a *list* and display it using a **flatlist** component.

Your app must have 2 screens. One screen (the “home” screen that appears when the app is started) will be used to display the data in the list. The second screen will be used to enter a new person.

The home screen will have a **flatlist** component to display the information for all people in your list and a button with the title “add person” to go to the second screen. Initialize your list with some hard-coded data (say 5 or so people). Instead of a Text field use a **flatlist** component to display the information for all people in your list. Don't use push to change screens. This will prevent problems like the home screen calling itself (without passing the necessary data to add a person),

The second screen must have TextInput fields for entering all of the data for a person. . Allow the user to enter the following information:

First Name
Last Name
Email address
Age

This screen should also have a button labeled “Update”. When this button is pressed, go to the home screen and add the new person to the list. The screen must also have a button labeled “Cancel” which will go to the home screen but won't update the list. You'll need to think how to pass information to the home screen so that it knows when to update the list. You will have to use this.props.navigation.navigate() function to go back to the home screen and pass parameters.

The app does **not** have to display the words “Legal” or “Not Legal”. Your app must be styled in an aesthetically pleasing way. This will be graded.

~~Each screen must have a colored header bar with unique text.~~

Submission. I'll look at your app in lab. Submit your App.js file and any other files that you create through Sakai.