**COMM644  
Assignment 11, Part2**

**The Mobile Employee Directory (40 points + 25 possible extra credit points)**

In this part of the assignment you will work with jQuery, jQuery Mobile, AJAX, and more, to create a searchable employee directory. This is your chance to not only acquire the points needed for this assignment but also to pick up 25 extra credit points. Everything related to extra credit is highlighted in red.

**General**

Divide up your project so that your web page is at the root. I’ve already included all of the headshots that you will need. Those are contained within an images folder. Separate out all of your JavaScript files into separate .js scripts and place those within a folder named js. You will have 1 external JavaScript file for each “web page” in your application. Your styles should be contained within a folder called CSS. All of your JSON data should be contained within a single JSON file and that will be stored within a folder called data.

There will be 4 “web pages” in the application. All of these can be placed within index.html.

The core of the application should be built using jQuery Mobile. Backend code will be done using JavaScript/jQuery. Service calls will be made via AJAX (XMLHttpRequest or jqXHR…take your pick).

**Data**

Your JSON data store should contain all of the relevant information for all 12 employees within your organization. Make sure to include the following entries:

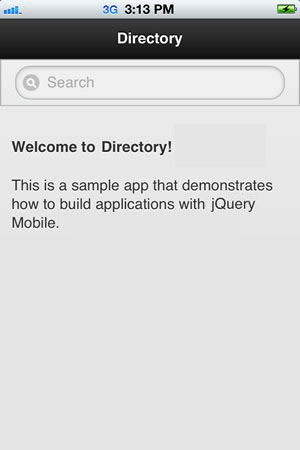
* ID (used in relationship between managers and subordinates)
* Name
* Title
* ImagePath (points to image file in images folder)
* OfficeNumber
* CellNumber
* Email
* ReportsTo (points to the ID of the employee that this person reports to)

The names of all of the employees are listed as the filenames for each image in the images directory. Reference those when adding names in your JSON file. Make up the data for the rest of the fields.

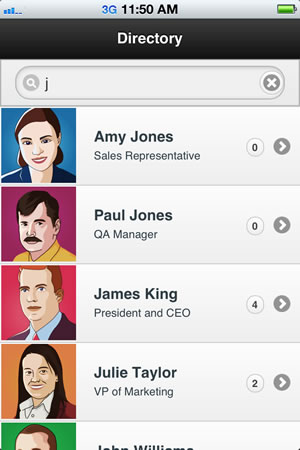
Create at least 2 relationships between employees and their managers. For instance, look at the second image below. James King and Julie Taylor each have subordinates. James King has 4 and Julie Taylor has 2. This can be set up in the JSON file by adding the ID of the manager to the ReportsTo field for the subordinate employee.

**The Application**

The application is made up entirely of jQuery Mobile. The main page will contain a header, a search box, and a basic introduction to the application. See the image below for a reference.

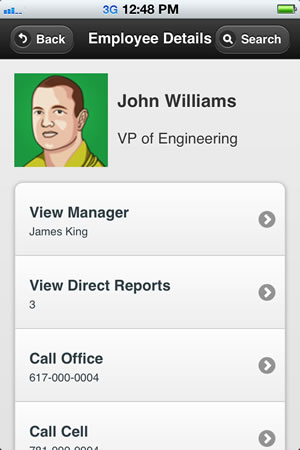


When a user searches for an employee, a filtered list will automatically appear based on the searched term. As you can see in the example below, “j” is entered into the search box and all matches that contain the letter “j” are shown.



Count up the number of subordinates for each employee in the list and display that value within the count bubble in the list.

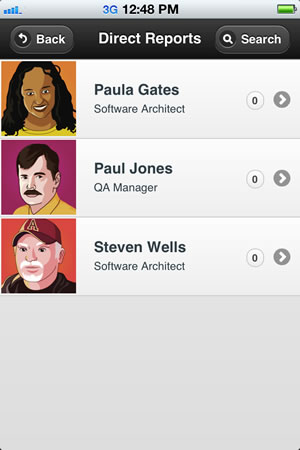
Once a user taps on an employee, details for that employee should be shown. The way I want everything laid out is displayed in the image below.



Notice that all of the details of the employee are shown. Also notice the toolbar at the top of the screen which include a back button and a search button that should redirect the user back the main screen to perform a new search. If the user taps either the Call Office or Call Cell buttons, the phone feature in the user’s mobile device should be engaged and a phone call should be automatically attempted.

If the user taps View Manager, the manager for that employee should be shown.

If the user taps View Direct Reports (assuming they’re a manager and they have subordinate employees), a list similar to the one outlined below should be shown.



If the user taps on one of the subordinates, it should redirect back to the employee details screen and that employee should be shown instead.