ITIS/ITCS 4180/5180 Mobile Application Development
In Class Assignment 11

Basic Instructions:

1. In every file submitted you MUST place the following comments:
   a. Assignment #.
   b. File Name.
   c. Full name of all students in your group.
2. Each group should submit only one assignment. Only the group leader is supposed to submit the assignment on behalf of all the other group members.
3. Please download the support files provided with this assignment and use them when implementing your project.
4. Export your Android project as follows:
   a. From eclipse, choose *"Export…"* from the File menu.
   b. From the Export window, choose *General* then *File System*. Click *Next*.
   c. Make sure that your Android project for this assignment is selected. Make sure that all of its subfolders are also selected.
   d. Choose the location you want to save the exported project directory to. For example, your *Desktop* or *Documents* folder.
   e. When exporting make sure you select *Create directory structure for files*.
   f. Click Finish, and then go to the directory you exported the project to. Make sure the exported directory contains all necessary files, such as the .java and resource files.
5. Submission details:
   a. When you submit the assignment, compress your exported Android project into a single zip file. The format of compressed file name is **Group#_InClassAssignment#.zip**
   b. You should submit the assignment through Moodle: Submit the zip file.
6. **Failure to follow the above instructions will result in point deductions.**

In Class Assignment 11

In this assignment you will get familiar with creating a simple PhoneGap project, that saves contacts into the device contacts using standard web technology (HTML5, JS and CSS) and Cordova Library.

Important App Requirements:

1. Android SDK should be installed, the following paths should be added to you environment variable $PATH: adt-bundle/sdk/platform-tools and adt-bundle/sdk/tools.
2. Cordova library should be installed. Installation Guide (http://docs.phonegap.com/en/edge/guide_cli_index.md.html#The%20Command-Line%20Interface)
3. You should have basic knowledge of Web development
4. Familiarize your self with jQuery and jQuery Mobile.
5. Familiarize your self with Cordova Contacts plugin.
6. This project targets Android API 19.

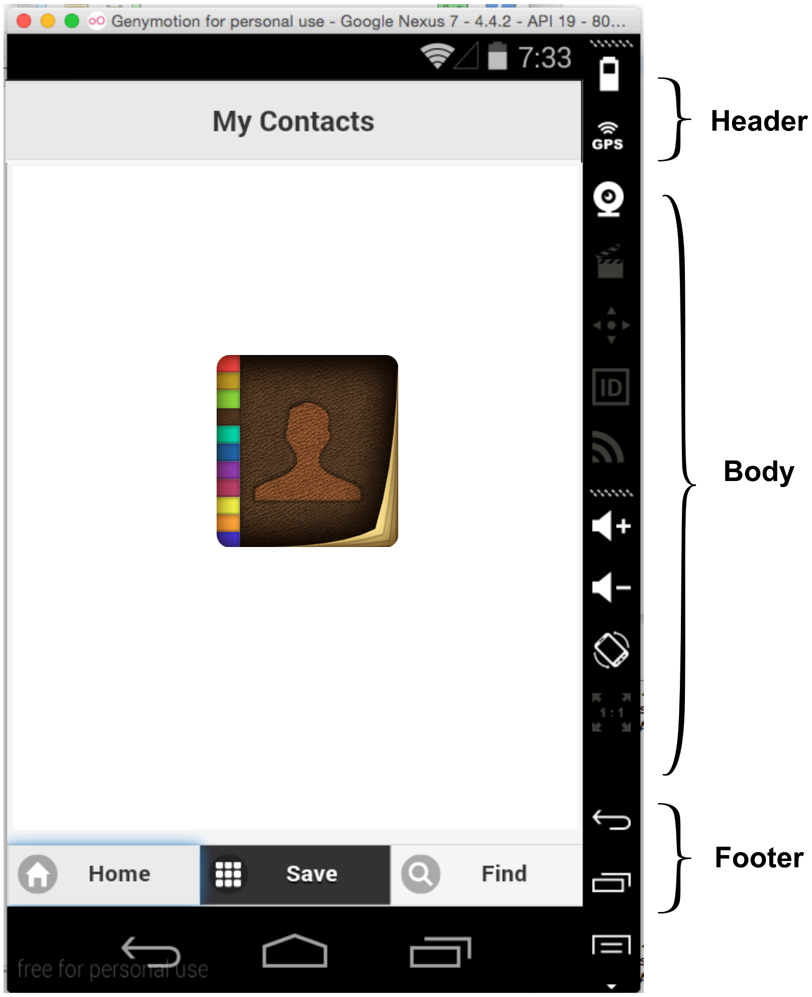


Figure-1: App Home Page

Part 1: Web App UI Design (jQuery + jQuery Mobile)

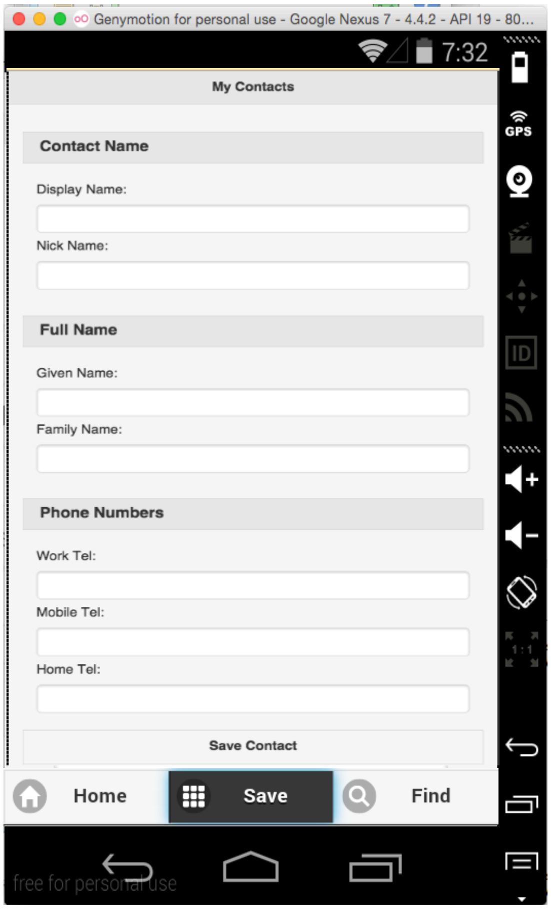
My Contacts App is a web-based app that is build using **jQuery** and **jQuery Mobile**. It is a single based app that basically use hash processing to switch between views on the client side in the same page without the need to post back to the server.

A single-page application (SPA), is a web application or web site that fits on a single web page with the goal of providing a more fluid user experience akin to a desktop application. In an SPA, either all necessary code – HTML, JavaScript, and CSS – is retrieved with a single page load, or the appropriate resources are dynamically loaded and added to the page as necessary, usually in response to user actions. The page does not reload at any point in the process, nor does control transfer to another page, although modern web technologies (such as those included in the HTML5 pushState() API) can provide the perception and navigability of separate logical pages in the application. Interaction with the single page application often involves dynamic communication with the web server behind the scenes.

In jQuery Mobile, there are many functions that helps navigate between different views in a single based app. To see code example, See : http://demos.jquerymobile.com/1.4.5/navigation/. In this example, we will use hash processing to switch between view. For code example, See: http://demos.jquerymobile.com/1.4.5/navigation-hash-processing/

1. Start with **index.html**, which will initially have a basic html structure (header, body , footer). See: http://demos.jquerymobile.com/1.4.5/pages/
2. The header displays “My Contacts”
3. The footer will hold a navigation bar of three main links, home , Save and Find which will help the user navigate through the different unctions of the App. To design a similar navigation bar, See: http://demos.jquerymobile.com/1.4.5/navbar/
4. The Home view will display an image, which is the logo of the app. You are free to choose any icon related to Contacts App.
5. The Save link will show the Save View. Save view will display a form of input fields as shown in Figure-2. It contains the following elements:
   (a) **Input fields** for Display Name, Name, Given Name, Family Name and Phone Numbers. For code examples , See: http://demos.jquerymobile.com/1.4.5/textinput/
   (b) **Groups**. Note that fields are grouped according to their type, examples like Full Name and Phone Numbers. For Code example , See: http://demos.jquerymobile.com/1.4.5/body-bar-classes/
   (c) **Button Save**. We will discuss the functionality of the button in the next section. For code example , See : http://demos.jquerymobile.com/1.4.5/button-markup/
6. The Find link well show the Find View. This view will display all your contacts and enable you to search them. It contains the following elements :
   (a) Filter reveal. This field enables autocomplete with local data. For code example, See: **http://demos.jquerymobile.com/1.4.5/listview/**

(b) **Divided List**. Listview that displays display Name + mobile number , divided alphabetically. For code example , See: http://demos.jquerymobile.com/1.4.5/listview/


Genymotion for personal use - Google Nexus 7 - 4.4.2 - API 19 - 80...
7:32
GPS
ID
1:1
free for personal use

My Contacts

Contact Name

Display Name:

Nick Name:

Full Name

Given Name:

Family Name:

Phone Numbers

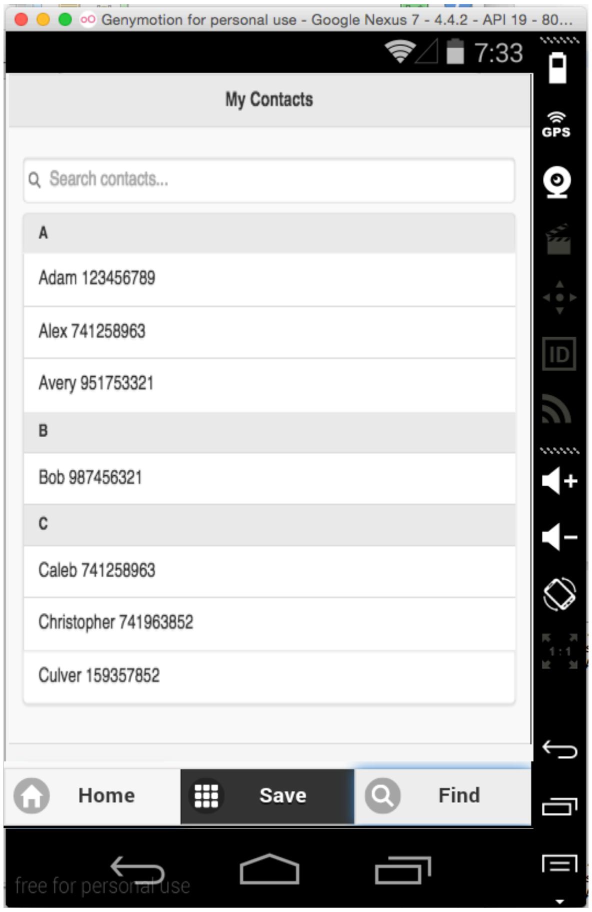
Work Tel:

Mobile Tel:

Home Tel:

Save Contact

Figure-2: Save View


Genymotion for personal use - Google Nexus 7 - 4.4.2 - API 19 - 80...
7:33
GPS
ID
1:1
free for personal use

My Contacts

Search contacts...

A

Adam 123456789

Alex 741258963

Avery 951753321

B

Bob 987456321

C

Caleb 741258963

Christopher 741963852

Culver 159357852

Figure-3: Find View

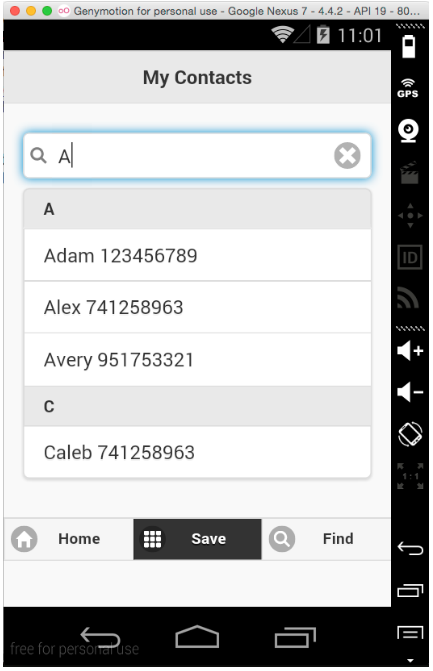
Part 2 : Functionality (Cordova Library)

1. Create New Cordova project
```
cordova create ContactsApp edu.uncc.contacsApp ContactsApp
```
2. Add Android platform
```
cordova platform add android
```
3. In this App we will two plugins, Contacts and Notification.
```
cordova plugin add org.apache.cordova.contacts
cordova plugin add org.apache.cordova.dialogs
```
4. Build Project, Open in Eclipse
5. Copy your index.html file, JS and CSS files into the www folder. At this point your PhoneGap app should look like the figures above. There is no functionality

implemented related to Contacts yet. Next steps will we start implementing the functions related to Contacts plugin.

6. Save Function: When the Save link is clicked the app will show input fields where the user can enter contact details. The Button Save should save all the details entered into the mobile contacts.
   (a) The corresponding functions/ objects provided by Contacts API related to this task are: navigator.contacts.create(…), Contact, ContactName, ContactField, Contact.save(..). See: http://plugins.cordova.io/#/package/org.apache.cordova.contacts
   (b) Once the Contact is saved Successfully , you app should display an **alert** that displays the message “Contact Saved Successfully!”. If there was any issue with saving the contact, the alert should display the corresponding error message. To learn how to use alerts see: http://plugins.cordova.io/#/package/org.apache.cordova.dialogs

7. Find Function: When the Find link is clicked the app should query ***all contacts*** saved in the device and populate it into the Divided listView.
   (a) The corresponding functions/ objects needed to do this task are: navigator.contacts.find(..), ContactFindOptions and ContactFieldType. See: http://plugins.cordova.io/#/package/org.apache.cordova.contacts
   (b) For the sake of this app, we only need two contact fields , display Name and mobile number.
   (c) Once you get all the contacts, you need to build the list view dynamically based on the data retrieved. Hint: use **`document.write(..)`**
   (d) link the filter field with the list View so that the list is filtered according to user input see Figure-4.


Genymotion for personal use - Google Nexus 7 - 4.4.2 - API 19 - 80...
11:01
GPS
ID
Home
Save
Find
free for personal use

My Contacts

A

A

Adam 123456789

Alex 741258963

Avery 951753321

C

Caleb 741258963

Figure-4: Filtering Contacts