

CSCI-4131 Internet Programming Fall 2015

Assignment 6: PHP basics

Due: 18th November, 2015

Objective:

In this assignment, we will build a web service that finds out the nearby places of interest, based on the information from Foursquare. Foursquare is a world leading location-based social networking service. Besides the common social networking functionalities like making friends, posting photos, users can also check-in and explore the nearby venues.

To accomplish the task, you will need to use some knowledge of JavaScript and Google Map APIs, some idea of JSON format, as well as some simple PHP code.

Part I: Requirements

Requirements:

1. Show a form on the left side of the webpage
 - a. Create a check box to choose the type of categories you want to explore nearby.
 - b. Create a slider box to select the total number of the nearby venues you want to see. (using the HTML5 range tag, from 0 to 50)
 - c. Create a slider box to select the radius (in meters) around the point of interest for which you want to see venues (using the HTML5 range tag, from 0 to 3000, with increment of 100)
 - d. Below each of the sliders, create 2 read-only text input showing the number selected in the sliders continuously.
 - e. Create a submit button
2. Show a map on the right side of the web page
 - a. Use Google Maps API to show your map (centred around Minneapolis, preferably)
 - b. User should be able to place a marker on the map. We will call this the query marker. This is the query location, which is our point of interest.

- c. Once user places a query marker, and has chosen categories and other items on the form, user should be able to submit the form. Ensure that form cannot be submitted unless a query marker has been placed.
- d. Once form is submitted, you should post a request to the foursquare API as shown in **References I** section
- e. The API will return a JSON result with the list of venues
- f. Each venue is placed as a marker at the correct location (**retrieved from the JSON result**), the name of the venue should be the title of the marker
- g. The icon of the venue should be presented based on their category (**you can get the icon URL information from the JSON result**)
- h. When the user clicks on the venue, it should pop up an info window with its name in bold at the top. In addition, display the address of the location (**if available from the JSON result**) and lat, long coordinates of the venue (as shown in screenshot)
- i. Make sure that the query marker is still visible after the results come back
- j. Make sure your selections on the form are also “saved” after the results come back (see **screenshot 2**)

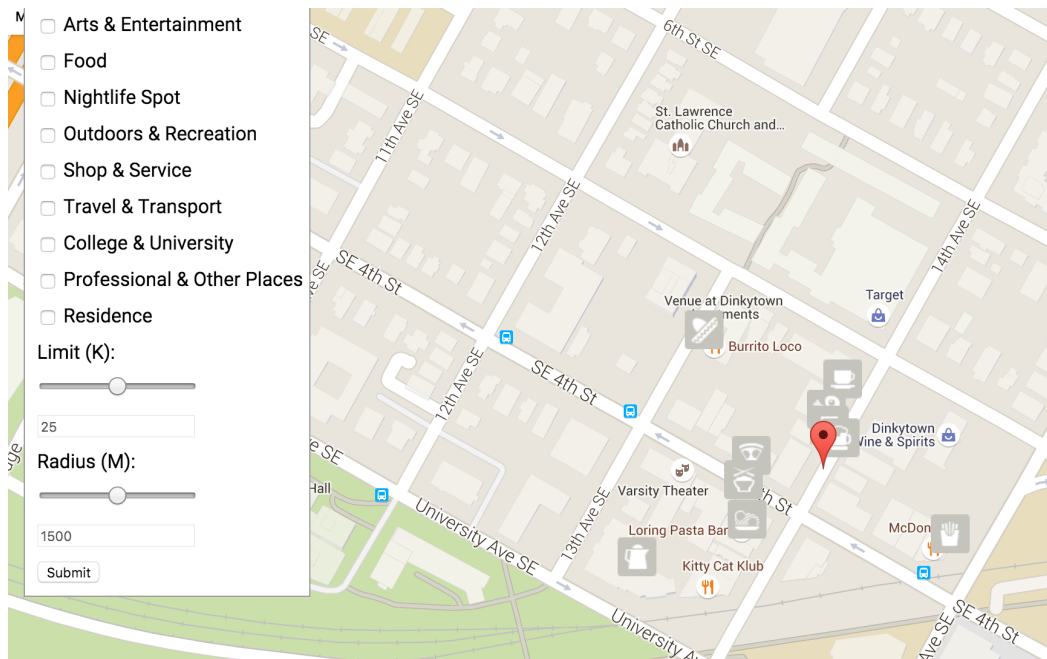
Supplemental information for the checkbox:

Category Name	CategoryID in foursquare
Arts & Entertainment	4d4b7104d754a06370d81259
Food	4d4b7105d754a06374d81259
Nightlife Spot	4d4b7105d754a06376d81259
Outdoors & Recreation	4d4b7105d754a06377d81259
Shop & Service	4d4b7105d754a06378d81259
Travel & Transport	4d4b7105d754a06379d81259
College & Universities	4d4b7105d754a06372d81259
Professional & Other places	4d4b7105d754a06375d81259
Residence	4e67e38e036454776db1fb3a

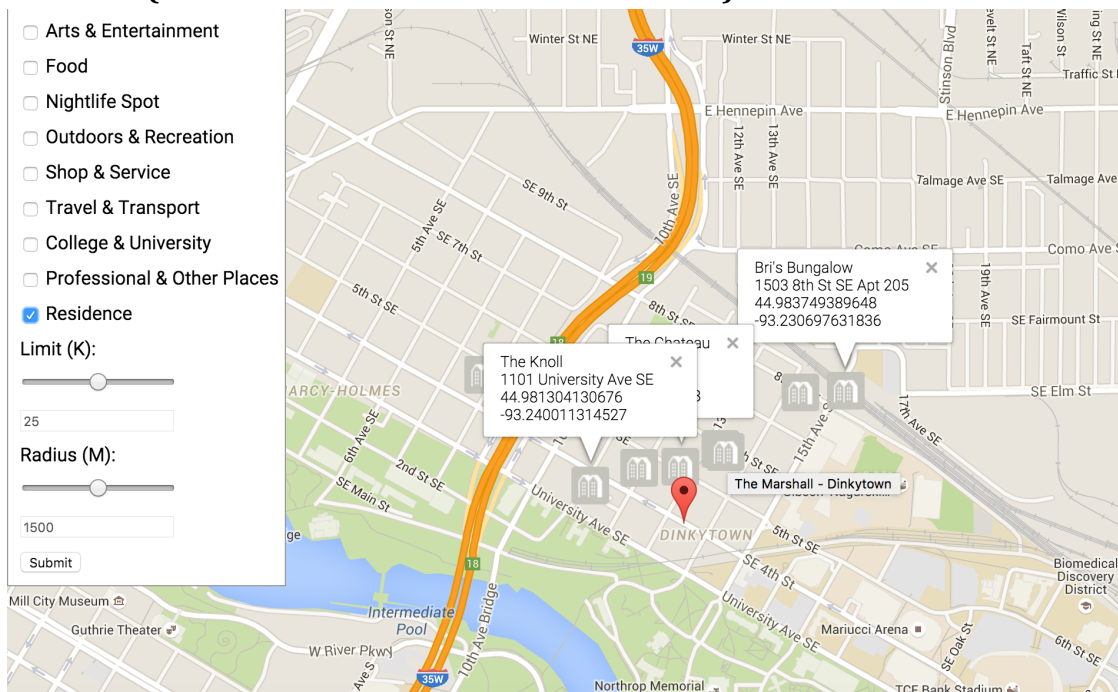
Part II: Screenshots

Screenshot 1: (With returned venues and markers)

Note: Not posting any category IDs to the API will bring back results of all categories



Screenshot 2: (With info windows of returned venues)



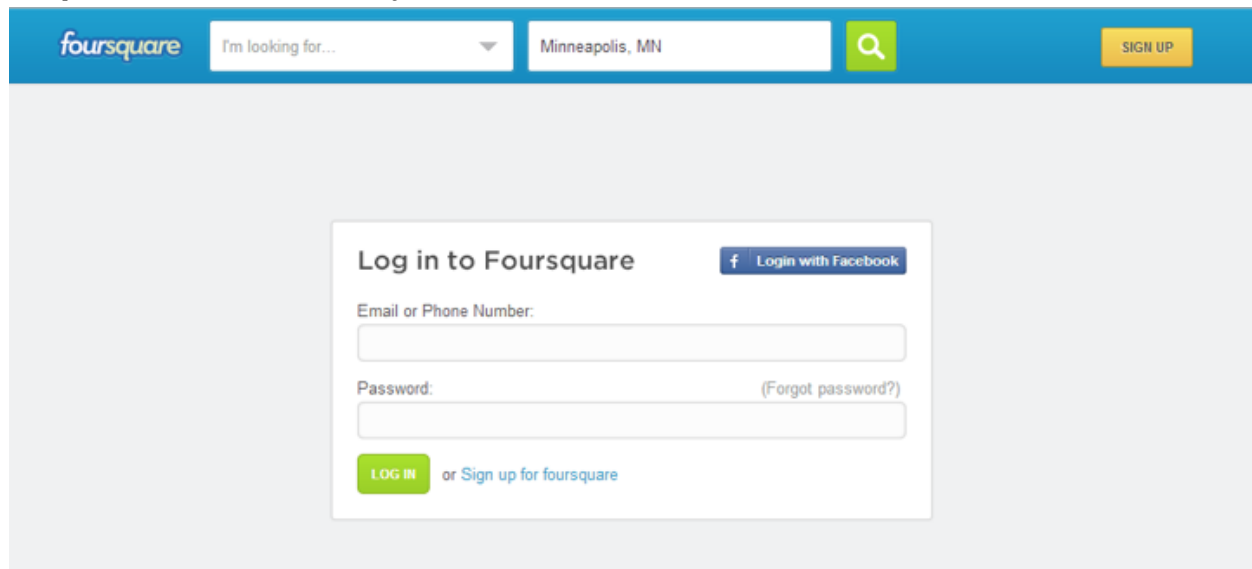
Part III: Grading Criteria

1. The page contains required form components (20%)
 - a. Checkbox (5%)
 - b. Sliders (10%)
 - c. Read-only text boxes shows the slider values (5%)
2. Map component (15%)
 - a. Map shows up on the page (5%)
 - b. User is able to place query marker on the map (10%)
3. Correct results are returned (45%)
 - a. Venue marker (10%)
 - b. Proper venue icon (10%)
 - c. Information window (10%)
 - d. Results limited by form choices- checkboxes, limit, radius (10%)
 - e. Venues centred around query location (5%)
4. Form content saved (20%)
 - a. Query marker is saved when results come back (10%)
 - b. Checkboxes, Sliders, Text Boxes are saved when results come back (10%)

References I: Foursquare Basics

1. Register an account in Foursquare (<https://foursquare.com>) or sign in using your Google or Facebook account
2. Go to the developer's page of foursquare :
<https://developer.foursquare.com/overview/>

For the people who are interested in the foursquare APIs, I recommend you to read into the details of this document. However, for this particular assignment, we will only focus on one API point to search the nearby venues.



3. Search Venue is the main API we are working with in this assignment, the detailed documents can be find here: <https://developer.foursquare.com/docs/venues/search>
Scroll to the end of this page


providerId	nymag	Identifier for a known third party that is part of our map of venues to URLs, used in conjunction with <code>linkedId</code> .
linkedId	4247	1002207971611 Identifier used by third party specified in <code>providerId</code> , which we will attempt to match against our map of venues to URLs.

Response fields

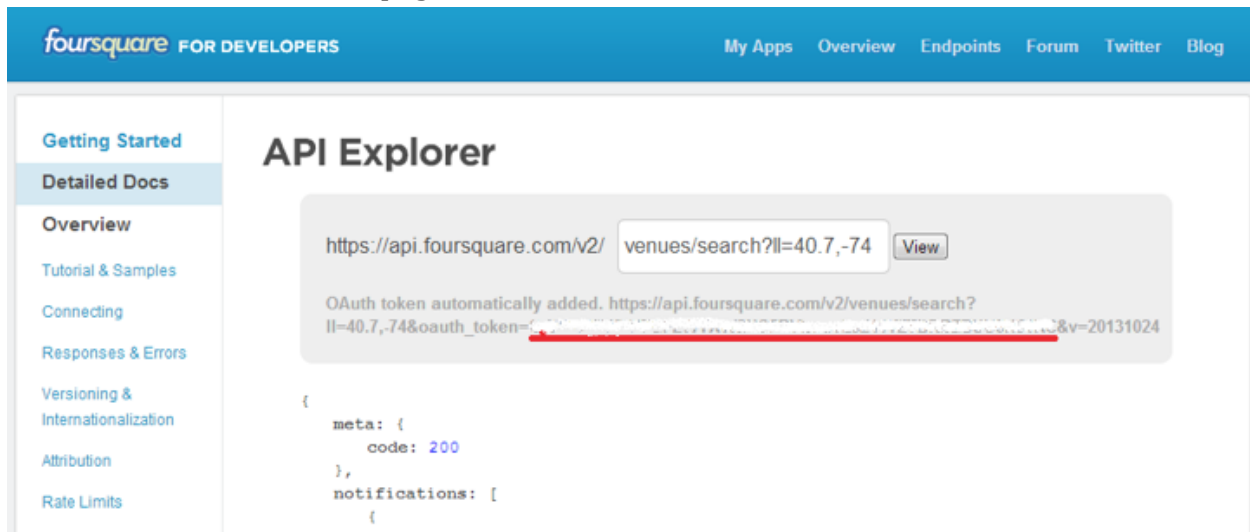
venues	An array of <code>compact venues</code> .
--------	---

TRY IT OUT

Home Overview Docs Resources Attribution Policies

POWERED BY  FOURSQUARE

Click on “try it out” button at the end of this page.
You should be able to see a page like:



The part under red lines is the auth token for your account, keep it. It will be used later.

4. This same page also gives an example of the JSON results from foursquare. You can get a flavor of the result from here.

Each returned venue is a JSON object including all the following information. In this assignment, we only need to use the ones under the red lines.

```

venues: [
  {
    id: "42377700f964a52024201fe3",
    name: "Brooklyn Heights Promenade",
    contact: { },
    location: {
      address: "Columbia Heights",
      crossStreet: "btwn Remsen & Orange",
      lat: 40.69829137715981,
      lng: -73.99663209915161,
      distance: 342,
      postalCode: "11201",
      cc: "US",
      city: "Brooklyn",
      state: "NY",
      country: "United States"
    },
    categories: [
      {
        id: "4bf58dd8d48988d163941735",
        name: "Park",
        pluralName: "Parks",
        shortName: "Park",
        icon: {
          prefix:
            "https://ss1.4sqi.net/img/categories_v2/parks_outdoors/park_",
          suffix: ".png"
        },
        primary: true
      }
    ],
    verified: false,
    stats: {
      checkinsCount: 16295,
      usersCount: 7631,
      tipCount: 104
    },
    specials: {
      count: 0,
      items: [ ]
    },
    hereNow: {
      count: 1,
      groups: [
        {
          type: "others",
          name: "Other people here",
          count: 1,
          items: [ ]
        }
      ]
    },
    referralId: "v-1382667403"
  },

```

To get the url for the venue icon: Prefix+'bg_32'+Suffix

For example:

https://ss1.4sqi.net/img/categories_v2/parks_outdoors/park_bg_32.png

5. Some information about the input parameters

Going back to the search API document

<https://developer.foursquare.com/docs/venues/search>

The things we care about are the following fields:

a) ll (latitude/longitude)

Parameters

All parameters are optional, unless otherwise indicated.

ll	44.3,37.2	required unless near is provided. Latitude and longitude of the user's location. (Required for query searches). Optional if using intent=global
----	-----------	--

b) limit

limit	10	Number of results to return, up to 50.
-------	----	--

c) category ID

categoryId	asad13242l,btbe24353m	A comma separated list of categories to limit results to. If you specify categoryId specifying a radius may improve results. If specifying a top-level category, all sub-categories will also match the query. Does not apply to match intent requests.
------------	-----------------------	--

d) Radius

radius	800	Limit results to venues within this many meters of the specified location. Defaults to a city-wide area. Only valid for requests with intent=browse , or requests with intent=checkin and categoryId or query . Does not apply to match intent requests. The maximum supported radius is currently 100,000 meters.
--------	-----	---

You can use the category id information we provided in the supplemental information section. Remember, if you want to submit multiple categoryIDs, they should be separated by commas.

Finally, you will post a request to foursquare something like:

https://api.foursquare.com/v2/venues/search?ll=latitude,longitude&intent=browse&oauth_token=YOURTOKEN&limit=K&radius=R&categoryId=SELECTEDCATEGORIES

The red parts should be filled by your php program.

References II: JSON

1. Basic json information

<http://www.w3schools.com/json/>

2. How to decode json files in php

<http://www.php.net/manual/en/function.json-decode.php>

3. An online json parser

<http://json.parser.online.fr/>

References III: PHP

1. Get content from web

<http://php.net/manual/en/function.file-get-contents.php>

2. Pass parameters via url

<http://php.net/manual/en/reserved.variables.get.php>

References IV: JavaScript

1. Get values from html form

<http://stackoverflow.com/questions/6069146/to-get-value-of-textbox-in-javascript>

2. Submit button in JavaScript

http://www.w3schools.com/jsref/met_form_submit.asp

Submission Instructions:

You should submit a single .zip/tar.gz file on moodle.