

# **Homework 6: Server-Side Scripting using PHP, JSON and Ticketmaster API**

## **Grading Guidelines**

Graders will test your homework on Firefox and Chrome.

### **Grading Guidelines:**

#### **Search Form (2.5 points in total)**

- The initial webpage with proper fields in the search form, along with “Events Search” header. **(1 point)**
  - The drop-down list should contain all the 5 categories, 0.25 points will be deducted for each missing category
  - Location edit box must be disabled when choosing “Here” radio and enabled when selecting it, 0.5 points
  - Both Distance and Location edit boxes should have a placeholder, 0.25 points
- Validation for the search inputs, the error message(s) should indicate if fields are missing. **(0.5 point)**
  - If there is no error message for invalid input, 0.5 will be deducted.
- The “Clear” button resets the search form to the initial state and clears the results area. **(0.5 point)**
  - For each not correctly reset or cleared field, 0.25 point will be deducted, for a maximum deduction of 0.5 point.
- When the “Search” button is clicked, the page maintains the previous input values after displaying the results table. **(0.5 point)**
  - For each not correctly maintained field, 0.25 point will be deducted, for a maximum deduction of 0.5 point.

#### **Search Result Table (5.0 points)**

- Show the search information retrieved from the API service in a tabular format as described in the homework description. **(2.5 points)**
  - For each missing column in the table, 0.5 point will be deducted.
  - For each incorrect column value or column order mismatch, 0.5 point will be deducted.
  - For each missing row/category date/icon/event/genre/venue in the table, 0.5 point will be deducted.

- For each incorrect row/category date/icon/event/genre/venue, 0.5 point will be deducted.
- The search should work for all 5 possible categories. For every missing search, 0.5 point should be deducted.
- Maximum deduction will be 2.5 points.
- Handle the case when the address in table is clicked. It should show a map with a marker indicating the location of selected place. **(1 point)**
  - For each missing/incorrect marked map, 0.5 point will be deducted.
  - Maximum deduction will be 1 points.
- Handle the case when the “X there” (X indicates direction mode) button on the map is clicked, it should show a direction map from the searched location (either “here” or the location in edit box) to selected place. **(1 point)**
  - For each missing/incorrect origin/destination, 0.5 point will be deducted.
  - For each missing/incorrect direction mode, 0.5 point will be deducted.
  - Maximum deduction will be 1 points.
- Handle the case when the API service returns an empty result set. **(0.5 point)**

#### **Details Information (2.5 points)**

- Show the information retrieved from the API service in a format as described in the homework description. **(1 point)**
  - Show the name of selected place on top of reviews, 0.25 points
  - For each missing key value pair in both tables, 0.25 point will be deducted
  - For each missing photo in the table, 0.25 point will be deducted.
  - It should display Google map correctly, 0.25 points
  - Event information have a similar format as description described, 0.25 points
  - Venue information have a similar format as description described, 0.25 points
  - Maximum deduction will be 1 point.
- Handle the case when the API service returns an empty result for either venue info or photos or both **(1 point)**.
  - For each missing message for empty photos and empty info, 0.5 points will be deducted, with a maximum 1-point deduction.
- Show the venue info and photos subsections in a dynamic view **(0.5 point)**

#### **\*\*IMPORTANT\*\***

- You should not call any Ticketmaster and Google APIs directly from JavaScript, except for the Google JavaScript Library. Doing this will result in 4.0-point penalty.
- If JQuery is used, 4.0 points will be deducted.

- You should implement your hw6 code in a single PHP file, which includes the external file **geoHash.php** . If multiple files are submitted, 3.0 point will be deducted.
- You should implement hw6 using AWS/GCP/Azure cloud server as described in hw5. Otherwise, 3.0 points will be deducted.

**Source code submission:**

Students should submit their source electronically to Github Class account so that it can be graded by graders and compared to all other students' code via the MOSS code comparison tool. In addition, on your course homework page, you **must** update HW6 link to refer to your new hw6 PHP page.