Short report on the submission of Web Cartography

385782 Sanghun Kang

[Functionalities]

It basically shows locations of restaurants, POIs, and when they are marked as favourite, they are identified with different Icons.

Upon clicking ‘GO’s with selection of either my favourites or POI, the centre of the display moves to that selection. This is the way “showing restaurants around some area” is realised.

It is possible to filter restaurants by types of food they serve, ratings (you give) or both. For the simulation purpose, most of them are given 3.

You can add or drop a star with simple plus and minus sign which become visible when you select a restaurant on a map.

Also, you can add a restaurant to your favourites or remove one from the list. When a restaurant is in the list, you can access it via dropdown list as well.

[Databases]

Somehow, I found a website where I could download some lists of URLs where JSONs of features are available. I scraped them with my python code (which is also included in the submission) and modified a little bit to fit in the tasks of this assignment. (e.g. add an attribute to indicate whether a feature is selected as favourite).

[Structure]

I didn’t use much of PHP. The biggest concern came from JavaScript (and JQuery). I expected that it might require to implement lots of PHP code for data interchange, but it turned out most of them can be done on browsers, so sending and importing the geoJSON file upon request was enough.

The geoJSON file, which I name ‘db’ contains all the data required to draw the features. I preferred to work with RDBMS, but it seemed it was not allowed to.

For importing CSS and JavaScript libraries, I used globally accessible sources, i.e. web sources. Partly to downsize my submission, partly to be robust to different situations as I’m submitting not from Berlin.

Core functions are realised by a single JavaScript script. Before document standby, variables regarding maps are defined. Functionalities, which are mostly on-click triggers, are defined after document standby. Referencing elements are mostly done by IDs of elements.

If an action is related to updating the database – e.g. adding a star to some restaurant – then the whole page will be refreshed. If some an action deals with just display, only the style is modified without page reload.

[etc.]

Some tried, but not finished features are included as (rather long) commented part. I hope I could finish them sometime later.

[References]

<http://getbootstrap.com/examples/cover/>

<http://getbootstrap.com/examples/dashboard/>

<https://silviomoreto.github.io/bootstrap-select/>

<http://stackoverflow.com/questions/7176908/how-to-get-index-of-object-by-its-property-in-javascript>

<http://openlayers.org/en/latest/examples/popup.html?q=popup>

<http://www.iconsdb.com/white-icons/restaurant-icon.html>