CMSI 370-01

INTERACTION DESIGN

Fall 2014

Assignment 1204

OK, back to the code gang. This time themed around the direct manipulation interaction style, of course.

Outcomes

This assignment will affect your proficiency measures for outcomes *2b*, *3a*, *3b*, and *4a–4f*.

Background Reading

Textbook reading is centered on the direct manipulation interaction style, which would be Shneiderman/Plaisant Chapter 5.

For the programming assignments, the most helpful additional material outside of the web and the *bazaar* code will be the case studies in Chapter 9 of the JavaScript textbook. These case studies demonstrate lower-level event handling with some direct manipulation elements.

For Submission

Direct Manipulation Exercise

This exercise gives you some firsthand experience in implementing direct manipulation. To do this, you will need ready access to a device with a multitouch web browser. If you have your own device, then great; if not, we can arrange for you to check one out from the Keck lab.

Modify the *boxes-touch* bazaar code so that it supports *creation* and *deletion*, allowing for more than one action at a time (one per finger). You should, of course, preserve the box-moving functionality that is already present in the web page. For deletion, provide visual feedback that a box will be deleted when the user's finger is lifted.

How to Turn it In

Commit a *copy* of your modified code under *direct-manipulation* in your private 370 GitHub repository (i.e., *don't* make the changes in place and issue a pull request), and upload your work to *my.cs.lmu.edu* so that it is available under the path *~<u>username</u>/cmsi370/direct-manipulation*.

For Submission

A User Interface Widget from Scratch

We end by going back to the basics: design and implement a reusable widget for use in web browsers in general, and for your role-playing game user interface in particular. To emphasize reusability, you will implement your widget as a *jQuery plug-in*.

The point here is to see how low-level event handling (e.g., mouse/keyboard activity) translate into higher-level ones (e.g., selection or change events). If the first programming assignment involved direct manipulation "in the large," this one exercises direct manipulation "in the small." Some ideas:

- A selection knob or slider
- A rolling or scrolling item selector
- An entry field that accepts text/numbers with drag-and-drop character tiles
- A "here-to-there" drag-and-drop area
- A directional pad ("d-pad") control

You may use jQuery but Bootstrap use may be *CSS only*—no Bootstrap JavaScript components allowed, whether in code or triggered by *data* attributes. If you have a widget idea that is not in this list, check with me to see if it will work.

How to Turn it In

Commit your code in two places. Under *widget-from-scratch/*, provide these distinct pieces:

1. The reusable code for the widget itself (typically CSS and JavaScript)
2. A "demonstration page" that shows a standalone instance of your widget in action
3. "Eat your own dog food:" Under *rpg/*, integrate your widget into the user interface that you have already built.

Finally, as before, upload your work to *~<u>username</u>/cmsi370/widget-from-scratch* and *~<u>username</u>/cmsi370/rpg*, respectively, on *my.cs.lmu.edu*.