CMSI 370-01

INTERACTION DESIGN

Fall 2012

Assignment 1120 Feedback

Carlos Agudo

1e — It looks like you did not have enough time to act on some of the feedback I gave you by email. In terms of affordance choices, the single resize handle is appropriate, although it does not look like a typical resize handle (see: almost any drawing program out there). It remains recognizable, but it would not have taken that much more effort to really make it look like the standard version.

In addition, I also mentioned the use of `mousemove` for deletion. You added the confirmation dialog, but this just reinforces the disruptive aspect of this choice. As stated, the preferred approach is to: (a) provide visible feedback while dragging the a box will be deleted when the mouse is released, then (b) deleting the box when released outside the drawing area. Note that this approach obviates the need for the confirmation dialog, which suffers from the additional disadvantage of somewhat "breaking" the direct manipulation interaction style that is established here. (/)

3a — You show some good advancement in user interface construction here. (+)

3b — Affordance choices and resize glitch (see *3d*) aside, your event-driven programming knowledge also shows quite well with your code. (+)

3c — You demonstrate MVC separation very well here. (+)

3d — Your processing of low-level system events is near perfect, except for how you are tracking the anchor coordinates upon resize. You are actually quite close—I fixed the issue in very few keystrokes—but it is important that you find the problem yourself. It is all about being clear on what element is receiving (and thus processing) a particular event. Be *absolutely sure* that you are clear on what element is processing an event at any given time (i.e., the `this` in the code). The resize anchor bug exists because there is some confusion on that in a couple of the event handler functions. (|)

4a — Outside of the event handling/resize issue, your code has a number of additional points of improvement. See the inline comments for details. Overall, things are generally functional and correct, but display a few bad habits that you should really not be showing anymore as a junior-level computer science major. This detracts from the final quality of your code. (|)

4b — Your code demonstrates proper separation of concerns. (+)

4c — Your code readability is OK but not great. You have spacing and indentation glitches and your comments are somewhat terse; see my inline feedback for some specifics. (|)

4d — You did a pretty good job finding necessary information, including asking me questions. (+)

4e — Commit phasing and timing is very good, with good messages. (+)

4f — Submitted on time, with further improvements after that also. (+)