Assignment 1117

Technology shift! Time to transfer your UI definition and event handling skills toward web pages.

Outcomes

Assignment 1117 will affect your proficiency measures for outcomes 1a, 1b, 1g, 1h, 3a, 3b, 3c, 3d, 3e, 3h, 3i, 3j, 3k, and 3l.

Not for Submission

By Now

Read Chapter 6 in Shneiderman/Plaisant (I’ve name-dropped this chapter a few times over the past few weeks so this should be no surprise, and you may have probably already read it). For older editions, this was previously Chapter 7.

For Submission

Make Change: the Web App

As a warm-up, code up a web version of the now-classic, iconic Make Change program. Be creative, be experimental, stay usable :) Link the web app to a short commentary web page comparing Cocoa/Objective-C and HTML/CSS/JavaScript. Identify:

- Common concepts and activities (independent of language or library)
- Notable differences (in any aspect that resonates with you)
- Anything else that comes to mind when comparing the two

Remember that, past a certain point, one should separate the issue of learning the technology from actually using it with a level of proficiency already attained. This is analogous to how the learnability and memorability metrics are separated from efficiency, errors, and satisfaction.

Interaction Design “Quiz Page”

Show off both your interaction design knowledge and your web app savvy by implementing an interactive “quiz page.” Ask 10 questions about the field and have the user answer them interactively. You should have at least one of the following types of questions:

- Multiple choice (radio button)
- True or false (check box)
- Simple fill-in-the-blank (where the answer can be easily parsed)

When the user completes the “quiz,” the page should show the correct answers and tally the user’s score.

Choose your questions so that they demonstrate knowledge from outcomes 1a, 1b, and 1g in general, and from 1h for the menus, forms, and dialogs interaction style.

Web Components from First Principles

Implement, from first principles, web versions of the following user interface components:

- A push button
- A labeled check box

Each web component must behave as similarly as possible to their desktop equivalents. Notably:

- Your web button must allow the user to back out of clicking it prior to letting go of the mouse button, and provide feedback to that effect.
- The check box’s clickable area must include its text label (yes, you are implementing, from first principles, the behavior of the label element).

Structure your code as modularly as possible, so that you can use it on almost any element on a page. Package a demo page that shows your handmade buttons and labeled check boxes.

How to Turn It In

Upload your files to your my.cs.lmu.edu web site (go to secure.cs.lmu.edu if you don’t have one yet):

- make-change.html
- make-change-commentary.html
- ixd-quiz.html
- web-component-demo.html

Please supply hardcopy of your source code too.