

# CMSI 370

## INTERACTION DESIGN

Fall 2008

### Assignment I030a

Alright, time to work on an assignment that gives you a taste of interaction design research, plus work on a Swing program with a full-fledged graphical user interface.

### Not for Submission

#### By October 21

1. Read Chapter 7 from Shneiderman/Plaisant.
2. Read the menu research papers that were given out in class.

#### By October 23

1. Check out *menus* from your respective *homework/cmsi370* CVS depots and get started on the *Menu Selection Experiment* portion.
2. Start the Swing part of the assignment and have questions ready for whatever isn't clear.

### For Submission

You should each find a *menus* directory on your respective *homework/cmsi370* CVS depots. We'll use this *menus* code in two ways for this assignment.

#### Menu Selection Experiment

Use *menus* to conduct an efficiency and error rate experiment among three menu styles: pull-down menus from an arbitrary location, pull-down menus from the top of the screen, and pop-up menus at or near the current mouse location.

Perform a series of menu selection tests using *menus*, gather the resulting data, and draw a conclusion about the relative efficiency and error rates of the three types of menus. A sample menu setup is included, but do feel free to change that menu to your liking (or even to test different menu fixtures).

Work on this as a group, and involve as many people as are willing to participate. Commit the data and your analysis of the data to CVS.

**Implementation Note** As written, pull-down menus from the top of the screen can only be achieved by running *menus* on Mac OS X, with the *apple.laf.useScreenMenuBar* system property set to true (i.e., *java -Dapple.laf.useScreenMenuBar=true ...*) or any other property-setting mechanism.

### User Interface Improvements

Implement the following improvements to the *menus* user interface. You may work on this in groups of any size, but *make sure to commit the new code in each of your CVS repositories*.

- Convert the session collection display within the subject view from a JList to a JTable
- Implement export to a text file, including a dialog box that appears to request the export file's name and location
- Add support for multiple test subjects: show a subject list first, allowing the user to add or remove subjects; a user can then "open" or select a subject to show that subject's view
- Make sure that Open Session only produces one session data window (i.e., if a session data view is already open for that session, just bring that window to the front)
- Implement a double-click, such that double-clicking on a session does the same thing as selecting that session and invoking Open Session

### How to Turn it In

Commit the results of your menu selection experiment as LaTeX source on one of your CVS repositories (tell me which one), under *homework/cmsi370/menu-report*.

Commit your improved *menus* code to your respective *menus* directories in CVS.

### Extra Credit

You will get extra credit if your new version of *menus* includes *customizable menu fixtures* — a mechanism that allows the menu fixture (i.e., the set of menus used for testing) to be customized, such as through an external file (1 extra credit) or a built-in menu fixture editor (2 extra credits).