

HCI-CM0233 Coursework Description

1. Administrative Details

Scheme	BSc in Computer Science		
Module	CM0233	Human Computer Interaction	
Coursework Title	Simple Web Browser in Java Swing		
Lecturer/s	Dr. Dave Marshall and Dr Alia Abdelmoty		
Date Set	Monday 09-02-04 (week 3)	Submission Date	Wednesday 5 th of May 2004 (week 12)
Submission Instructions	You MUST get your program listing signed by the lab tutor (in a regular lab session- Week 11) in order to prove that it works. Please use the coursework submission cover sheet for this purpose. The Tutor will only sign programs during tutorial times. Unsigned programs will get 0 marks.		
Work Will Be Marked out of	100	Contribution to subject mark (%)	30%
This work should take about	25-30 Hours		

No marks will be awarded to work submitted after the above submission date.

2. Learning Outcomes

Learning Outcomes
<ol style="list-style-type: none">1. To test your understanding of HCI design principles2. To give practical experience of interface development and prototyping3. To give practical experience of evaluating user interfaces

3. Detailed Requirements

This assignment is made up of two complementary parts.

Part 1

Implement a very simple web browser in Java Swing. The browser should provide simple functionality in the following areas:

- Web link bookmark/favourites and filter application,
- Facilities for opening and saving web pages locally,
- Facilities for allowing the browsing of several web pages within tabbed panes,
- Access to main functionality of the program through a variety of interface component, e.g. menu-bars, toolbars and popup menus.

Your application should be well organised with a sound graphical user interface. A fully working professional level Web browser/link editor application is not expected but you are expected to develop an application with the following minimum requirements:

- You should provide a basic mechanism for viewing HTML pages in Java Swing application.
- You should be able to monitor any web link that is selected from the HTML page display.
- You should be able to log recently visited web pages and view them in an appropriate manner in the application.
- You should be able to store selected web links (and recall the web pages over the Web) as bookmarks and view these in an appropriate manner.
- You should be able to set up a set of web link filters that prohibit the user to access the pages marked (and stored) as filter pages in your application. Your application should actively filter web links and suitably alert the user to this fact.
- You should be able to view and edit in a suitable fashion favourites and filters.
- You should be able to open web pages including local files from appropriate interface components.
- You should be able to save accessed web pages to local disk.
- You should provide a means of opening multiple web pages, each of which should be displayed in a separate tabbed pane.

- You should provide access to commonly used functions via a variety of appropriate interface components.

How you actually organise the application and editing/viewing of Web links etc. is up to you. BUT the application should adhere to standard HCI/GUI guidelines.

Part 2

- 1) Produce a listing justifying your choice of all Java Swing components used in building your interface.
- 2) Produce a textual HTA description of the required system. Present the same information in a diagrammatic form.
- 3) Use two of the tasks required above to demonstrate how your design and implementation have adhered to the three usability principles of a) learnability, b) flexibility, and c) robustness.
- 4) Illustrate how you have incorporated at least one evaluation technique in both the design stage and the implementation stage of your development.

For example, you could use a model-based evaluation technique to evaluate your design, by producing a GOMS and KLM analysis for some tasks and use those to predict user performance on your system. To evaluate your implementation, you may choose to design a questionnaire, conduct an interview or do an observational evaluation using fellow student(s) as test subjects.

You may also choose to evaluate your system against other similar systems using a suitable evaluation technique.

4. Deliverables

- 1) Your answer to Part 2 above should be in the form of a report of between about 3000 and 4000 words in length.
- 2) Your report should also include a complete code listing as an appendix.
- 3) The coursework submission sheet suitably signed by your Tutor to verify that your program works

5. Marking Scheme and Assessment Criteria

The coursework mark is divided equally between both parts 1 and 2. I.e. Part 1 and 2 each carry 50% of the coursework marks.

Part 1:

Average marks (up to 69%) will be obtainable if you adhere to the following application specification, demonstrate it working and document it properly. Additional marks will be awarded for additional functionality that extends the application usability and fits in with good GUI design philosophy.

Part 2:

Credit in the report will be given for:

- clarity of exposition and logical organisation of the report;
- depth of understanding and analysis of the problem;
- ability to synthesise information and highlight the important issues;
- critical evaluation of the described methods;
- evidence of consulting a variety of sources.