
CS 456, Spring 2009
Assignment #2
Putting your behind past us [Pumba]

Due 2/6/09, in class

Assignment

Your goal in this assignment is to put eBay out of business! How? By constructing a web page that supports the auctioning off of items, of course. Like many web applications there will be a front-end and an back-end. The front-end provides the following functionality to a user:

- Create an item (*i.e.*, accept a name, a brief description, the initial cost, and a closing time).
- Bid on an item (*i.e.*, accept a name, selected item, and a bid, which must be larger than the current bid or an error page is returned).
- check for stuff I won (*i.e.*, accept a name).

The back-end will maintain a data-base of current and past auctions. It this must accept data associated with the three front-end requests and process it. This includes the following:

- Add - an item to the “date base”, which for now can be as simple as a text file with one time per line.
- Bid - update the bid for an item. Respond with a page showing the updated item status or an error.
- Check - compare a name and current time with the database an report all won items.

All back-end CGI code must produce nicely formatted web pages reporting the results and then allow further navigation.

Finally, to give you some practice with cascading style sheets, you must use one to give your pages a common look-and-feel.

After construction, validate each of the pages you produce and a sample of the response pages generated from your back-end. Save, but to not print, a dump of each successful validation.

What to hand in

- (1) A 2-up (ask me if need be or see man a2ps (1)) copy of your web pages
[as an encouragement non-2-up printouts loose 5 points.]
- (2) Email me the source for both the front-end and back-end and a sequence of screen shots showing off how well your system works. Note that “showing off” means testing. Put some thought into testing.

Notes

- (1) Creativity a plus.
- (2) For this assignment you will work in pairs. One person in the pair must either have survived CS 451 or know perl.
- (3) All static and dynamic (generated) web pages must be valid xhtml.