

Software Engineering Homework #10 – due April 29, 2008

Instructions

The following is a testing report template. Each team must send a copy of this completed document to barrett@cs.nyu.edu by midnight on the due date. The subject line of the email message must contain only the word “hw10”.

Your group has been assigned to perform testing for one of the other projects (the same team for which you reviewed the design documents). The purpose of this assignment is twofold:

- To give you the experience of being a tester on code you do not fully understand and cannot change. This can be a frustrating and challenging job, and not enough software developers appreciate it.
- To give concrete feedback to the group you are testing for, at a time when it may still have some effect on the final version of their project.

You will be testing the same project for which you did the design review. It is up to you to contact the group you are testing for and arrange to get access to their system. I do not expect you to do more than manual testing. The group developing the project is responsible for doing any automated testing of their project. Your role is to be a proxy for an actual user. You should use the system and report any problems; your grade for this assignment will be based in part on how thoroughly you managed to manually test the project.

You must also give access to your code to the group who is doing the testing for you. You must provide complete instructions on how to download, install, and run your code on the CIMS machines. These instructions should be sent to the other team AND to me at barrett@cs.nyu.edu. Teams that do not provide access to their code in a timely manner or that provide inadequate instructions will be penalized.

Your Project Name
Testing for *Name of Project You are Testing*
Date
Version *major.minor*

1 Summary

Here you should give a summary of your findings. The most serious problems you discovered, or the fact that there were none, goes here.

2 Findings

Here you should give the detailed list of your findings. A “finding” can be a bug, something that is unclear, inconsistencies, features that appear to be unimplemented, or any other aspect of the system that affects usability for the purpose for which it is being built. For example, you may wish to comment on installation, the documentation (including on-line “help”), the user interface, performance, (mis)behavior on various kinds of input or user actions, scalability, etc. This list of areas is a guide; you may add or change the areas if they seem more appropriate for the project you are testing.

Each finding should contain three things: a description of the problem, the severity level, and any suggestions or comments. Use the three severity levels: severe (fatal problem), moderate (much better if this is fixed than not in the current version), minor (worth fixing eventually, but probably not now). Suggestions are for conveying information beyond the description that may not be immediately apparent to someone reading the document. For example, if you suspect that a particular problem is the result of a race condition, you would put that hypothesis in the suggestions section of the finding. Your suggestions can be empty and should not restate the obvious (e.g., if you report that a feature is unimplemented, please don't bother to suggest “implement the feature”).