

CS 443/576, Spring 2014

Project 4 — High fidelity prototyping

Due date: The **beginning** of class on Tuesday, April 29.

Design crits: Tuesday, April 29 and Thursday, May 1.

Project Goals

- To create a high-fidelity prototype of your system design
- To create scenarios of use and task lists that will be used in Project 5

Project Description

Prototype Programming

After incorporating design changes that resulted from feedback received and evaluation results thus far, create a high-fidelity prototype implementation of your system. You are free to use any available method/programming environment to develop your system. Keep in mind that your overall goal is to implement your design as an *interactive* system that users can test with in Project 5. This means that your system should *appear* to the users as though it is completely functional, but it need not necessarily be so. For example, if you have an initial Login function in your design, when a user “logs in” during testing, your system should appear as though it’s checking the username/password information against an underlying database, but it doesn’t actually have to do this. On the other hand, your system must be implemented to a sufficient level of detail to provide a robust environment for thorough user testing. This means that users should be able to complete tasks that relate to at least all of the high priority functional requirements that you designed for, and hopefully some of the medium priority ones as well.

For this project, each team member should contribute equally to the prototype implementation. Create a master list that shows which team member was responsible for implementing each of the functional requirements included in the system. Make sure you plan your time accordingly such that you will have a chance to merge all programming together to create one seamless user interface. The final implementation should not appear as though it was programmed by separate team members, but rather should be cohesive, consistent, and well-integrated.

Please note: Make sure that all data asked for and maintained by your prototype is fictitious. Do not store any personal information about any real people in your system.

Scenarios of Use / Task lists

In addition to creating your high-fidelity prototype, each team should also create a number of scenarios of use and task lists for your system (I would hope to see at least five scenarios of use, each containing numerous tasks/functions). These are what you will give to your users during user testing in Project 5. Your scenarios/task lists should include the high-level representative tasks that your users should be able to perform using your system, accompanied by the lower level tasks that make up each scenario. These tasks are critical to the success of your user testing, so think them through carefully. Be sure to make them clear and realistic, and try to incorporate as much of your system’s functionality in them as possible. You’ll need to communicate to the users *what* their tasks/goals are without telling them *how* to complete the tasks (as this is what you’ll be testing). We will discuss how to create scenarios and task lists in class. An example follows.

Example Scenario of Use / Task list from previous class assignment:

You have been making great progress on your geology group project, and decide to celebrate by taking the afternoon of next Friday off. Of course, you can't have a party by yourself, so you intend to invite your group members.

Tasks:

- Login to CoLab using the following credentials:
 - user name: geologyrocks@domain.com
 - password: mypassword
- Choose the Medium screen size
- Choose the Grid Layout
- Edit the "Group Lunch" event scheduled for April 17
 - The event starts at one in the afternoon and doesn't have a specified length of time
 - Call the event "Good job party" and describe the house, bring chips!"
- Tell Bob and Alice about the party
- Wait around for a few seconds to see if anyone replies
 - *At this point, Bob will reply, "sounds like fun! While you're here, you should take a look at the changes I made to file3.doc."*
 - Download the suggested file to the desktop
- Tell Bob that you are busy and will examine the file later
- End your CoLab session

Design Crit

For the design crit sessions to be held during class on April 29th and May 1st, your team should prepare a 20 minute presentation to showcase your prototype. Since we have already seen your initial designs during previous projects, focus your presentation on the functionality you chose to implement from your design and any new and interesting features/functions you may have added. Discuss any implementation difficulties you may have had and how you chose to resolve them. Additionally, for your design crit, plan to have a classmate from the audience test one or two of your prepared scenarios of use and corresponding task lists; we will see if we can figure out how to accomplish those tasks together as a group during your crit session without any input from you. This will be interesting! The total amount of time teams will be allocated for design crits is as follows:

- 3 person team → 25 minutes
- 2 person team → 20 minutes
- 1 person team → 15 minutes

Team Deliverables

On the due date your **team** should hand-in a report describing:

1. How you implemented your system (i.e., what software did you use?). (5 pts)
2. Your current list of functional requirements, prioritized. This list should clearly indicate which functions you implemented, and who implemented each requirement. (10 pts)
3. Scenarios of use and their accompanying task lists. (15 pts)

I also need a copy of your working prototype, and a way for me to interact with it for grading purposes. In other words, you need to make your system accessible to me. Be sure to include instructions as to how I can access and interact with your prototype (including any required login IDs and/or passwords). (65 pts)

Individual Deliverables

1. A peer-review for each member of your team for this project. These forms can be downloaded from Moodle. (5 pts)