Computer Science Capstone Design

Assignment: Acceptance Demo

(50 pts)

Overview

The goal of any project, ultimately, is to produce a piece of software that satisfies the requirements and specifications that were developed for it in the early stages of project analysis and design. More practically, many real software projects have to pass a final "acceptance test" with the client, in which the software is extensively tested by the client's organization for functional performance, usability, and overall acceptability. The practical and legal basis for acceptance testing is the requirements document specifying the agreed-upon functional and performance requirements for the product. Your requirements may have evolved somewhat during the implementation phase (which, in real life, would have been documented carefully), and it is this final version of the requirements that you'll be measuring yourself against.

The Assignment:

What you will do in this closing assignment is to work with your mentor develop a "Final Demonstration Plan", which is clearly driven by the functional and performance requirements that were developed for this project.   This jointly-developed "demonstration plan" then becomes the agenda for an in-person demo with your project mentor, where you walk through and demonstrate all of the functionality promised. The agenda is just a minimum outline of what is planned; your mentor, may, of course, ask for additional or clarifying demonstrations along the way.

Assignment specifics: What to do

Obviously, the product and its key features will be different for each project. Thus, you will work with your mentor to develop a final demo testing plan; this plans and organizes your demonstration clearly, and ensures that you include everything that your mentor would like to see demonstrated.

Step 1: Develop a proposed test plan. There is a basic template (MSword) for developing a "Final Product Demo Plan" linked from the course website. You will download this template, then flesh it out, filling in the sections detailed in the template. You aim is essentially to:

1. review all of the requirements that you promised, i.e., taken directly out of your Requirements Document that was produced in CS476; and

2. propose a set of "demonstration sequences" that hit all of the requirements. These are essentially step-by-step demo walk-throughs of some particular part of your product aimed at showing clearly how your product satisfies each specific requirement promised.

Step 2: Review and refine your proposed test plan with your mentor. **This step is absolutely required!** You mentor must explicitly "sign off" on the adequacy of your proposed test plan, either with a clear email, or by physically signing a hardcopy of the final draft; keep

this email/signed draft as documentation. Your mentor may ask you to add/upgrade certain missing/weak aspects of your demo plan, so be sure to start on this early to allow time for revision cycles.

Step 3: Arrange a time. When the final test plan has been agreed upon, arrange a time with your mentor for the demonstration. Also upload your final test plan to BBlearn.

Step 4: Do the demonstration. Arrive at the demo ready to show off your product, and **with a hardcopy version of your test plan ready to hand to your mentor**. Your mentor will then write evaluative comments directly on your test plan hardcopy while scoring your demo; it becomes the "grading sheet" for your demo.

Deliverables

- A detailed testing plan, approved in advance by your mentor, and brought along in hardcopy to the demo session for your mentor.

- A live in-person demonstration of your software demos for your CS Faculty mentor, following the established Demonstration Plan.