CS 320
β Release

Due: **Thursday, April 18, 2024, 12:00PM noon EDT** via [Moodle](#). For this assignment, you must work in your project group. Submit only one assignment per group. You will also give an up-to 10-minute group presentations on Thursday, April 11.

Overview

CSRocks Inc. is excited to see the progress you're making on the product. This assignment is a milestone delivery. While the β isn't a feature-full product, CSRocks Inc. does want to see that all the major pieces of the product are in place and integrated at a very basic level. CSRocks Inc. would also like to see that your team is using good software engineering practices for your work, ensuring that the project will be completed on time, with high quality, and in a good position to evolve with customer needs.

As you are working on your β release (but a week before it is due), you will present it to your customers and peers.

Deliverables

1. A β release of your software that shows basic functionality in place, integrated, and working, for the major components of the product. It should be possible to perform multiple operations that start at the client, reach through to the server, and return back to the client.

   Some functionality planned for in the final product may be missing in the β release, but the system should already be useful. A significant number of use cases should work, at least minimally. The documentation should reflect which commands or features are working and which are in progress.

   Your product need not be 100% bug-free. All known bugs should be documented in the bug database, and a user testing the system should not encounter a non-trivial number of unlisted bugs. Your system should be robust: errors should be gracefully handled as much as possible.

   The β release includes several elements packaged in two distributions:

   - Binary (installable) distribution includes the elements needed for an administrator to install your product on a server and run it remotely. For your web or android application, this may be a server package (zip file) or a url to invoke or android app to download. You do not need to package the database as part of the server for this assignment.

     The release should include clearly identified *release notes* in the package, describing (1) how to install and run the software, (2) which commands are working, and (3) a list of any known issues. Please deploy your β in a way that your development efforts won't affect the submission (e.g., isolated from your development, at a different URL, or on a different http port). This way, your stable β will persist, independently of your ongoing work.

   - Source distribution includes the elements needed by someone who is going to pick up the project at this stage and do further development. A separate package for each host target is often a clean way to organize this. (Again, you do not need to describe the underlying database installation for this assignment.)

     The source distribution should include clearly identified *release notes* in the package that describe how to build the product from the original sources. In grading, we will be looking for the ease with which a build can be made.

The release notes should state any assumptions, such as Visual Studio, Eclipse, or Android emulator installations. The release notes should also identify the source code repository and how to look at file change logs. A new developer may be looking at the change logs to understand why code evolved in a certain way, and we expect good developer documentation to be in place.

The release notes should also identify the issue-tracking database used in the project. We expect to see good usage of the issue-tracking database, especially for bugs that span development groups, by your team members.

Lastly, your software should be supported by tests, and a new developer will need instructions on how to run at least the following:

- unit tests for one major class or component in your system
- integration or system tests for the functionality represented by at least one use case.

Don't forget that tests need documentation, just like all code.

2. An updated system requirements specification (SRS) and system design specification and plan (SDS) documents, with change tracking on. Focus on the content changes, not the polish. If you have changed your feature list delivery schedule, add a note discussing the tradeoffs involved in making this decision.

3. A description of how you have applied one design pattern or principle (other than the iterator) from class, in your product design. Identify the pattern or principle and the code (files, lines, etc.) to which it applies. In grading, we will be looking for a meaningful (as opposed to a trivial) application.

Presentations

Your group presentation will be up to 10 minutes long. **The time limit is strict.** Your grade will be based in part on the presentation, so you should make sure you practice the presentation, and get outside feedback (from friends or classmates).

Your presentation will happen a week before the β release is due, specifically to allow for you to use feedback from the presentation to improve your submission.

Remember: You are presenting to the customer! Keep technical jargon to a minimum, and present at a high level. Convince the customer that your project will be a success and that you are aware and ready for the risks, and have a well-laid-out plan. Make sure you start with a brief recap of the product — those who have not been working on the project as hard as you may not remember your goals.

The structure of the presentation is up to you, but including a live tool demo is always a great way to generate excitement for the product.

As I mentioned before, Prof. Michael Ernst, at the University of Washington, has written up some helpful advice on giving technical talks. This is a good place to get hints on how to prepare.

Submission and Grading

Remember that your product must be live on the URL you give us when we grade it. As mentioned before, you may want to move your development work to use a separate port (in the case of a web app) so that it does not interfere with our testing of this release.

As before, a small part of your grade comes from the looks or aesthetics of your documents. They do not need to be beautiful or excessively formatted, but developers and your customers need to be able to read them and extract information from them. This means they should be clearly written, with proper spelling and grammar, clear wording, and formatted with enough organization to present your ideas clearly to the reader.

Your β release should reflect customer interaction. If your customer is surprised by some features you have chosen to implement (or not to implement) or by a direction your project has gone, this will reflect poorly on your grade.

One of your team members should turn in all the deliverable material together so that there is one coordinated input for the team. Put the team name in the filename of all components submitted.