

**CSE 403**  
**Autumn, 2013**  
**Beta Release (Phase 4a)**

**Beta Release Implementation:**

For the “Beta” release, you must produce a working, fairly stable, initial version of your project, reflecting most (but not necessarily all) of the overall features from your project, such as those listed in your requirements and design documents, and expanding upon the partial functionality implemented in your “Alpha” release.

By the “Beta” release, the major components of the product should be in place and integrated at a basic level. We are looking to see significant effort to implement a testable, usable product that implements an adequate amount of functionality to get a feel for the system and how it will work. At a minimum, try to complete at least one use case that touches all the major parts of your system. (For example, in the case of a web application, a user command invoked from the front end should be able to return data from the back end.) Most teams will have more than one use case operational.

The final phase, “Version #1”, will focus more on partial feature completion (if time allows), bug prioritization and fixes, unit and system testing, user interface polishing, code review, and documentation updates, rather than starting to code unimplemented features.

**Management-Oriented Documentation:**

**1) Testing Instructions, High-level Summary, and Schedule**

Please create a separate wiki page (title “Beta Release”) on your GitHub wiki that contains either the following information directly, or a link to a Google document that contains the following information:

- a direct link to your “Beta” deployment website or your APK binary download site,
- any necessary instructions about how to run and test your “Beta” release ,
- a list of features that are fully implemented for this phase,
- a list of features that are partially implemented for this phase (including a brief description of what is working/not working for a given feature),
  - additionally indicate which of these items are likely to be left unfinished by “Version #1”
- a list of features that are still completely unimplemented for this phase (if any),
  - additionally indicate which of these items are likely to remain completely unimplemented by “Version #1”
- along with any other information or instructions that would be required by the grader.
  - Especially make sure to provide information about how to log in or access any authenticated resources related to your project as appropriate (for example, if test user accounts already exist and should be used, or if the grader should create a new user account; if separate database authentication is required; if the grader needs to be logged into the CSE domain, cubist, Heroku, etc., before running and testing the product).

## **Management-Oriented Documentation:**

### **2) Issue- / Bug-Tracking**

Your bug tracker should reflect that at least some bugs have been found and fixed by this time. While the spec for the “Alpha” release was tolerant of major bugs and missing features (as long as they were documented), the bugs and errors tolerated for implemented functionality in the “Beta” release are expected to be more minor. Any bugs that remain in implemented features for the “Beta” release should mostly be side issues, such as an incorrect message text or minor error message, and not a scenario-stopping bug that prevents the customer from being able to perform normal usage and evaluate your work. As before, any known bugs or issues in your product should be documented using your group’s bug tracker. At a minimum, bug entries should include status (open, resolved, etc.), and a timeline to fix them.

## **User-Oriented Documentation:**

Update your user documentation from the “Alpha” release to reflect any new or updated behavior in the “Beta” release. By the final phase, “Version #1”, user documentation should be incorporated into your “product”; therefore, start moving towards that goal now if you haven’t already done so in the “Alpha” release. How to incorporate user documentation into your product is up to your team, but here are a few ideas (non-exhaustive list):

- part of the HTML of your site’s home page (or landing site for downloading the APK)
- split across several pages of your site (how-to basics, in-depth tutorials, troubleshooting FAQs, YouTube demos, etc.)
- part of user feedback (data validation messages, pop-up tool tips, etc.)
- a separate manual downloadable as PDF from the site’s home page/landing site

## **Deployment Logistics:**

Your team will continue to work on your project’s next release, “Version #1”, after turning in the “Beta” release, but you should leave the “Beta” release deployment available so the staff can test it. One way of achieving this is to put the “Beta” release at a separate URL or separate APK binary from your current version, so that one does not overwrite the other. When the grader goes to your “Beta” release location, the binary or site there should be in beta form and not in the form of the latest version of the code that you have completed since the beta. **Inform staff immediately if this constraint does not seem doable.**

## **GitHub Repo:**

Your project’s source code, along with any other resources necessary to deploy and run your product, must be checked into your version control system by the due date of this phase and **marked appropriately as belonging to the “Beta” release**. We expect you to be using your version control system properly, making small but coherent check-ins properly labeled with descriptive commit comments to indicate what has changed.

## **Phase Meeting:**

For this phase’s meeting, please meet with any of cse403 staff by the end of Friday 11/22 to outline your plans for the following week’s demo, as well as to discuss progress, get feedback, or address any concerns.