Rapid Storage Technology Reporting Project

Team Standards

September 24, 2013

Chad Dulake
Forrest Townsend
Nakai McCarty

Table of Contents

Introduction

This document discusses the organization of our capstone team, its meeting patterns and practices, and documentation style.

Team Roles

Although members have been assigned specific roles, all team members will participate in all aspects of project planning, design, implementation and testing.

Team Leader: Forrest Townsend

The team leader coordinates tasks and ensures work is progressing. They will also run meetings and to the best of their ability to resolve conflicts before presented to the team's mentor.

Customer Communicator: Forrest Townsend

This member is required to coordinate meetings and manage most communication with the team's sponsor.

Documentation Manager: Nakai McCarty

The documentation manager is in charge of the formatting and distribution of documents within and outside of the group. They will also be responsible for making sure that the documentation workload will be evenly distributed to each group member.

Mentor Communicator: Nakai McCarty

This person is required to communicate with the team's mentor about the operations within the group. They will also be required to schedule meetings with the mentor.

Recorder: Nakai McCarty

The recorder documents all key points discussed in team meetings. They will also be responsible for sending out what was discussed at each meeting.

Architect: Chad Dulake

This member ensures that the architectural decisions are followed through implementation till project complete.

Release Manager: Chad Dulake

The release manager is responsible for managing the versioning of the project. Along with versioning they will practice good repository habits with accurate commit logs and ensure that features get their own branch. They will be primary contact for merge conflicts and issues.

Quality Assurance Manager: Chad Dulake

This member will be responsible for the quality assurance throughout the project.

Webmaster: Chad Dulake

The webmaster is responsible for the management of the team website. They will be responsible for getting the server space and that they website clearly displays everyone involved and the goals of the project.

Meeting Procedures

Meeting Times

Each team member has a busy schedule. We have presently decided to meet regularly on Tuesdays after CS 476 with supplementary ad hoc meetings if need be. Scheduled meeting times are likely to change as our schedules evolve. If it becomes necessary to choose a new regular meeting time, it will be established by a unanimous team decision. In case a team member is unable to make a regular meeting, they will notify the other team members via e-mail as soon as they know they will be unable to attend.

Planned Topics

An agenda for each meeting will be assembled (likely by the team leader with input from the members) and issued prior to the related meeting. The date of issue for each agenda will be based on the nature of the topics to be addressed. At minimum, agendas will be issued 24 hours before the beginning of a scheduled meeting.

Proceedings

At minimum, each team member will provide a status update concerning their assigned tasks including recent progress and any problems they have encountered. This information will be recorded in documentation which is freely accessible to every team member. Next, the team will perform an overall progress evaluation. This process serves to remind team members of approaching milestone accomplishment dates, allow accomplished milestones to be marked down, and then adjust the timeline if necessary. After these required processes, the various items of the meeting’s agenda will be addressed according to their nature.

Proper Uses of Meeting Time

Settling Disputes

When disputes arise, regardless of their nature, they will be settled by an open discussion between all team members. If a consensus is unable to be reached, then the team mentor will be invited to share their opinions on the matter to further the discussion process. After this point, if the dispute still cannot be settled, then the team leader will use their discretion to make the decision they deem best for the team.

Planning and Meta-planning

Of course, regular meetings will be the time where 'the next steps' are arranged and discussed. Additionally, these meetings will serve as time to arrange future meetings and alert other team members of upcoming due dates, events, and personal schedule anomalies.

Team Self-Review

During the development process we will need to review ourselves every so often as we implement. Creating a schedule to make sure that we are still on track with what we need to accomplish and that we are doing it in a way that makes sense and is easy to follow. We will make for code-reviews at our demo's every week to make sure that the person in charge of a feature can go through their implementation and be reviewed by the other members all at once. These discussions will be constructive and should be a positive experience. The most important thing is that the person presenting does get put down by the members of the group; the other members need to be as positive as possible while reviewing the code because we are all in this together. This review will be informal with little documentation but will be a mandatory review. The only kind of documentation will be the suggestions. This will be in an email made by the members reviewing; everyone will get a review email.

Document Procedures

Tools

Professional documents shall be prepared in Microsoft Word (with compatibility for version 2010 on Windows), final submission shall be in the form of a printed or PDF document with any necessary fonts included, custom fonts or proprietary fonts like Adobe Garamond Pro should be avoided. Internal documents shall be written in reStructuredText or Markdown markup languages and should not use any special functions (like inline HTML) that would make it difficult to read as a plain text file, team members may use their favorite text editor to create or edit the documents.

Version Control

Documentation drafting will be done on Google Docs or another form of collaborative editing; final documents will be formatted accordingly in reStructuredText or Markdown and uploaded to our Git repository. Code version control and later modifications to documents will be done on the Git repository that is hosted on BitBucket. Git commits shall be meaningful and encompass a small amount of changes at a time. Merging will be handled by the Git merge command and merge conflicts are to be solved by the programmer, possibly with a need to contact the commit manager about it. We will commit to the trunk until a certain point is reached, after which feature development will be done on a branch basis, with only bug fixes being applied to the trunk until a feature is complete.

Formatting

Internal documentation will be written in reStructuredText or Markdown and will use the stylesheet of the reStructuredText/Markdown interpreter (or plaintext since they can be read as plain text) the writer does not need to worry about font/text size/page numbering/etc… but should structure separate sections with headers and format code using the code/monospaced markup for the respective markup language.

Professional documentation (which is that released to the client or mentor) shall have a cover page, consisting of the project name centered in a 20pt sans-serif font like Ubuntu, separated by a horizontal rule shall be the document title centered in a 24pt sans-serif font like Ubuntu, directly below the date will be written in centered 16pt sans-serif font in the following format “Month DD, YYYY” (eg. “September 24, 2013”). Two lines below (one line of space) shall be the names of the team in alphabetical order on separate lines in centered 16pt sans-serif font. A professional document will have a table of contents if its length or contents would benefit from it. Page numbers will always be listed in a large 24pt sans-serif font like Ubuntu at the middle right of the header. Sections/headers shall be formatted in left-justified 16pt bold sans-serif font like Ubuntu, subsection headers shall be 14pt bold sans-serif. The document body shall be comprised of 12pt serif font like Georgia or Garamond and shall also be left-justified. When the need arises for a monospaced font, Ubuntu Mono will be used. If a segment of code is being displayed, it should be displayed in a 1x1 table of a light gray background to differentiate it from the rest of the document, if necessary if may have a caption or figure number. Code syntax should not be highlighted by any style, except to point out a particularly interesting or talked about segment of code. The entire document shall be formatted with 1.15 line spacing except when necessary for a diagram or table or other object.

Composition and Review

All professional documents will be delivered to the document manager for final review and formatting no less than 48 hours before they must be submitted. Document draft review will occur during team meetings or in online synchronous collaboration.