

CSCI 265 Team Standards and Processes (Phase 2)

Team name: We Be Daves

Project/product name: See a Neevle, Hear a Neevle

Key contact person and email

- Dave Narealdave, nareal@somewhere.ca

Document structure

In this document we will be addressing three core areas of standards and processes:

- Documentation standards and processes
- Coding standards and processes
- Version control standards and processes

Each section includes discussion of how those standards/processes will be enforced, and how they will be reviewed for potential updates if/as needed.

Documentation standards and processes

A 'Documentation' directory will be maintained in the top level of the team project repository. Within that directory we will maintain the files shown in the list below (note that this list should be updated if/as needed as the project progresses). For each file we have listed a primary and secondary author/maintainer who will be responsible for ensuring the document adheres to standards and is kept up to date as the term progresses. Generally these are the lead/understudy for relevant project areas. Each team member will likely be contributing to each document, under the guidance of the primary/secondary author.

- [proposal.md](#) (Dave): the preliminary proposal deliverable (phase 1)
- [charter.md](#) (Dave): the team charter deliverable (phase 1)
- [requirements.md](#) (Davey): the product requirements deliverable (phase 2)
- [standards.md](#) (Duhyve): the standards and processes deliverable (phase 2)
- [logicaldesign.md](#) (David): the logical design deliverable (phase 3)
- [userguide.md](#) (Davey): the user guide/manual deliverable (phase 3)
- [technicaldesign.md](#) (David): the physical/technical design deliverable (phase 4)
- [proofconcept.md](#) (David): the proof of concept deliverable (phase 4)
- [testplan.md](#) (Duhyve): the test plan deliverable (phase 5)
- [closeout.md](#) (Dave): the project closeout deliverable (phase 6)

An 'Images' subdirectory will be maintained within the documentation directory, and any/all images referenced in the various .md files will be located within that directory. The file names used must begin with the name of the relevant document followed by an underscore and relevant expansion, e.g. 'technicaldesign_networklayout.png'.

To the extent possible, documents are to follow the organization/layout structure provided in the course project/phase specifications and the sample documents provided. If/when/where additional document sections are needed they will follow the same heading and layout choices to maintain a consistent look and feel.

Each contributing team member is expected to thoroughly check both spelling (Canadian English) and grammar for all the content they provide, and the primary author for each document will conduct final checks. The secondary author will carry out a final check for content written by the primary author.

The mechanism for submitting document content/revisions is via a 'Pull request', as described in the version control section later in this document.

Phase 2 (requirements and standards/processes) will serve as a trial run for our document standards and processes. One meeting agenda item in the first team meeting following the completion of phase 2 will be discussion of how well the standards and processes worked, and any possible changes required. Similar discussions can be held after later phases as the need arises.

Coding standards and processes

It is expected that this section will be refined once final choices are made for the development environment(s) and programming language(s) to be used, but the initial standards and processes are as follows:

- All code/revisions produced by team members will be reviewed by either the Development lead (David) or understudy (Dave) before being incorporated into the current code base. It may take up to 72 hours for the lead or understudy to carry out such a review, so team members are expected to have any proposed code/revisions submitted at least three days before the deadline for the relevant project phase/deliverable. Code/revisions made by the Development lead are to be reviewed by the understudy and vice-versa.
- The mechanism for submitting code/revisions is via a 'Pull request', as described in the version control section later in this document.
- Unless agreed otherwise, the Google Style Guides (google.github.io/styleguide) will be followed for the relevant chosen programming languages. Any exceptions must be approved by the Development lead and are to be listed here.

This section is to be reviewed/refined during phase 4, as the technical design and proof-of-concept stages should make clear the specific set of tools and languages and should reveal any weaknesses/gaps in the current plan.

Version control standards and processes

The team's github repo (TEAM_REPO_NAME_GOES_HERE) will be set up and maintained by the version control lead (Dave) and understudy (Davey), each of whom will have admin access.

The project will have three core branches: production, testing, and dev. The general purpose/philosophy for each of the three branches is as follows:

- production: the tested-and-approved branch, typically the version most recently submitted for marking

- testing: the we-think-it's-ready-for-marking branch, taken from dev when key milestones are reached, a clean and testable branch that we hope will become our next version for production
- dev: the working branch, containing code that the development lead/understudy have approved for inclusion but where the overall product is not yet ready for the next round of testing.

Each team member will have their own clone of the dev repository, and will carry out their own work within their own dev repo. Each team member needs to pull regularly from the team dev repo to ensure that their local dev is has incorporated any new approved changes made by other team members.

When a team member's local work reaches a point where they believe it is ready for incorporation into the team dev repo they need to:

- pull from the team dev repo
- resolve any issues/conflicts
- test their code to ensure it works correctly
- double-check that their work follows the relevant standards At this point they can post in the Pull-Request channel of the team discord. The version control lead (or understudy) can then pull their work into the team dev repo. Should issues unexpectedly arise during this pull, it may be necessary to coordinate between the author of the pulled code, the development lead (or understudy) and the version control lead (or understudy).

TEAM MEMBERS SHOULD NEVER DIRECTLY EDIT CONTENT IN THE TESTING OR PRODUCTION BRANCHES

With respect to commits by team members:

- no single commit should ever alter more than 200 lines of code
- commit titles should clearly and specifically identify the nature of the change made
- separate commits should be used to resolve separate issues
- the commit body should clearly explain the impact of the change from a behavioural aspect (how does the game look/act differently as a result of this commit?)

This processes outlined in this section are to be reviewed/refined following the completion of phase 2, as we will then have tested our processes on the production/revision of the requirements and standards .md files.