Nachhilfebörse HTL-Leonding Configuration Management

Versioning Methods and Tools

For sharing our files, we use GitHub.

Do you have all documents under version control?

Yes, we do have all of our documents under version control. The only exception are the Word documents (Project Proposal, System Specification…) when they are in progress. As soon as we submit them, we will add them to GitHub. During progress, we will work with Google Docs or just on one computer, because GitHub does not work well with Word documents.

Organization of artefacts

Due to the size of our project we think we will not need all eight folders of the proposal. Instead we will classify our files in the following folder structure.

* 01\_SourceCode (All the source code files, with sub division?)
* 02\_MicsDocuments (Project Proposal, System Specification…)
* 03\_Planning
* 04\_Others (Other files)

Variant Management

Organization of different versions and branches

The different versions will be stored on GitHub. That way we do not have to back our project up regularly and in case of emergency (bugs, project cannot build anymore) we can revert to an older version and start again.  
For branching we thought it would be convenient to use the given branching plan of the PowerPoint presentation, which includes:

* Master branch: This is the main branch of the project. It is just edited by merging.
* Development branch: This branch represents the progress of the project. It can be edited by directly committing of small changes or merging feature branches.
* Feature branches: Small features (usually done by one person) will be done in extra feature branches. They will be merged to the development branch when they are finished.
* Release branch: When a new release is planned, a new release branch will be added. It will be created from the development branch and when the release is done, the release branch will be merged to the master branch and the development branch.

Naming of new project versions

For the name-giving of the version of our project we will stick to the easiest way: The structured way (Code\_v1.2.4). The first number, e.g. 1, describes the major version (Version 2 could be completely different than version 1). The second one is the minor version (Interfaces from v1.1 are still available in v1.2, but new interfaces are added) and the third number describes the bugs fixed (v1.2.4 has less bugs than v1.2.3).