

Group 7 Brainstorming – GitHub code management tool

Code management tool helps development teams facilitate teamwork, track the complete development history, generate release notes, manage documentation and test and inform team members the changes of the source code. Version control systems, such as SVN and Git, allow version branching from particular states of code developed using SVN or Git repositories, and merging of different branches into a final state. In terms of large amount code, using version control tools can avoid uploading or downloading large files. Instead, only the changes, the log file, will be transmitted.

Search and name few systems/services which address the problem proposed in your project & find other services in the same category:

Alternate popular version control code hosts include [RubyForge](#), [Tigris.org](#), [BountySource](#), [Launchpad](#), [BerliOS](#), [JavaForge](#), [GNU Savannah](#). Taking a couple of these as examples compared to GitHub:

SourceForge is best known for its revision control system and majorly contains project wikis, metrics and analysis, access to a MySQL database and etc. However, it does provide Code Review, Private Branch, Shell Server, Mailing List, Forum and Announce components

Launchpad is currently widely used in the Ubuntu OS development and it does not provide web hosting or project wiki but including some additional functions like Translation System, Mailing List, Announce and Build System.

Describe how you found these services, e.g. through Googlesearch or a colleague told you:

Information was researched using Google, Wikipedia and ProgrammableWeb.com

Explain who is using these systems/services, e.g. end-users or programmers. Moreover, explain what problems do they solve using these systems/services!

- Team managers
 - Manage their team by adding as many developers as they want to the team define their roles (read-only, read-write, and admin-level access).
 - Divide project work into different pieces of branch, which can be combined after finished by individuals.
- Programmer
 - Hold living discussions for their codes and the changes.
 - Submit their comment for every version.
- Project testers
 - Download the very latest changes.

Explain what are the artifacts you need from these systems/services? How can you reuse them?

Github provides a REST API to give developers access to its services. They provide a developer page that outlines provided services (<http://developer.github.com/v3/>). Most useful here will be the Git Data and Repository services that allow us update or get versions by standard Git types (commits, trees), forking the repository by user, while also providing user management and social networking tools, such as adding collaborators and listing user followers.

Finally, explain how you discussed about these issues in your group:

The discussion was mainly held with face to face meeting. We also use Skype, SMS and emails to exchange our ideas.