

WikitoLearn Developers' meeting

2015/10/26

Today the responsables for the developing of WikitoLearn meet the newbies at the university of Milano Bicocca to start talking about the main changes ongoing our project.

Luca Toma:

The basic structure of our system is the **Docker**. It is a sort of “**machine in the machine**” that allows us to have a **common enviroment** for our work. First of all we need to clone the official repository from github site (**github/WikiToLearn**), then we are able to use a common enviroment for everybody. The code has to be **uploaded on github**. Once updated on github it is available for everybody. Git is a collaborative developing system that allows different people developing different parts of code to have a common structure that enables them to work on a common project. When we update a new version of code on github the machine steps into a “**pre-production**” phase. In this phase it checks that there is no loss of performance and stability in WikiToLearn. Then it enters the **production phase**, during which we call the server and tell it to use the last **tagged stable version**. At the moment the tagging process is not perfectly implemented.

The **main projects** are:

- Dockers and network design
- GUI ;
- OCG: offline content generation
- general code for the site;
- Latex-mediawiki conversion.

The **docker** creates in the personal machine a virtual container that runs a **minimal system**, so that every docker works on a specific and **minimal task** (database, main server, balancing beetween versions). These allow us to have a **minimum time** required to update and make the site running from an older version to the newer one. Since the docker system is linux-based it is **strongly recommended** for every developer to have a **linux distro** available for working.

Main Dockers are:

1. Mysql where we deposit site data ;
2. Websrv that exposes the site to the web
3. OCG that manages the generation of pdfs from the pages;

4. memcache is a trick to enhance site performances;
5. Haproxy is a network system to minimize downtime of the site during version updates.

So when you contact our system you first meet HAproxy that then speaks with ocg and webserv..

It is possible to run on the same machine two separate instances of our system and then switch them at please.

Davide Valsecchi:

OCG is an engine that gets data from the mediawiki page and converts them in a **json** file through the **parsoid code** (not a lot documented) and sends them to a **bundler** that prepares data for the rendering in different formats (text, latex -->pdf)

We are interested in the latexer file "**index.js**", that is a **javascript** file that receives data from bundler and translates it in latex code. We are focusing on some objects that are crucial to be implemented in this translation, such as the "**dmath**" instruction in wikipolylearn, the **tables** and the **images**.

The **main challenge** for this section is to implement the **tables** objects in the index.js file. By now this is totally absent from upstream code.

We are also starting a **new project** completely made by us. This is the parser **LatexToWiki** that translates a **latex** file into wikitext, managing the creation of the a tree of pages from latex structure.

Gianluca Rigoletti:

CSS configuration: front-end and graphic part. We do like the site as it is by now... but there are still **major improvements** to be done.

What we'd like to have is a **dynamic structure** that responds differently according to the device that is trying to access WikiToLearn.

We started from the navbar. We are not writing in pure CSS but with the **bootstrap library** that allows us to create a lot of content with ease and without compatibility problems.

By now WikiToLearn is using bootstrap 2 library that is quite old and actually not practical.

Main things to be done:

- Bring WikiToLearn to bootstrap 3 through the update of most classes and objects in the code.

Phabricator

Phabricator is a system of **task management**, divided by field of working. Every task has a **priority** and a **manager**. We use this system to write down every step that needs to be done to improve our site. Even if it is easy just to say what we need to do it'd be best to have everything written down in order to get a good **list of priorities**.

We also remember to use the **official mailing list** for important messaging.