MEMORANDUM

From: Molina Garca-Retamero, Antonio

Subject: Summer Student Report

Date: September 4, 2015

---

1 Project sum up

The aim of the project was to perform the processing required for automatically get a PDF/A version of the CERN Library documentation. For this, it is necessary to extract as much metadata as possible from the sources files, inject the required data into the original source files creating new ones ready for being compiled with all related dependencies.

Besides, I've proposed the creation of a HTML version consistent with the PDF and navigable for easy access, I've been trying to perform some Natural Language Processing for extracting metadata, I've proposed the injection of the cern library documentation into the HTML version of the long writeups where it is referenced (for instance, when a CERN Library function is referenced in a sample code) Finally, I've designed and implemented a Graphical User Interface in order to simplify the process for the user.

2 Sources processing

As it has been said in the sum up, the main task is to get as much meta-information as possible. It will allow us to index and query the documentation as well as build a PDF/A version with metadata attached.

Taking a look at the source files of the short writeup library, there is a common structure made of custom latex commands that can be parsed by regular expression for getting information like $Autor, Lenguaje, etc....$ So it was easy to parse them and to build a data base up with the information recovered.

For the long writeups, as they do not share a common structure, there not exist a justification for automatic processing that will be hard to implement. So, for easyly processing, I have designed a simple graphical for metadata captation and processing.

I would add that before disposing of the sources in the first weeks, I was performing some test and approaches for using natural language processing techniques for extracting information from the long writeups as keywords, categories, etc...

3 PDF/A and HTML creation process

For the creation of the PDF/A and HTML versions from the sources, I have considered them in two ways depending on if they are long or short writeups.

For the short writeups, they were sharing a common structured with custom latex commands. There were also a common template in which each short writeup is included for compile them. So, once I have parsed them for extracting metadata and injected the dependencies and metadata for the PDF/A generation, I just compile them and store the data into a database. I also generate a simple HTML from the latex for being injected in other documents references. Finally, It generates a HTML version which looks like the PDF for easy publication.

By using the GUI, the user can select a folder where the short writeups are stored. It will automatically look for all documents trying to match them with the structure it is suppose to have. Once all short writeups are found and parsed, it will process them in a batch automatically.

For the long writeups, they are intruduced one by one. The user can choose a folder where the files are stored and introduce the relative data throughout the user interface. It will copy all files needed for the compilation, inject the dependencies and the metadata for PDF/A compilation. Finally, it will create a HTML version consistent with the PDF.

As I said before. I'm working in the feasibility of look for the possible references to other documented sources, like references to functions documented in the short writeups, for instance. After that, it takes the reduced version of the html if exists and inject it in the final HTML version of the long writeup.

4 Uploading to CDS process

With all this documents, matadata and files, I have written an script for generate a JSON file that is sent to de CDS. The generated files $PDF/A, HTML$ are copied to a public server and its recovered and linked from the CDS to for make them available.

5 GUI for easy use

Finally, I have design a GUI that makes easy to perform the single and batch processing. It also allow the user to process long writeups by introducing the corresponding metadata. The script looks for all needed files, inject the metadata, generates the PDF/A and the consistent HTML. At this moment, I'm still working in look for all possible references to already processed documents and injects the referenced. documentation in an friendly user way for auto-documenting as much as possible.

6 Week short report

- **Week 1:**
  - First Contact with my supervisor
  - Getting access to virtual HTML documentation
  - First attempts to parse the documentation
- **Week 2:**

  - Creating data model from HTML parsing
  - Creating web crawler for getting data from web and filling data model
  - Firsts attempts to extract information from *PDF* documentation
  - Proposal of process the documentation by using Natural Language processing in order to auto-extract meta-information from the documentation
- **Week 3:**
  - Getting access to *tex* bibliography
  - First attempts to compile *tex* files
  - Designing of the mechanism to modify the *tex* file for compile to PDF/A
- **Week 4:**
  - Dependencies and some other minor fixing for latex compilation with modern latex compilers
  - Working on different attempts to get the most consistent version of the PDF file in HTML for easy access and navigation through it
- **Week 5:**
  - First attempts and prototyping for individual and batch processing
  - Working on issues related with the compilation of long writeups sources by using modern latex compilers
- **Week 6:**
  - First attempts and prototyping for injecting short writeups documentation into the HTML version of the long writeups.
  - Keeping working in issues with some long writeups compiling
- **Week 7:**
  - Designing and implementing of a Graphical User Interface for easy use of all the functionality developed
- **Week 8:**
  - Keeping working in the implementation of the Graphical User Interface
- **Week 9:**
  - Fixing minor issues in the whole process
  - Keep working in the injection of short writeups documentation into the HTML version of the long writeups.