Unplagged Developers Manual

Building the Plagiarism Detection Cockpit

Term paper for the master project I

Mentoring Teacher: Prof. Dr. Debora Weber-Wulff

Department Economics II
HTW Berlin – University of Applied Sciences

Elsa Mahari (s0534217) <Elsa.Mahari@gmx.de>
Heiko Stammel (s0534217) <heiko.stammel@googlemail.com>
Benjamin Oertel (s0522720) <benjamin.oertel@me.com>
Dominik Horb (s0534217) <dominik.horb@googlemail.com>
Tien Nguyen (s0534217) <idontwant2missathing@yahoo.com>

Contents

1. Preface

Even though the big media coverage and interest in plagiarism in Germany has very much subsided[Goo12] after Minister Guttenberg had to resign, because of the plagiarisms found in his doctoral thesis, the initial idea for the creation of the “Unplagged” project, whose development approach will be described here, can be found in this very case of plagiarism. Related to it were the formation of the GuttenPlag and it’s descendent VroniPlag. Both are Wiki-based communities that are collaboratively discovering and collecting plagiarism in their respective cases and are kind of the role models for the way the Unplagged system is developed.

The initial project idea and context were provided by our professor Dr. Debora Weber-Wulff and the two term master project every media informatics student at the HTW-Berlin has to take. As professor Weber-Wulff is a well known expert in Germany on the topic of plagiarism, does research in this field for over ten years[SO11] and is actively involved in the VroniPlag community, she came up with the idea to build a dedicated system — a “Plagiarism Detection Cockpit”[WW11], that is modeled after the experiences that were made with the workflow used in VroniPlag and GuttenPlag.

So, to put it in a catchy marketing phrase, here is what Unplagged aims to become:

> **Unplagged is a simple, web-based, collaborative system to help discovering, collecting and documenting plagiarism in scientific papers.**

To make things a bit more conceivable, we also often refer to it as a mixture of a very specialised text editor, with a focus on comparing texts and marking passages and a modern project management tool like Redmine or JIRA, to manage the collaborative aspects of the system. The big distinction we make to other plagiarism software on the market is, that the approach is not to autodetect plagiarism, but focused on aiding the workflow of the users while searching for plagiarized fragments inside a scientific paper, a homework assignment or any other kind of probable plagiarism.

This present document will be your handbook, if you want to get started helping in the development of this open source project, which is licensed under the GNU GPLv3.

1.1. Chapter Overview

One of the biggest problems we faced at the start was, that none of the team members had written a longer scientific text than a bachelors thesis and therefore the experience we got with actual scientific writing was very limited and very specific to the field of computer science. We understand the ethical problems, that come with the betrayal of good scientific practice of plagiators, but we simply can not relate easily to the amount of work that has to be put into a Ph D., or be as passionate about plagiarism as Prof. Weber-Wulff always is, because we never experienced it.

That is why we had a lot of catching up to do on the most important history behind VroniPlag, the different types of plagiarism, different citation styles and the research Prof. Weber-Wulff and others had already done on systems that try to help finding plagiarism. The chapter 2, The current situation – A plagiarism overview, will give a brief overview of the most important topics to get you up to speed with the domain of the software, if you are not already familiar with it.

Although we are using agile methods for the development process, the chapter System Requirements will give a more classical collection and description of the parts of the system, that already exist or that we identified as necessary parts of Unplaggedhow we understood the requirements of the VroniPlag workflow.

If you know all those things already and simply want to get started working and coding, you should probably jump to Developing Unplagged. This chapter will give insights into the project workflow, the basic installation steps and all necessary tools for you as a developer.

2. The current situation – A plagiarism overview

2.1. Basic Classification of Plagiarisms

2.1.1. Copy&paste

2.1.2. Copy, shake&paste

2.1.3. Patchwriting (rewording)

2.1.4. Structural plagiarism

2.1.5. Translations

2.2. How to detect plagiarism

2.2.1. Software systems

2.2.2. Human approach

2.3. Vroni Plag

3. System Requirements

3.1. Target Group

3.2. User roles

3.3. Basic functionalities

3.4. Document Parser

3.5. Detection Modes

3.6. Plugin Architecture

3.7. Use Cases

4. Developing Unplagged

4.1. Installation

4.1.1. Tesseract

4.1.2. Simtext

4.1.3. Imagemagick

4.2. Development Environment

4.2.1. Git

4.2.2. Netbeans

4.2.3. Staging and Preview System

4.3. Architectural Goals

4.3.1. Progressive Enhancement

4.3.2. Test Driven Development

4.3.3. Responsive Design

A. Sprints

B. Minutes

C. Time Logging

D. Selected Sources

Bibliography

[Goo12] Google. News search interest: plagiat. http://www.google.com/insights/search/#q=Plagiat&geo=DE&date=1%2F2011%2013m&gprop=news&cmpt=q, 2012. [Online; accessed 07-February-2012].

[Mar11] Ethan Marcotte. *Responsive Web Design*. A Book Apart, 2011.

[SO11] Spiegel-Online. Streit über vroniplag-gründer ”verprellter liebhaber oder spd-mitglied, das ist egal”. http://www.spiegel.de/unispiegel/wunderbar/0,1518,778626,00.html, 2011. [Online; accessed 26.02.12].

[WW11] Dr. Debora Weber-Wulff. Master project: Plagiarism detection cockpit. http://www.f4.htw-berlin.de/~weberwu/classes/HTW/projects/Plagiarism-Detection-Cockpit.shtml, 2011. [Online; accessed 26.02.12].