

Anonymouth

Integration Test Plan

ArmonEntezari
Karan Hansaria
Chris McGuire
Jon McGrath
Paul Vu

Advisor: Rachel Greenstadt

Table of Contents

1. Introduction.....	9
1.1. Purpose.....	9
1.2. Structure of the Document	9
1.3. References	9
1.4. Glossary.....	10
2. Testing Environment	11
2.1. Structure	11
2.2. Constraints.....	11
2.3. Assumptions.....	12
2.3. Participants.....	12
2.4. Entry Criteria	12
2.5. Exit Criteria	13
3. GUI Testing	13
3.1. UML.....	14
3.1.1. MainFrame.....	14
3.1.1.1. Attributes.....	15
4. Test Cases	26
4.1. UML.....	26
4.1.1. FeatureHandler.....	26

1. Introduction

1.1. Purpose

The purpose of this document is to provide a testing plan for the Anonymouth anti-stylometry tool. This testing plan is directly formulated from the Software Design Document and its goal is to provide total coverage for the requirements set forth in the Software Design Document. Priority 1 requirements are the primary concern but all aspects of the application shall undergo stringent testing.

1.2. Structure of the Document

This document is divided into three different sections:

- Section 2 (Testing Environment) – The minimal requirements for a satisfactory testing environment and prior considerations
- Section 3 (GUI Testing) – Testing that pertains to the front-end GUI
- Section 4 (Test Cases) – Testing that pertains to Anonymouth and its internal functions directly

1.3. References

[1] - "Hallmarks of Good Writing." *College of Business » University of Illinois at Urbana-Champaign*. Web. 04 Feb. 2011. <http://business.illinois.edu/ce-brown/accy304spg01/hallmarks_writing.htm>.

[2] - "Penn Treebank P.O.S. Tags." *University of Pennsylvania Department of Linguistics*. Web. 04 Feb. 2011. <http://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html>.

[3] - "NodeBox | Linguistics." *NodeBox | Home*. Web. 04 Feb. 2011. <<http://nodebox.net/code/index.php/Linguistics>>.

[4] - Button, A. Mouse. "What Is Graphical User Interface? -A Word Definition From the Webopedia Computer Dictionary." *Webopedia: Online Computer Dictionary for Computer and Internet Terms and Definitions*. Web. 24 Nov. 2010. <http://www.webopedia.com/TERM/G/Graphical_User_Interface_GUI.html>.

[5] - "About." *Python Programming Language –Official Website*. Web. 24 Nov. 2010. <<http://www.python.org/about/>>.

[6] - "Synonym | Define Synonym at Dictionary.com." *Dictionary.com / Find the Meanings and Definitions of Words at Dictionary.com*. Web. 24 Nov. 2010. <<http://dictionary.reference.com/browse/synonym>>.

[7] - "The Parts of Speech." *Le Centre D'écriture - The Writing Centre B*. Web. 04 Feb. 2011. <<http://www.writingcentre.uottawa.ca/hypergrammar/partsp.html>>.

[8] - "TkInter - PythonInfo Wiki." *Python Software Foundation Wiki Server*. Web. 04 Feb. 2011. <<http://wiki.python.org/moin/TkInter>>.

1.4. Glossary

Corpus: A large collection of writings.

Feature: A writing characteristic that can be identified, for example, word count of a writing sample.

GUI: Acronym for Graphical User Interface. It is a program interface that takes advantage of the computer's graphics capabilities to make the program easier to use [4]

Part of Speech: Explains how a word is used in a sentence [7]. The parts of speech Anonymouth uses for analysis are interpreted and provided via the Penn Treebank [2].

Python: A remarkably powerful dynamic programming language that is used in a wide variety of application domains [5].

Stylometry: The study of writing style. Used to examine the writing style and identify the author of a piece of writing.

Synonym: A word having the same or nearly the same meaning as another in the language [6].

2. Testing Environment

This section details the required environment for acceptable testing as well as goes into detail about how detailed the testing will be and the assumptions that are necessary to obtain complete coverage.

2.1. Structure

To create a comprehensive test plan, all modules of the application must be considered. As of such, the main areas of testing are between the GUI and the back-end, the back-end with external tools, and the back-end with Stylo. The test cases will focus on but are not exclusive to the interactions between these modules.

<TABLE? / graphic?>

2.2. Constraints

The Anonymouth tests will only be able to test its interface with Stylo. Tests will not include the effectiveness or implementation of Stylo. The tests are in no way designed to check the information it receives from Stylo other than the type of information that comes from the program and not the integrity of said information.

2.3. Assumptions

Acceptance testing has been or shall be done separately and not included in the unit testing. User input will have no part in the integration testing and is to be handled in other test plans. The tests herein must be automated and offer complete coverage for the interoperability of the modules defined in the Software Design Document.

2.4. Participants

Test Team Leader: Armon Entezari

Testers: Karan Hansaria, Chris McGuire, Jonathon McGrath, Paul Vu

2.5. Entry Criteria

Testing shall not begin until the following have been completed:

- Acceptance Testing and Unit testing have been completed.
- The testing environment has been set up and meets the system minimal requirements outlined in the Requirements Document
- The latest Software Design Document has been approved and reviewed
- The testing environment has a fully functional version of the Anonymouth application

2.6.Exit Criteria

Testing shall not be considered completed after the following have been completed:

- All requirements set forth in the Requirements behave according to specification, passing
- At least one (1) requirements set forth in the Requirements behave not according to specification, failing

3. GUI Testing

3.1.Corpus Submission via New Project

Description:

Create a new user project

Precondition:

- New Anonymouth instance is running
- Listeners are created and active

Action:

1. Simulate Action: Mouse click 'Accept Corpus'

Postcondition:

- New corpus is created through the FileHandler

- Folder is created containing the corpus

3.2. Corpus Editing via Options

Description:

Edit corpus through the Options menu

Precondition:

- An anonymouth project is open
- Listeners are created and active

Action:

1. Simulate Action: Mouse click 'Accept Corpus'

Postcondition:

- Corpus is modified through the FileHandler
- Original corpus folder is modified containing the new corpus

3.3. Corpus Selection via New Project

Description:

Select existing corpus through the New Project prompts

Precondition:

- An anonymouth project is open
- Listeners are created and active

Action:

1. Simulate Action: Mouse click 'Accept Corpus'

Postcondition:

- Corpus being pointed to is changed by the FileHandler
- No change to the corpus folders but correct corpus folder is returned

4. Test Cases

4.1. Anonymouth obtains Feature List 'Pickle' object

Description:

Anonymouth calls Stylo program to retrieve a features list

Precondition:

- Stylo is running concurrently with Anonymouth
- A corpus and sample document are present

Action:

1. Call Stylo's function and set variable for list

Postcondition:

- New features list is returned
- No error occurs and the list is saved successfully and is what is expected

4.2. Anonymouth obtains Feature List and saves as XML

Description:

Anonymouth calls Stylo program to retrieve a features list and save it

Precondition:

- Stylo is running concurrently with Anonymouth
- A corpus and sample document are present

Action:

1. Call Stylo's function and set output to XML

Postcondition:

- New features list is returned
- No error occurs and the list is saved successfully in XML format to a specified location

4.3. Anonymouth opens a Feature List saved as XML

Description:

Anonymouth calls Stylo function to open previously saved XML feature list

Precondition:

- XML loaded is formed correctly
- XML exists in a specified location

Action:

1. Call function to open XML file and parse into a feature list

Postcondition:

- New features list is returned
- No error occurs and the list is parsed successfully and is what is expected