CARDIFF UNIVERSITY SCHOOL OF COMPUTER SCIENCE AND INFORMATICS

INITIAL PLAN

Web Interface for FlexiTerm

ADEDAMOLA AGBONYIN
1466169

SUPERVISOR
IRENA SPASIC

MODERATOR
JING WU

CM3203 One Semester Individual Project 40 Credits
February 05, 2018

Project Description

Term recognition is a method of information extraction by which we extract terms (relevant to a particular domain) from texts. In this context, a term is a word or set of words used to express a concept in a domain or branch of study.

FlexiTerm is a command-line Java application for automatic recognition of multi-word terms in texts such as medical journals, research papers etc. Put simply, FlexiTerm works by first performing linguistic filtering to select term candidates, followed by a frequency-based measure to qualify a candidate as a term (Spasić et al., 2013). The recognised terms are then output in three formats:

- a HTML table, ranked by their term-hood scores,
- a plain text file containing a list of all recognised terms,
- a list of regular expressions in Mixup (a simple pattern-matching and information extraction language).

This project will however not be concerned with the inner workings of FlexiTerm, but provide a web interface through which users can directly interact with the application and its output, so that rather than downloading and running the source code, the user can simply access the FlexiTerm web page, provide their input, and be able to view and download the output.

The user interface for the web application will adhere strictly to Ben Shneiderman's principles for designing interactive systems from his book *Designing the user interface*.

The supervisor is the developer of the FlexiTerm tool and is thus the client for this project. Information about the tool and its source code are available here:
http://users.cs.cf.ac.uk/I.Spasic/flexiterm

Objectives and Features

The main aim of this project is to create a user-friendly, full-stack web application that uses the existing functionality of FlexiTerm. The back end of the application will be connected to the FlexiTerm Java app for processing user input and receiving output. For the front-end of the application, I plan to use JavaScript, HTML, jQuery and Ajax, and for the back-end either Python (main choice) or PHP. No additional hardware will be required for development.

The main objectives for this project are as follows:

- Design a user-friendly web interface using Shneiderman's Eight Golden Rules of Interface Design (Shneiderman, 1997) as a guide.
- Implement a back-end for the web interface that allows a user to input and process texts for term recognition, and view and re-use/download the output.

- Conduct functionality testing against requirements to ensure the functionality of FlexiTerm is maintained and not altered in anyway, but only made easier through the web interface.
- Conduct usability testing with a small group of users to ensure that the average user can easily access and use the functions of the web application.
- Host the application as a web page on the client's home page [users.cs.cf.ac.uk/I.spasic].

The application will be tested against the proposed main and additional features (eventually revised into functional and non-functional requirements) to ensure all objectives are met.

Features are expected to include the following:

Main

- The user must be able to input their text file by pasting into a textbox, drag and drop or upload, or by URL.
- The user's input must be validated first, and then processed by the FlexiTerm back-end.
- The output must be returned in a web page such that the recognised terms are highlighted, with the option to also view in plain text, table format, ranked according to term-hood scores.
- The user must be able to download the output in appropriate formats, e.g. csv for the table, plain text etc.

Additional

- The application should use cookies such that when the page is refreshed, the data/output is not lost.
- The application should be able to handle several users at the same time.
- The application should include basic search engine optimisation so that users can easily find the FlexiTerm site when searching for text recognition tools
- The application should be compatible with different browser types and versions.
- The application should be lightweight enough to load with little delay.
- The application should be able to handle large-enough files, though a reasonable size limit should be put in place.
- The web page should contain additional details such as source code, terms of use and a simple quick start guide for new users.
- The code should strictly adhere to good coding standards that can be easily re-used/expanded by others.
- The user should be able to adjust some FlexiTerm settings.

Possible Limitations

- Misunderstood/difficult/changing requirements.
- Ineffective project/time management, bad estimation.
- Poor development methodology: Insufficient testing etc.
- Skill/knowledge limitation.

Ethics

This project will involve human participation when conducting usability testing of the finished application and will thereby require ethical approval by the university. Following submission of the initial plan, I will confirm with my supervisor whether or not this is necessary and proceed as advised, making sure that if applicable, ethical approval is obtained before usability testing commences.

Work Plan

The Agile methodology is best-suited to this project: each major feature to be added will go through iterative stages of planning, design, coding and testing before being deployed. This way, there is always a functioning version of the application available, ensuring that all features work well together, thus, avoiding a situation where errors are caught at the last-minute.

Using Agile also makes it easier to revise requirements peradventure a feature is too difficult to implement or deemed unnecessary, for example. Trello will be used to manage this project in line with Agile standards, and code base will be maintained on GitLab [gitlab.cs.cf.ac.uk] for proper version control.

Meetings with the project supervisor will take place Monday of each week, along with other students working on this project topic. I aim to achieve a different milestone by each meeting.

Week	**Task**	**Deliverables**
Week 1 29/01/2018 – 05/02/2018	- Research into feasibility of project; identify methodology to use, project management tool(s) to use - Test run and read paper on FlexiTerm in order to gain an understanding of its main functionality - Work on the draft of the initial project plan, submit and meet with supervisor for feedback (05/02), make changes accordingly for final version	Final version of Initial project plan
MILESTONE: Completion and submission of initial project plan		

Week 2 06/02/2018 - 13/02/2018	- Identify and document functional and non-functional requirements - Research and make a decision on what back/front-end technologies will be best suited (e.g. PHP or Python) following requirements analysis - Begin to design first draft of the user interface wireframes using Balsamiq following closely Shneiderman's principles and drawing from similar existing tools such as *Termine* [http://www.nactem.ac.uk/software/termine]. - Meet with supervisor for update on progress/feedback (12/02)	- Functional & Non-functional requirements - First draft of UI design - Balsamiq wireframes
MILESTONE: Completion of requirements and UI design (first draft)		
Week 3 14/02/2018 – 21/02/2018	- Revise UI design following feedback - Design UML diagrams based on UI design and unit tests based on requirements - Break down coding tasks into sub-tasks and create Trello boards to reflect this - Meet with supervisor for update on progress/feedback (19/02) - Begin with back-end code that will enable linking the output from the java application onto the web front-end - Create rough version of the front end with simple buttons to accept input from URL/uploads/copy and paste.	- Final revision of UI design including UML diagrams - Unit tests to test against functionality - Basic skeleton of front-end - Ground work for back-end functionality
MILESTONE: Completion of UI design (final draft)		

Week 4 22/02/2018 – 1/03/2018	- Continue work on back-end functionality: should accept input from front-end, process and produce output - Also continue work on front-end, include functionality for processing different types of input - Test front and back ends against test cases/unit tests [following agile methodology] - Meet with supervisor for update on progress/feedback (26/02)	- Input from the user can be accepted and processed - User can now input data in different formats
Week 5 02/03/2018 – 09/03/2018	- Include input validation on the front-end, limit file sizes to 5mb (or a reasonable amount), input should contain words rather than just symbols and blank space - Back-end output should be returned in a format (such as JSON) for easy conversion to different display formats for the user – this should have been discussed with the supervisor - Begin implementing UI design on front-end, including functionality in increments - Test front and back ends against test cases/unit tests [following agile methodology] - Meet with supervisor for update on progress/feedback (05/03)	- User input now validated before processing to prevent crashing - FlexiTerm output convertible to different formats for the user and downloadable
MILESTONE: First phase of back-end completed, first phase of front-end started		

Week 6 10/03/2018 – 17/03/2018	- Finalise back-end functionality - Perform thorough functionality testing of back-end against test cases and record results - Include reset buttons and additional useful options on the front-end; add functionality to change min/max settings on the FlexiTerm back-end (optional, but ideally should be added around now) - Meet with supervisor for update on progress/feedback (12/03) - Begin work on using cookies - Add additional details to the web page	More functionality and useful information added to the web page
MILESTONE: Second phase of back-end completed, second phase of front-end started		
Week 7 18/03/2018 – 24/03/2018	- Meet with supervisor for update on progress/feedback (19/03) - Finish cookie functionality - Add some metadata to increase SEO (possibly publish on own homepage and test) - Begin thorough front-end functionality testing	- Cookies added - SEO implemented - Application must be fully-functional with all the functional requirements, and [some/all of the] non-functional requirements implemented
MILESTONE: Second phase of front-end completed		
EASTER BREAK BEGINS		
Easter break 24/03/2018 – 15/04/2018	- Finalise front-end functionality testing - Perform usability testing with a small group of users, also with different browser types/versions - Identify and fix any issues revealed - Begin first draft of final report and code documentation	- Browser compatibility tested - Usability testing concluded - Lay groundwork for final report and code documentation
MILESTONE: Usability testing concluded		
Weeks 8 – 9 16/04/2018 – 30/04/2018	- Meet with supervisor for update on progress/feedback (16/04) - Additional checks/changes on app [if applicable] - Finish first draft of final report	First draft of final report completed

	- Meet with supervisor (23/04) for feedback on first draft of final report - Begin final draft of report with respect to supervisor feedback	
MILESTONE: First draft of report completed		
Week 10 onwards 24/04/2018 – 11/05/2018	- Complete final report, taking into account supervisor feedback and using a grammar/plagiarism checker - Possible final meeting with supervisor - Final checks/changes - Deploy application - Submit project on 11/05	Final draft of final project report completed
MILESTONE: Final draft of report completed and submitted		

Gantt Chart

	1	2	3	4	5	6	7	E	E	E	8	9	10	11	12
Initial plan															
UI design															
Back-end development															
Back-end functionality testing/fixing															
Front-end development															
Front-end functionality testing/fixing															
Usability testing															
First draft final report															
Final draft final report															

References

Shneiderman, B. (1997). *Designing the user interface*. Reading, Mass.: Addison Wesley.

Spasić, I., Greenwood, M., Preece, A., Francis, N. and Elwyn, G. (2013). FlexiTerm: a flexible term recognition method. *Journal of Biomedical Semantics*, [online] 4(1), p.27. Available at: https://jbiomedsem.biomedcentral.com/articles/10.1186/2041-1480-4-27[Accessed 31 Jan. 2018].

Cs.cf.ac.uk. (2018). *Computer Science and Informatics Ethics*. [online] Available at: https://www.cs.cf.ac.uk/ethics/ [Accessed 1 Feb. 2018].