Lab II – Prototype Product Specification for CLASH

Blue/Purple Teams

Justin Bennett

CS411W

Professor Brunelle, Professor Price

March 25, 2015

Version 1

Contents

[**1. Introduction 3**](#_Toc415041997)

[**1.1 Purpose 4**](#_Toc415041998)

[**1.2 Scope 5**](#_Toc415041999)

[**1.3 Definitions, Acronyms, and Abbreviations 6**](#_Toc415042000)

[**1.4 References 8**](#_Toc415042001)

[**1.5 Overview 9**](#_Toc415042002)

[**2. GENERAL DESCRIPTION 10**](#_Toc415042003)

[**2.1 Prototype Architecture Description 14**](#_Toc415042004)

[**2.2 Prototype Functional Description 15**](#_Toc415042005)

[**2.3 External Interfaces 17**](#_Toc415042006)

[**2.3.1 Hardware Interfaces 17**](#_Toc415042007)

[**2.3.2 Software Interfaces 17**](#_Toc415042008)

[**2.3.3 User Interfaces 17**](#_Toc415042009)

**List of Figures**

*Figure 1.* Improved Process………………………………………………………………………6

*Figure 2.* Prototype Major Function Component Diagram ……………………………………..15

**List of Tables**

*Table 1.* Feature Comparison between Real World Product and Prototype …………………….12

# Introduction

America has always been a country with welcoming arms to people across the world. Not only does this make the culture more diverse, it results in more diversity in primary, secondary schools and institutions of higher learning. As English is the primary language of instruction of classrooms, teaching students who are not proficient is a significant challenge in subjects other than English. These students identified as English as Language Learners, or ELLs, face a difficult problem when attempting to learn or improve low English skills while also enrolled in collegiate classes. Currently there are interactive tools built to help students identify parts of speech, aid in comprehension, and increase reading speed but none of these tools are built together. This senior design product intends rectify this problem with a interactive learning suite, the Color Lexical Analysis algorithm and Slash Reader, or CLASH.

Nearly seven hundred thousand foreign students were enrolled in US universities in the year 2010 with that enrollment trending upward in the last thirty years[[1]](#footnote-1), with many of these students coming from non-English speaking countries the need for English as a Second Language, ESL, teaching aids is needed to graduate on time. Considering that approximately 2.5% of teachers who instruct ELLs possess English as a second language degree or are bilingual[[2]](#footnote-2), this fact emphasizes the need for additional teaching resources.

* 1. **Purpose**

The CLASH tool is designed to aid the instruction of ELLs with an interactive program that will make these modifications to entire text documents that students will be able to read outside of the classroom. The program will allow for more specialized instruction during classroom time as the more common examples can be learned without the aid of the instructor. The CLASH program is a web based solution to aiding English Language Learners. Using the Single Page Application model the user will access all functionality from a webpage that has a persistent connection.

. The CLASH program will parse user documents identify different parts of speech and certain groups of words that when requested, be highlighted. Slashes will be inserted between groups of words that are frequently grouped together to help shape meaning and coherence in text also known as a “lexical bundle”. CLASH also includes a mode that will display lexical bundles; CLASH will improve both reading speed and comprehension.

*(This Space Intentionally Left Blank)*

* 1. **Scope**

The process for teaching English to ELLs currently is: the instructor at a whiteboard, will write a sentence, identify the different parts of speech using different colored markers and draw a slash in between logical partitions. This is a tedious and time consuming method when time and attention are in short supply. The proposal for this software was made by Greg Raver-Lampman, a Professor in the English as a Second language program at Old Dominion University. He is looking to facilitate learning of the English language with a program that will assist ESL students in identifying different parts of speech and lexical bundles in a text document.

*(This Space Intentionally Left Blank)*

**1.3 Definitions, Acronyms, and Abbreviations**

**CLASH:** Color Lexical Analysis algorithm and Slash Handler.

**Client Side:** The user-interface of CLASH.

**COLRS:** Colored Organized Lexical Recognition Software. Module of CLASH that colorizes part of speech based on what NLTK tokenizes and returns.

**Document Processor:** A Server Side component responsible for processing the text entered by an Instructor user type.

**ELC:** English Learning Center at Old Dominion University.

**ELL:** English Language Learner.

**ESL:** English as second language.

**GUI:** Graphic User Interface

**HTML:** Hyper Text Markup Language

**IBT:** International Benchmark Test

**Intensive English Program:** A short and intensive English language training program offered by US colleges and universities to improve the English language skills of international students who did meet the minimum TOEFL scores for typical enrollment.

**JS:** JavaScript

**JSON:** JavaScript Object Notation. A nested data structure commonly used to pass data between a server and a client.

**Lexical Bundle:** A group of words that occur repeatedly together, or represent a single thought

**MFCD:** Major Functional Component Diagram.

**NLP:** Natural Language Processing

**NLTK:** A suite of libraries and programs for symbolic and statistical natural language processing (NLP).

**Node.js:** Open source, cross-platform run-time environment for server-side and networking applications.

**NoSQL:** (often interpreted as Not only SQL) database provides a mechanism for storage and retrieval of data that is modeled in means other than the tabular relations used in relational databases.

**POS:** Part-of-Speech such as noun, adjective, verb, etc….

**Python:** a widely used general-purpose, high-level programming language.

**Server Side:** The back-end of the CLASH system responsible text processing, the database, user-authentication, and web-hosting.

**Slash:** Module of CLASH that Slashes text into lexical bundles and displays them

**Slash Playback**: Module of CLASH that displays a text stream showing one lexical bundle, of three to five words, at a time with the feature of speed control for display time. Speed reader that plays pre-slashed lexical bundles in the fashion of Spreeder.

**Software as a Service (SaaS):** Software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet.

**SPA:** Single page application. A highly responsive web application that fits on a single page and does not reload as the web page changes states.

**Spreeder:** Speed reading tool [www.spreeder.com](http://www.spreeder.com)

**TOEFL:** English language proficiency test required by universities for enrollment for internationally based students.

**Token:** Text that has been processed into individual words by the Document Processor

**Ubuntu:** a Debian-based Linux operating system

**VM:** Virtual Machine.

*(This Space Intentionally Left Blank)*

**1.4 References**

Anderson, N. (1999, April 1). Improving Reading Speed - Activities for the Classroom. Retrieved February 1, 2015, from http://dosfan.lib.uic.edu/usia/E-USIA/forum/vols/vol37/no2/p2.htm

Engelbrecht, K. (2003, June 18). The Impact of Color on Learning. Retrieved February 25, 2015, from http://sdpl.coe.uga.edu/HTML/W305.pdf

English Proficiency. (2015, February 2). Retrieved February 6, 2015, from https://www.odu.edu/content/odu/admission/proficiency.html

Hoffman, D. (n.d.). Academictips.org - Reading and Highlighting Tips. Retrieved February 25, 2015, from http://www.academictips.org/acad/literature/readingandhighlighting.html

Improve Reading Speed and Comprehension. (2006, January 1). Retrieved February 25, 2015, from http://spreeder.com/

McKeon, D. (n.d.). Research Talking Points on English Language Learners. Retrieved December 11, 2014.

Mikowski, M., & Powell, J. Single Page Applications. Manning Publications 2014.

Monarch Diversity. (n.d.). Retrieved February 6, 2015, from https://www.odu.edu/admission/international/global

Nishida, H. (2013). The Influence of Chunking on Reading Comprehension: Investigating the Acquisition of Chunking Skill. *THE JOURNAL OF ASIA TEFL, Vol.10*(No. 4), Pp. 163-183.

Raver-Lampman, Greg. (2014. August). Personal Interview.

The Condition of Education 2014. (2014, January 1). Retrieved February 6, 2015, from http://nces.ed.gov/fastfacts/display.asp?id=96

Tremblay, A., Derwing, B., Libben, G., & Westbury, C. (2011, January 15). Processing Advantages of Lexical Bundles: Evidence From Self-Paced Reading and Sentence Recall Tasks. Retrieved December 10, 2014.

Lab 1 CS 410 Team Blue

Lab II – CLASH Specification Outline

**1.5 Overview**

This product specification provides the hardware and software configuration, capabilities and features of the CLASH prototype. The information provided in the remaining sections of this document includes a detailed description of the hardware, and software architecture of the CLASH Prototype; the key features of the prototype; and the parameters that will be used to control, manage or establish that feature.

*(This Space Intentionally Left Blank)*

1. **GENERAL DESCRIPTION**

The CLASH prototype will be as close as possible to the real world product, however there are some aspects of the prototype that will require changes to demonstrate the proof of concept.. The product will not be built on a physical server but on a virtual machine that will act as both the web server and database server.

The prototype will not include any “homework” modes, in which students are tested on their ability to identify the parts of speech in the text. Copy and paste into a text window will be the input that is to be parsed with the real world product having this ability and the ability to take text directly from a multitude of text files such as .txt, .doc., and possibly other formats as well.

The prototype will not include actual student data and the database will be populated with generic student data so that features such as authentication for user accounts and permissions to documents can be tested. Text documents used for testing and demonstration purposes can be any text that can be copied into the text box. A table featuring the prototype capabilities is displayed on the next page.

*(This Space Intentionally Left Blank)*

|  |  |  |
| --- | --- | --- |
| **Features** | **Real World Project** | **Prototype** |
| Parsing Capabilities | Ability to Parse different kinds of documents | Ability to parse text copy and pasted into form |
| Text Modification | Ability to modify and store previously parsed documents | Ability to modify and store previously parsed documents |
| Color Capabilities | Ability to Color chosen parts of speech using a JSON format and javascript functions. | Ability to Color chosen parts of speech using a JSON format and javascript functions. |
| Slashing Capabilities | Ability to identify lexical bundles through the inserting of slashes. | Ability to identify lexical bundles through the inserting of slashes. |
| Displaying lexical bundles in a single bundle form | Ability to speed up, slow down and pause lexical bundles being displayed. | Ability to speed up, slow down and pause lexical bundles being displayed. |
| Exception list | Lists of commonly used expressions that would otherwise be incorrectly parsed and tagged. | Lists of commonly used expressions that would otherwise be incorrectly parsed and tagged. |
| Login interface | User Authentication in a stand alone environment | User Authentication in a stand alone environment |

*Table 1. Real world VS Prototype Features*

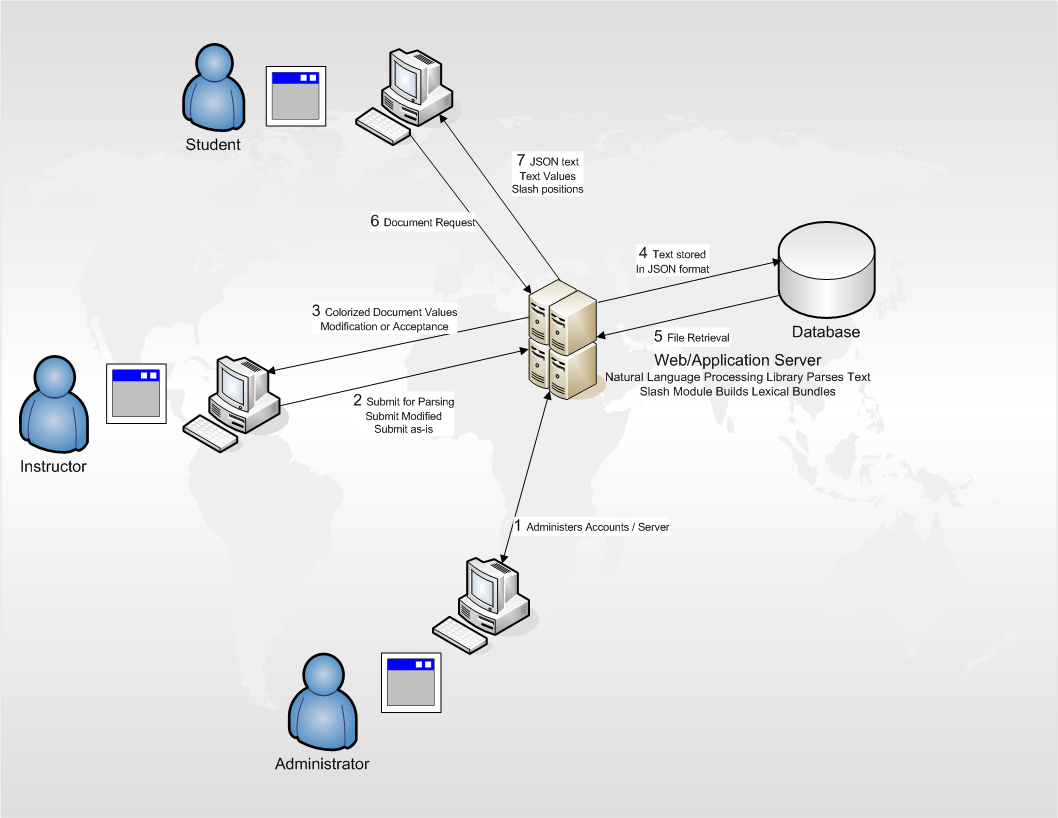
| **Features** | **Real World Project** | **Prototype** |
| --- | --- | --- |
| Student Data reporting | Tracks individual and collective student progress. To include lexical bundles per minute, total time and total lexical bundles. Data to be stored in database. Displayed in graphs and statistics. | Not included. |
| Homework Mode | Instructors have the ability to remove coloring of words and have students correctly identify the part of speech. | Not Included. |
| Administrative Privileges | Administrators are able to edit, add, or remove anything in the system. | Administrators are able to edit, add, or remove anything in the system. |
| Print mode | Ability to print documents with slashes inserted. | Ability to print documents with slashes inserted. |

*Table 1. Real world VS Prototype Features*

*(This Space Intentionally Left Blank)*

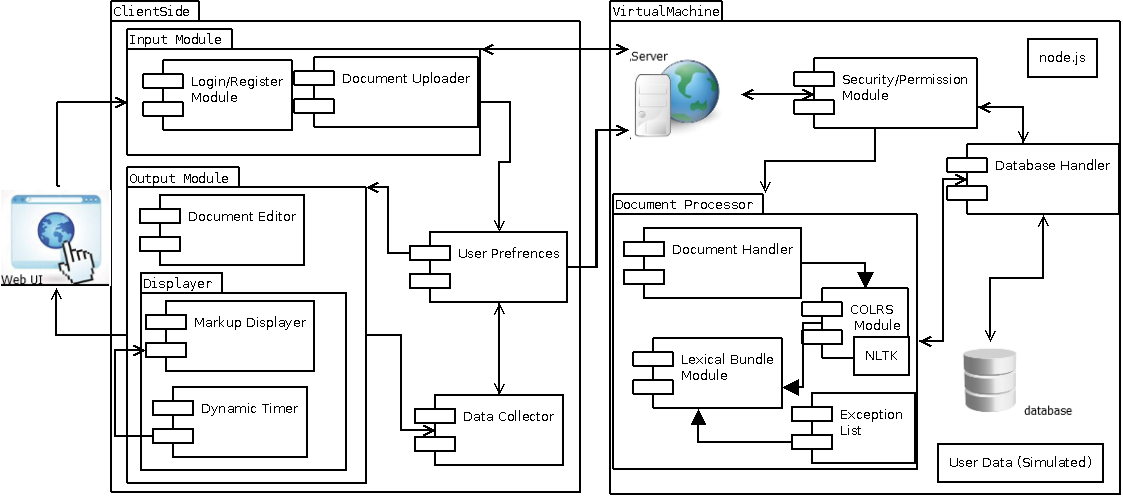
## Prototype Architecture Description

The CLASH prototype will be hosted on a Virtual Machine hosted by Old Dominion Universities Computer Science Department. All server hardware functionality will be emulated on the VM with the web server and database server located on that same “machine”. The operating system running on the VM will be Ubuntu 14.04 LTS. The web server and application server will be Node.js. The web server will host the application at the URL [http://esl-clash.cs.odu.edu](http://esl-clash.cs.odu.edu/) and will allow the users to interact with the single page application with functionality according to their permission level. The figure on the next page displays the process flow

.

*Figure 1. Process Flow*

## Prototype Functional Description



*Figure 2. Major Functional Component Diagram*

The Major Functional Component Diagram displays the major components of the prototype. The server side of the program contains the “engine” of the product. The COLRS module receives text when submitted and parses the text into parts of speech using a natural language tool kit. The output is stored in a JavaScript object notation file. This result is then handled by the lexical Bundle Module where using the exception list and the slashing algorithm, slashes are inserted into the text to differentiate the lexical bundles. The slashing information is also added to the JSON file. The text is displayed on the client side for approval by the user and then saved to a database. The database is a relational database that holds records of user accounts, class records of users, documents and data collected from users accessing parsed documents.

Users will access the program thru the use of a web browser and the program URL. Users select the required document from a drop down menu, the server will return the document and the client side will display the results. The client side will utilize JavaScript and the JSON file to associate the part of speech with a predetermined color to color words as well as inserting slashes into the text. Users can selected or unselect any part of speech or slash options. Additionally there is a SLASH playback module that allows the text to be displayed one at a time. Users can increase and decrease the speed of bundles being displayed, and can also pause as well.

*(This Space Intentionally Left Blank)*

* 1. **External Interfaces**

The prototype will not interact with any external systems.

* + 1. **Hardware Interfaces**

The prototype will not have separate servers for database and the application as all of the server operations will be located on the Virtual Machine. For the client to access the program the user must have an internet connection.

* + 1. **Software Interfaces**

The application will interact with a relational database for user access/management and document storage and retrieval. For parsing the text the application will run a python script using the NLTK library.

* + 1. **User Interfaces**

The user will interact with the CLASH application using a standard Personal Computer. A keyboard for text input, a monitor for displaying text output and mouse for page navigation is required.

*(This Space Intentionally Left Blank)*

1. College Enrollment by Sex, Age, Race, and Hispanic Origin: 1980 to 2009." U.S. Census Bureau [↑](#footnote-ref-1)
2. Research Talking Points on English Language Learners [↑](#footnote-ref-2)