Lab 2 – CLASH Prototype Product Specification

Blue/Purple Teams

Old Dominion University

Author: Ali Albattah

CS411W

Janet Brunelle & Hill Price

Version 1

March 24, 2015

Table of Contents

[**1. Introduction** 2](#_Toc415417645)

[**1.1 Purpose** 2](#_Toc415417648)

[**1.2 Scope** 3](#_Toc415417649)

[**1.3 Definitions, Acronyms and Abbreviations** 5](#_Toc415417654)

[**1.4 References** 7](#_Toc415417655)

[**1.5 Overview** 8](#_Toc415417656)

[**2. General Description** 8](#_Toc415417658)

[**2.1 Prototype Architecture Description** 8](#_Toc415417660)

[**2.2 Prototype Functional Description** 10](#_Toc415417662)

**2.3 External Interfaces …………..………………………………………………..…………....**11

[**2.3.1**](#_Toc415417663) **Hardware Interfaces ……….……………….…………………………..…………......**11

**2.3.2 Software Interfaces ……….……………….…………………………..………….....**12

**2.3.3 User Interfaces …….……………..….…………………………….…...………..…...**12

List of Figures

[**Figure 1. Prototype Hardware/Software Architecture** 9](#_Toc415002301)

**Figure 2. Major Functional Component Diagram** …….……………….……………….....…10

[**Figure 3. CLASH Process Flow** 11](#_Toc415002302)

List of Tables

[**Table 1. Real World Product vs Prototype Continued** 5](#_Toc415417247)

[**Table 2. User Interface Views** 13](#_Toc415417248)

1. **Introduction**

COLRS and Slash are the modules, which make up CLASH. COLRS helps users in better understanding grammar and sentence structure by highlighting particular components of the text, while the Slash module helps users comprehend and read more fluently by grouping words into lexical bundles, which provides the perfect combination of tools to aid English as a second language (ESL) students on their academic journey.

The processes used to teach reading and grammar to ESL students has been outdated for over a decade. In 2001 only 18.7 percent of ESL students were successful in accomplishing normal or above normal comprehension level. In 2004 an estimate of 15 percent out of nearly 5 million ESL students had no resources to help aid them in achieving a greater comprehension level of the English language and in February of 2001 approximately 4 times as many ESL students dropped out of school compared to native English students. Also 10 percent of ESL students between 7-12 grades were held back (McKeon). However, ESL student must pass Test of English as a Foreign Language (TOEFL) or International benchmark test (IBT) with the required score by Old Dominion University in less than a year and half to be a full time college student. Or gets enrolled in the bridge program in case which the student get less than the required score.

* 1. **Purpose**

CLASH is the answer to help ESL understudies learn English less demanding and quicker than customary means, but will also be an electronic based program that will enhance perusing velocity and understanding. It will have a Graphic User Interface for cooperation with the understudy and the database; also permit teachers to screen the utilization of the program by every student. The CLASH application comprises of two modules Slash and COLRS.

Slash’s straightforward program interface provides ESL students an easy to follow and understand program. Slash will enhance perusing velocities of ESL understudies, separate texts into lexical packs for understudies to process a section easier, it will likewise significantly enhance perusing perception and convert pursuers from word to word pursuers into lexical pack pursuers and last, but not least permit the conversion of physical text to electronic text for recreational perusing.

The COLRS module helps understudies recognize parts of discourse through the colorizing of its text. For future updates on the product, focus will be dedicated to the 8 customary parts of discourse, which could include more nuanced parts of discourse. Colorization of content will be carried out through JavaScript control of HTML content from records beforehand parsed and a content stream that was taken from the record.

**1.2 Scope**

CLASH provides students with an exceptional and easy learning experience through its web based interface. By using lexical bundles students are able to control the speed that the text is being shown and also only view certain parts of the text.

CLASH provides administrators, instructors and students each with their own individual log in information. Students are able to control reading speeds and document viewing options by using SLASH PLAYBACK. Instructors are able to add/ remove students, add/edit documents and monitor student activity. The administrators have the same abilities as the instructors, but can also add and remove them.

CLASH allows you to view both parts of speech and lexical bundles, thus allowing students to better improve their reading speed and comprehension. CLASH also has a feature that highlights certain parts of the text to identify it easier. Instructors can also ensure that the text is accurate and allow them to adjust their lectures by reviewing the students’ progress. Being that CLASH is specifically designed for ESL students it is the most optimal method for students to learn from.

A feature that makes CLASH unique is that it both of the required functions such as displaying the parts of speech and lexical bundles. This provides the students with the ability to improve their reading comprehension by having a speed reading application that does not break up the individual lexical bundle. Another ability found within the CLASH is the ability to pause and change the display speed. CLASH will include a text parser this enables the program of the ability to automatically identify parts of speech, and color them respectively, allowing for easy identification. It will also allow instructor to review the parsed text, to ensure accuracy. Another feature that CLASH will have is the ability to save usage data, allowing instructors to display the student progress, and fine tune their lectures accordingly. CLASH is the first web-based speed reader designed specifically for use in ESL instruction. Instructors will be able to have documents available to the students at their specified level. With the required focus by ESL students, CLASH will give students and instructors enormous help to increase reading speed and comprehension.

*(This Space Intentionally Left Blank)*

|  |  |  |
| --- | --- | --- |
| **Features** | **Real World Product** | **Prototype** |
| Parsing Capabilities | Ability to Parse different kinds of documents | Ability to parse text copy and pasted in text block |
| 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 handled by the SLASH Algorithm. | Lists of commonly used expressions that would otherwise be incorrectly handled by the SLASH Algorithm. |
| Login interface | User Authentication in a stand-alone environment | User Authentication in a stand-alone environment |
| Student Data Reporting | Tracks individual and collective student progress. To include words per minute, total time and total Lexical Bundles. Data to be stored in database. Displayed in graphs and statistics. | Limited basic student metrics will be available such as Lexical Bundles per Minute. |
| 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 users and saved documents in the system. | Administrators are able to edit, add, or remove users and saved documents in the system. |
| SLASH Document Viewing Mode | Ability to view documents with slashes inserted and SLASH Handler. | Ability to view documents with slashes inserted and SLASH Handler. |

Table 1. Real World Product vs Prototype Continued

* 1. **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.

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.

ESL: English as second language.

ELL: English Language Learner.

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

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.

Lexical Bundle: A group of words that occur repeatedly together within the same register

MFCD: Major Functional Component Diagram.

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.

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

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

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.

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

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.

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

VM: Virtual Machine.

*(This Space Intentionally Left Blank)*

* 1. **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 Purple

Lab II – CLASH Specification Outline

* 1. **Overview**

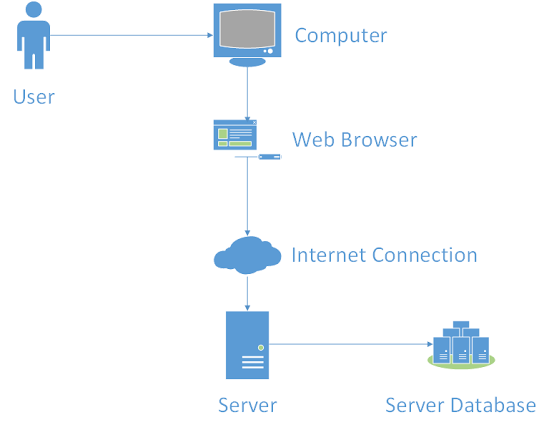
This product specification provides the hardware and software configuration, capabilities and features of the CLASH prototype. The information provided in the remaining sections of this document includes a detailed description of the hardware, and software architecture of the CLASH Prototype; the key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

**2.** **General Description**

CLASH will display the ability to identify specific parts of speech by, dashing, coloring, and slashing the text in a lexical bundle stream, with the capability to speed up, slow down, or pause the word stream. In addition, the functional success of CLASH will be determined by the accuracy of the core features and the integration of team-built modules. Determining the risks of the involvement of the mentor will be mitigated by the development of the product for its ease of use, and accuracy. A well-built program with these core features and functions will result in the authentic use of CLASH against a control group. With these tasks being completed, this confirms the ability to demonstrate the feasibility of the system’s achievement; which is, the utility of CLASH in an academic environment.

## **2.1 Prototype Architecture Description**

Initially CLASH will require a laptop from the developers, but will eventually be migrated with a virtual machine (VM) hosted by the ODU Computer Science Department. CLASH will also require a variety ofopen source or liberally licensed software to permit distribution or potential commercialization and will also be consist of a series of packages installed under Ubuntu 14.04 LTS. The Node.js will provide the web/application server and allow incoming connections over the server and process text documents. It will also interact with the MySQL database server, which runs the same machine and Natural Language Toolkit. The SPA built into CLASH's Node.js server will create the web pages for the users by integrating custom programming which will parse out the input text via NLTK generate a markup stream return that stream to allow the users browser to render the text as described elsewhere in this document.



#### Figure 1. Prototype Hardware/Software Architecture

*(This Space Intentionally Left Blank)*

## **2.2** Prototype Functional Description

*Figure 2. Major Functional Component Diagram*

**Overview**: CLASH can be easily accessed with no extra equipment except for internet and an internet enabled device. With CLASH, users will have access to 2 major area: Server side and Client side, as shown in Figure 2.

**Server-side:** COLRS Module use open source Natural Language Processing (NLP), and Natural Language Tool Kit (NLTK) to tokenize and parse the input document, and it will generalize the Parts of Speech (POS) tags in more readable format. Each token is then tagged with a POS using the open source Natural Language Processing (NLP) library, Natural Language Tool Kit (NLTK). The output is stored in a JavaScript object notation (JSON) file.

Lexical Bundle Module will accept tokenized stream from COLRS Module and decide how to break sentences into digestible lexical bundles. There will be a sub-component to deal with the exception list and the slashing algorithm that inserted into the text to differentiate the lexical bundles, which improves the accuracy of the final output.

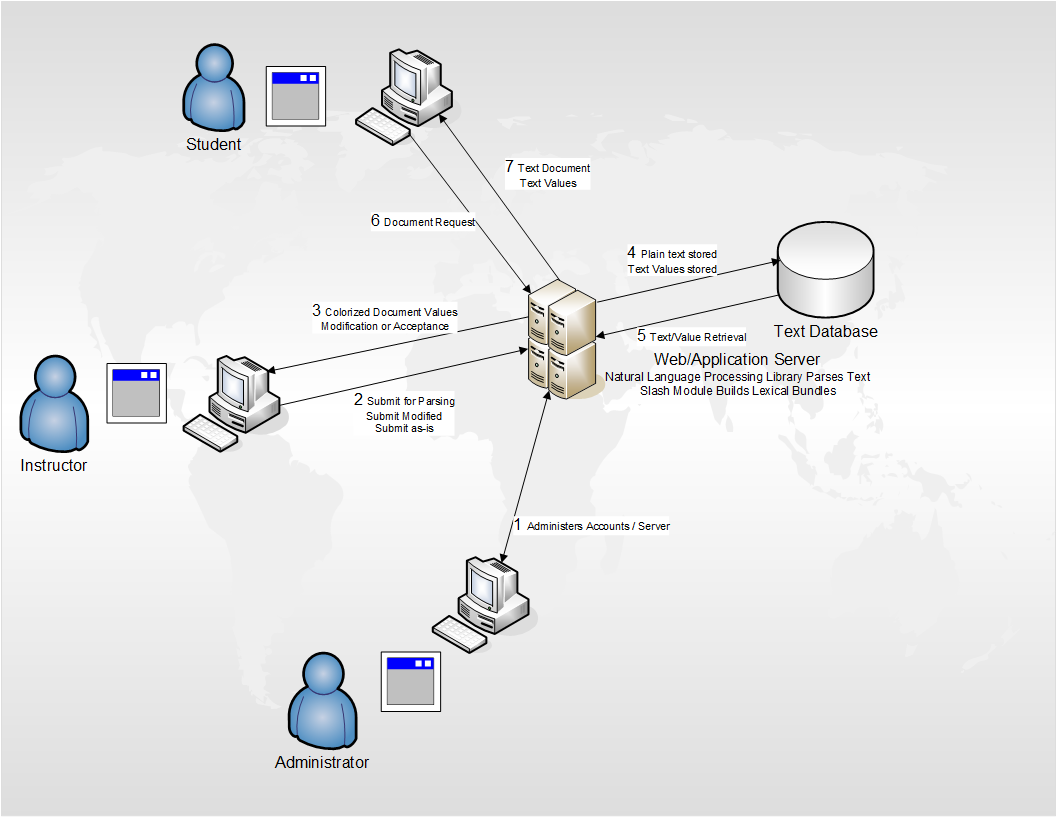


Figure 3. CLASH Process Flow

**Client-side:** The reader program will receive data from the server and parse the text according to their POS tag. There will be two display modes, color mode and speed mode. Under color mode, user can choose what POS tag to be colorized, and choose whether the Slash need to be displayed. User can choose the reading speed under the speed mode, and the reader program will display each lexical bundle according to the reading speed.

**2.3 External Interfaces**

The prototype will not interact with any external systems.

**2.3.1 Hardware Interfaces**

Initially the program will require a laptop from the developers, but will eventually be migrated with a virtual machine (VM) once received by the ODU Computer Science Department.

*(This Space Intentionally Left Blank)*

**2.3.2** **Software Interfaces**

The program will require a variety ofopen source or liberally licensed software to permit distribution or potential commercialization and will also be consist of a series of packages installed under Ubuntu 14.04 LTS. The Node.js will provide the web/application server and allow incoming connections over the server and process text documents. It will also interact with the MySQL database server, which runs the same machine and Natural Language Toolkit. The SPA built into CLASH's Node.js server will create the web pages for the users by integrating custom programming which will parse out the input text via NLTK generate a markup stream return that stream to allow the users browser to render the text as described elsewhere in this document.

**2.3.3 User Interfaces**

The user will interact with the CLASH web service using a standard Personal Computer. That will include a keyboard for text input, a monitor for displaying text output and mouse for page navigation is required. Table 2 provides an overview of the available views in the user interface and the user type that has access to each view.

*(This Space Intentionally Left Blank)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Views | Brief Description of View | Administrator | Instructor | Student |
| Logon Screen | Landing Page and initial logon screen. | X | X | X |
| COLRS Module | Colorizes the eight essential POS within the document and displays the output text in the original sentence and paragraph structure. | X | X | X |
| SLASH Handler | A playback environment that displays a selected document in Lexical Bundles one at a time. | X | X | X |
| Document Management | Add, Delete, Edit documents stored in the database | X | X |  |
| Document Editor | Set the POS or boundaries of the lexical bundles within a document. | X | X |  |
| User Management | Add, Delete, Edit user accounts stored in the database | X | X |  |

Table 2. User Interface Views