Lab 1 – CLASH Product Description

Justin Bennett

CS411

Professor Brunelle, Professor Price

February 28, 2015

Version 2

Table of Contents

1. INTRODUCTION 3
2. PRODUCT DESCRIPTION 5
2.1 Key Product Features and Capabilities 7
2.2 Major Hardware and Software 9
3. IDENTIFICATION OF CASE STUDY 10
4. CLASH PRODUCT PROTOTYPE DESCRITPTION 11
4.1 Hardware and Software Prototype Architecture 14
4.2 Prototype Features and Capabilities 16
4.3 Prototype Development Challenges 17
Glossary 18
References 20

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

1. 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], 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], this fact emphasizes the need for additional teaching resources.

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

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

attention are in short supply. 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. In addition the CLASH program will include a mode that will display lexical bundles, groups of words that are frequently grouped together to help shape meaning and coherence in text. CLASH will improve both reading speed and comprehension.

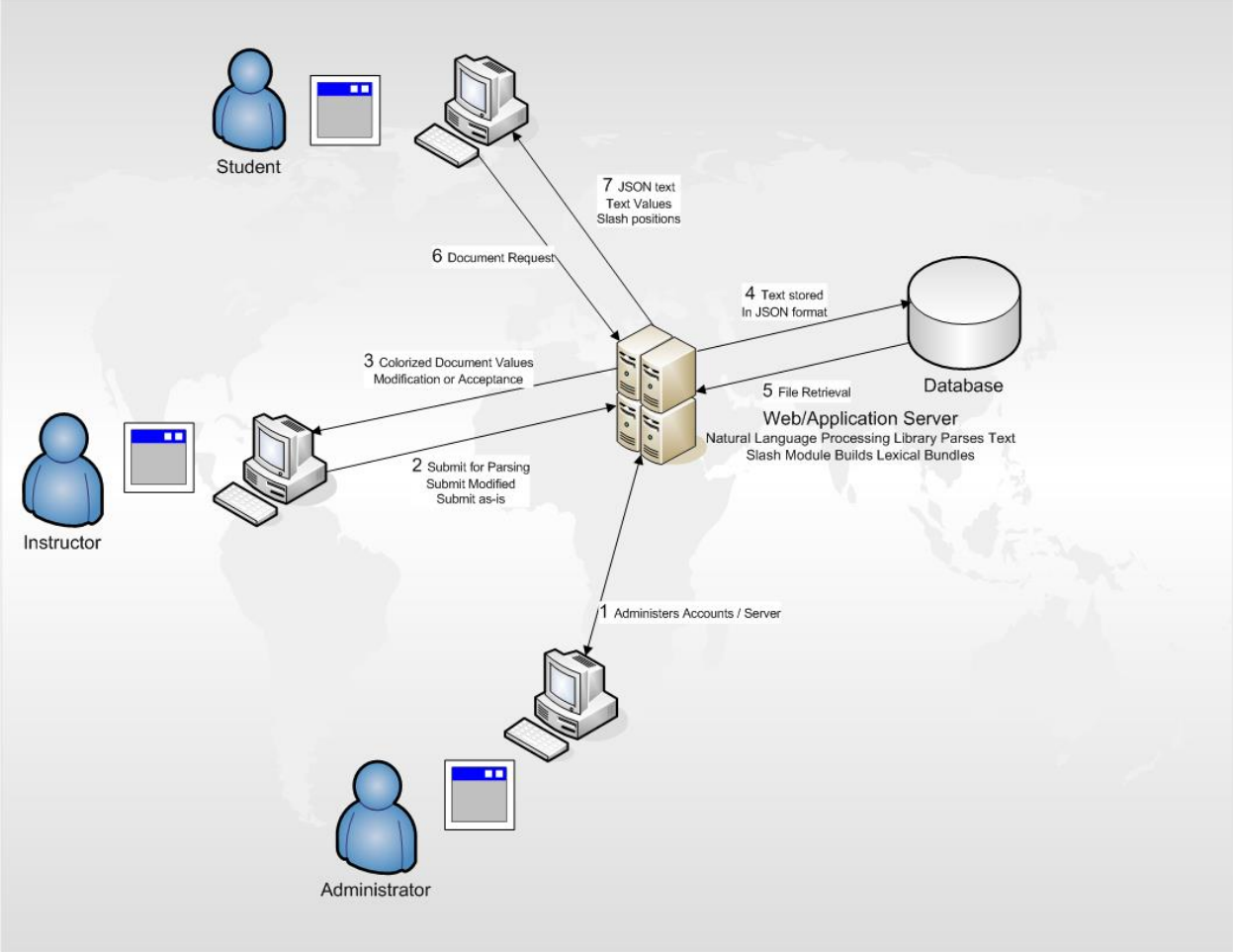
(This Space Intentionally Left Blank)

2. Product Description

CLASH will be a web browser application that will aid in the comprehension and reading speed of ELLs. The interface will be intuitive with buttons to upload documents for parsing and storage as well as a drop down menu for quick access to available documents. The program consists of two different modules, a module that colors different parts of speech with optional slashes inserted and a second that will display the lexical bundles one at a time, with the user having the ability to control the display.

The COLRS module will assist students in identifying parts of speech through the colorizing of text. The eight traditional parts of speech, less interjection, each will be displayed with a unique color that can be toggled off if the user wants focus on a certain part of speech Future updates will add more nuanced parts of speech for more advanced users. The SLASH Reader module will use an algorithm to insert slashes to break text into easily understood and readable chunks. These chunks are then available to be displayed on the web browser display that has the ability to be user controlled.

The application allows the instructor to submit a document through the browser interface that will be passed to the web/application server. The server will parse the document and return the text with a formatted file that includes part of speech coloring and slash insertions. The submitter can adjust the results if there are errors. Upon submitter approval of the text, the document is the resent to the server where it will be stored in the database for future retrievals by student users. This is process is illustrated in Figure 1, Improved Process.



(This Space Intentionally Left Blank)

2.1 Key Product Features and Capabilities

As described in section 2, CLASH will use a web browser based solution to accomplish the intended goals. Each module focuses on either increasing comprehension, recognition of parts of speech and reading speed. All functionality is located on the same page with modules appearing fluidly depending on user selection.

The key feature for the COLRS module is ability to color the parts of speech that the user will select. Selection of the parts of speech will be handled by HTML check boxes, that will each have an assigned universal color. The ability to display the slashes can also be activated by a check box. Individual check boxes can be unchecked to focus on areas of difficulty.

The key feature for the SLASH Reader module is that of the lexical bundle display. The display will be controlled by a user interface with familiar controls such as play, stop, rewind, to slow pace, and fast forward, to quicken the pace of words being displayed as the user reads. The product database will include a exception list of commonly grouped words and words such as abbreviations that would be incorrectly parsed by the processing software due to their context in the text.

Tthe system will be capable of handling user accounts, document and CLASH file format storage and the ability to print a document with the slashes inserted so that a physical copy may be distributed. User accounts will have different capabilities and permissions depending on the user role. Administrators will have the ability to create, delete and set permissions for user accounts, as well as all capabilities afforded to the instructor and user accounts. Instructors will have the ability to enter text into the program for parsing of parts of speech and inserting of

slashes, along with modifications to the output from the system should the Instructor wish to change a part of speech tag or slash insertion. Instructors also can create user account groups that reflect the students in a course. This will allow for instructors to limit the students' access to the documents for a specific class. Instructors will have all capabilities afforded to student accounts. Student users will have the capability to log in and select documents that they have access to, using either the COLRS or SLASH reader module.

(This Space Intentionally Left Blank)

2.2 Major Hardware and Software

The system consists of two major hardware components. A computer that the user will use for uploading, manipulation and display of the documents and a server that will host the web application page, as well the database for storage of the data required. Users are also required to have an internet connection.

The system will require installation of node.js. Node.js is an open-source run-time environment that will be used for developing the single page application, interacting with the JavaScript built modules that will provide the client side functionality of displaying HTML manipulation of the text. The system also requires a database that will use the traditional relational based model of data storage. A web server to serve the application that is friendly to Node.js is needed as well.

The ability to parse the text and identify the parts of speech is done by using natural language processing software. The product will use an open-source software package, the Natural Language Tool Kit. The software is based in python and will be ran on the server-side of the application.

(This Space Intentionally Left Blank)

3. IDENTIFICATION OF CASE STUDY

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.

Students come to ODU's English learning center with different proficiency in speaking English; therefor it is necessary to tailor to the students' needs. The CLASH program allows the student to alleviate some of the difficulties with color to parts of speech mapping and displaying of lexical bundles at a comfortable pace. By allowing the user to choose which parts of speech are displayed and/or slashed the user is less likely to be overwhelmed and frustrated by too much information.

(This Space Intentionally Left Blank)

4. CLASH Product Prototype Description

The CLASH prototype will be as close as possible to the real world product. 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 have all the functionality that was described in Section 2.1, in addition, the real world product will have the ability to track student data. Items of interest to the instructors include lexical bundles per minute, total time spent on a document, and total of lexical bundles read. Production versions of the software will chart these characteristics of students in graphical form and advanced statistics so that progress can be tracked and compared.

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.

(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

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)

4.1 Hardware and Software Prototype Architecture

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 and will allow the users to interact with the single page application with functionality according to their permission level. The Natural Language Toolkit will be the natural language processing software that the prototype will utilize. The process of text parsing will be done by a python script that will run when text is uploaded to the application server. On the client side, when a document is selected a canned SQL query will be sent to the application server and the appropriate text will be returned with a JSON format output. These two components will be utilized in the JavaScript dependent on the module selected. The JSON data holds the text to be displayed along with each words tagged part of speech that is taken from the python script. Each paragraph in the text will be an array, with each sentence also in array form. Individual words will have a have a key and value relationship. Currently there is a "word" key with the actual word representing the key, a "tagged" key representing the part of speech tag, a "mod" key that indicates if the tag has been changed and finally a "slashed" key with a Boolean value that indicates if a "slash" should appear after the word. The Product Prototype system architecture is described in Figure 2 on the following page.

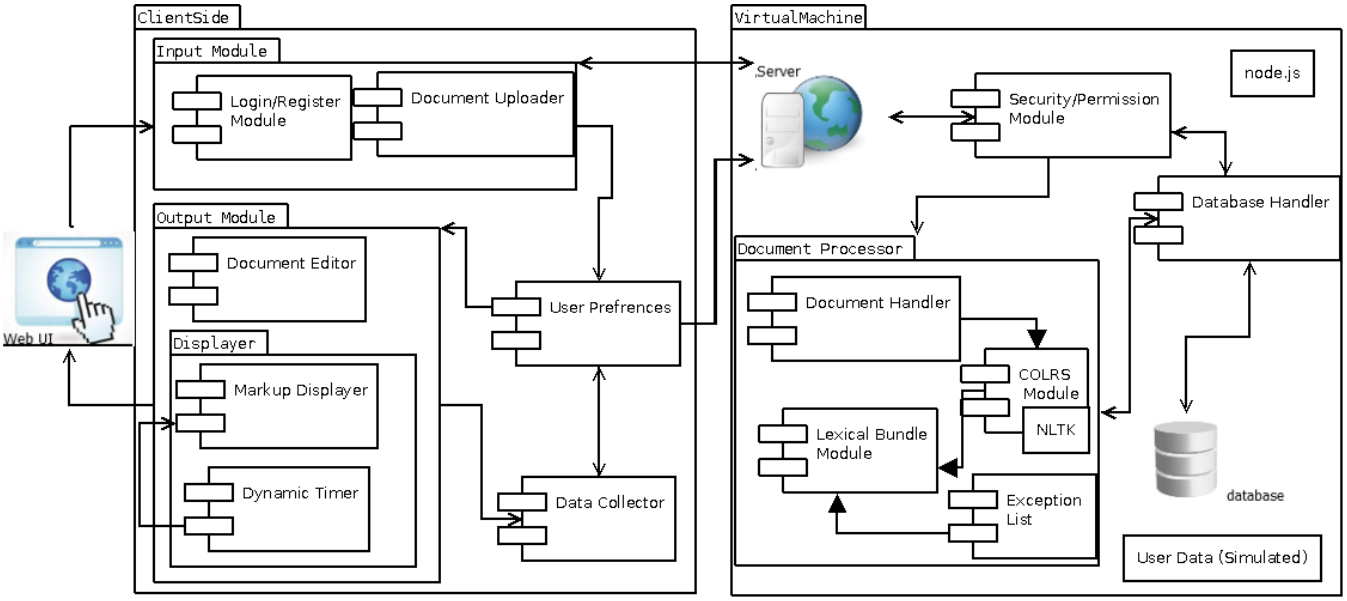


Figure 2. Prototype Major Function Component Diagram

(This Space Intentionally Left Blank)

4.2 Prototype Features and Capabilities

The CLASH program prototype will demonstrate the ability to identify specific parts of speech by way of coloring the text and will also identify lexical bundles through the inserting of slashes. The program prototype will only have the ability parse text that is copy/pasted into the browser. A risk associated with the parsing of parts of speech and slash insertion is the accuracy of said parsing and insertions. Inaccurate results are detrimental to learning and serve will cause users to abandon the software. This risk can be mitigated through the ability for the Instructor user to modify and correct any errors before student users can use the document.

The prototype must be easy to use for all user roles else risk the user abandoning the software. The user must be able to look at the prototype, quickly find and use the desired function. This risk is mitigated by having a simple and easily understood User Interface. Buttons must clearly identify their functions, such as switching modules, uploading text and selecting desired parts of speech and slash insertion. By keeping the prototype simple and easy to use the user is more likely to stay engaged with the software.

Completing these proofs of concept, the accuracy of the parsing and identification of lexical bundles, the program prototype will demonstrate the feasibility and utility of such a system in an academic environment. With the proof of concepts proven a more complete program can be built in the future.

(This Space Intentionally Left Blank)

4.3 Prototype Development Challenges

When building an application there will always be challenges that are either unforeseen or underestimated and this project is no different. Some of the systems and environments are unfamiliar to the team. The single page application is programming environment that is not taught at the university and the team has had to take the initiative to learn the components that will be needed for the project. To alleviate this issue, documentation has been distributed to the team to help in bridging the knowledge gap.

Another challenge of building this prototype is the ability to integrate another team into the project. Two separate views of the prototype were at odds when the teams were merged. As such there were disagreements with design decisions due to differing knowledge bases and the desire to use familiar programming paradigms. This is mitigated by communication among team members and achieving a clear vision of the product decided by and held by all members.

(This Space Intentionally Left Blank)

Glossary

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.

GUI: Graphic User Interface

HTML: HyperText 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 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.

Spreeder: Speed reading tool; www.spreeder.com

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.

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)

References

"College Enrollment by Sex, Age, Race, and Hispanic Origin: 1980 to 2009." U.S. Census Bureau, 1 Jan. 2012. Web. 2 Feb. 2015. <http://www.census.gov/compendia/statab/2012/tables/12s0282.pdf>.

McKeon, Denise. "Research Talking Points on English Language Learners." National Education Association, 1 June 2005. Web. 2 Feb. 2015. <http://www.nea.org/home/13598.htm>

Choudaha, R., and L. Chang. "Trends in International Student Mobility." World Education Services, 1 Feb. 2012. <http://www.wes.org/ras/TrendsInInternationalStudentMobility.pdf>

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.

Educational Testing Service. (2015). About the TOEFL iBT® Test. Retrieved February 7, 2015, from ETS.org: <http://www.ets.org/toefl/ibt/about?WT.ac=toeflhome_ibtabout2_121127>

Hanel, S. D. (2011, March 5). Lexical Bundles. Retrieved February 7, 2015, from Communicating in English - An Internet-based, English Language Resource : <http://sdhanel.com/corpuslinguitics/lexicalbundles.html>

Haynie, D. (2014, Novemeber 14). Number of International College Students Continues to Climb. Retrieved February 6, 2015, from US News: http://www.usnews.com/education/best-colleges/articles/2014/11/17/number-of-international-college-students-continues-to-climb

Institute for International Education. (2014). Intensive English Programs: Leading Places of Origin. Retrieved February 7, 2015, from Institute for International Education: http://www.iie.org/Research-and-Publications/Open-Doors/Data/Intensive-English-Programs/Leading-Places-of-Origin/2012-13

Institute of International Education. (2014). Open Doors Fact Sheet: Virginia. IIE with support from the U.S. Department of State's Bureau of Educational and Cultural Affairs.

Open Doors. (2014). Open Doors 2014: A 15-Year Snapshot. Washington DC: Institute of International Education.

Powell, M. &. (2014). Single Page Web Applications. Shelter Island: Manning Publications Co.

(This Space Intentionally Left Blank)