CLASH Product Description CS410 Lab 1

Blue/Purple Teams

Ming Jiang

Old Dominion University

**Table of Contents**

1 Intro .......................................................................................................... 3

2 Product Description ......................................................................................4

2.1 Key Product Features and Capabilities ........................................................4

2.2 Major Hardware and Software Components ................................................5

3 Identification of Case Study ..........................................................................7

4 C.L.A.S.H. Product Prototype Description ………........................................10

4.1 Hardware and Software Prototype Architecture ….......................................12

4.2 Prototype Features and Capabilities ............................................................13

4.3 Prototype Development Challenges ……………………...............................13

Glossary ...................................................................................................................14

References ..............................................................................................................15

# **Intro**

Color Lexical Analysis algorithm and Slash Handler, or C.L.A.S.H., is a web-based application that aim to improve the reading speed and reading comprehension of English as Second Language (ESL) students. The CLASH includes two basic modules, “COLRS” and “Slash.” The COLRS module process the document by identify part of speech (POS) in each sentence and colorize different POS in different colors. The Slash module process the document to identify lexical bundles, or thought groups, in the sentence by scan through the POS tags generated by COLRS module and insert slash in between lexical bundles.

According to National Education Association, there is about 5 million English Language Learner in the 2003-2004 year, which represent over 10 percent of total public school students. Among those numbers, 15% of those students did not have any special programs designed to help them learn English, and only 19.7 percent of ESL students were assessed as being at or above average in 2001. During the same year, almost 10 percent of ESL student in grade 7-12 were retained, and the dropout rates of ESL students was four times that of native English-speaking students (McKeon). The problem also existed throughout universities, even with an entire ESL department to help the international students, the lack of software and outdated teaching method had resulted inefficient ESL courses.

The current teaching method is simple. Professor display some articles on the screen, and mark the part of speech on each sentence. This process is very time consuming, since the professor need to manually mark each essay to identify the part of speech, and it can also limit the amount of resources a student can access. With COLRS module, professor can submit an article, wait for the colorized output with all the part of speech marked, and make necessary change on part of speech before share with the students. Students on the other hand, can download and read the marked document from their home, which can save class time for other exercises.

# **2 Product Description**

CLASH is a software that aim to help improve the instructors’ teaching efficiency and students’ reading performance. CLASH is a web-based software that can be access from any computer with the Internet connection, it will be a virtual classroom for instructors to provide reading materials to students. CLASH will also present a Graphic User Interface for users to interactive with CLASH server.

The product will reduce the time instructor need to manually mark each documents, and it can help student to access the material in after-class environment. It will also provide lexical bundles reading experience for users to improve their reading performance, and help to turn reader from word by word reader into lexical bundle reader. The product will also provide tracking functionality to instructor that can track students’ reading speed and activities in a timely manner, which can help identify students’ participation and improvement throughout the course period.

## 2.1 Key Product Features and Capabilities

CLASH will be a web-based application that consist of multiply modules. It aims to improve teaching efficiency and a better learning experience for ESL students.

Login / Authentication:

Users will be allow to login the website. Instructor user will have more functionalities like upload document and edit document.

Download Document:

Users will be allow to download and read documents with proper permissions.

Upload Document:

Instructor users will be allow to upload and process the document, the result document will be colorized with part of speech and slashed into lexical bundles.

Edit Document:

Instructor users will be allow to retrieve parsed document and edit mistaken part of speech or slashes accordingly.

Read document with COLRS:

Users will be allow to read document in COLRS module that display selected POS with different colors.

Read document with Slash:

Users will be allow to read document in Slash module that display sentence in lexical bundles. Users can set the display rate and adjust it according to their own pace.

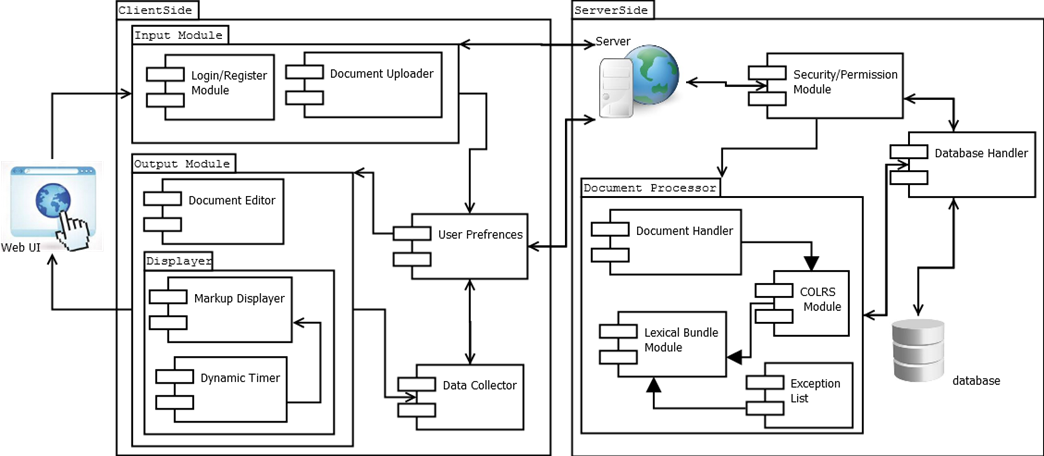
Reading statistic:

Students will be allow to view their own reading performance report, instructors will be allow to view their students’ reading performance reports.

Add / Remove User:

Instructors will be allow to add or remove student in their course, and share the documents with the class.

## 2.2 Major Hardware and Software Components



Major Functional Component Diagram

**Overview.** The CLASH will be a web-based application that hosted on the server, and can be easily access through web browser. There is no special hardware requirement other than an active server and a database on the server end, and an internet-enabled device for the client to access the application. There are three major software components for CLASH, COLRS Module, Slash Module, and Client-side Reader.

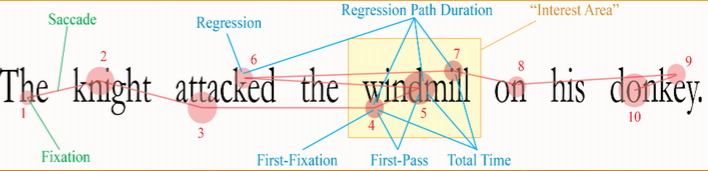
**COLRS module.** COLRS Module use open source Natural Language Processing (NLP) to tokenize and parse the input document, and it will generalize the P.O.S. tags in more readable format. The output will be a tokenized stream with token and tag pair.

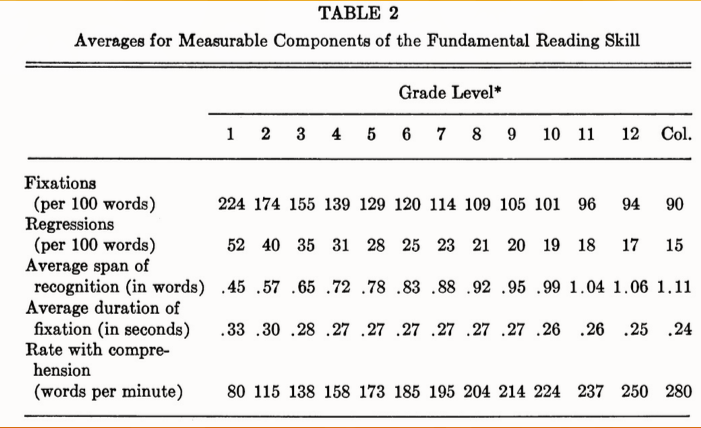
**Slash module.** Slash Module will accept tokenized stream from COLRS Module and decide how to break sentences into digestible lexical bundles. There will be a sub-component to deal with the exception list that improve the accuracy of the final output.

**Client-side reader.** The reader program will receive data from the server and parse the text according to their P.O.S. tag. There will be two display modes, color mode and speed mode. Under color mode, user can choose what P.O.S. tag to be colorize, 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.

# **3 IDENTIFICATION OF CASE STUDY**

Old Dominion University is well-known for its international student’s communities. Many students from all over the world come to Old Dominion in hope to pursue a college degrees in varies fields. However, the most challenge part for those students were the language barrier. Many of the international student need to spend over years in our English Language Center, or ELC, to learn English before they could start their academic studies, but this process can be boring and expensive for those students.

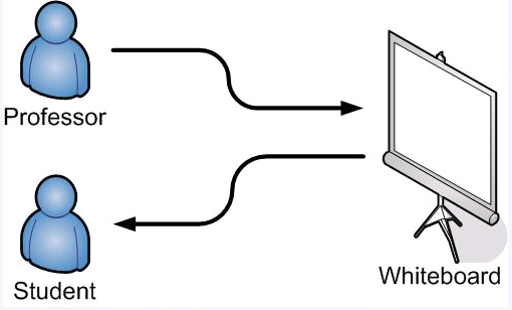




Eye Movements in Reading: Facts and Fallacies, Author(s): Stanford E.

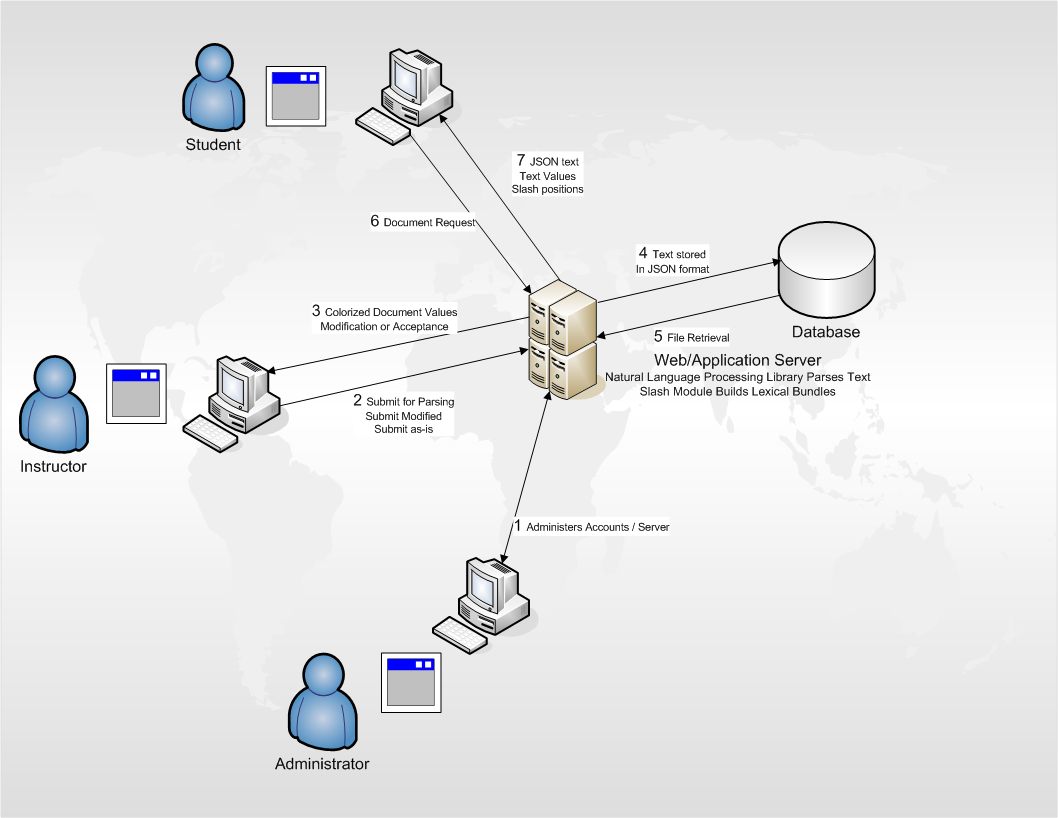
Taylor American Educational Research Journal, Vol. 2, No. 4

Professor Greg Raver-Lampman, who proposed this project, is an ELC instructor at Old Dominion University. He had found that ESL students generally have a bad reading performance due to fixations and regressions.



Current teaching mode

Professor Raver-Lampman also felt there was lack of software that can help instructor to automatically identify lexical bundles in the document, to share reading materials with students, and to track students’ participation and improvement of reading performance.



CLASH teaching mode

To solve this situation, we come up with CLASH software. CLASH will help instructor automatically slash the document into lexical bundles, and help instructors to store the file on server. Also, instructor will be able to share the document with students and keeping track of their activities. The most important functionality is the Slash read mode that aim to reduce reader’s fixation and regression by display the document in lexical bundles.

# **4 CLASH Product PROTOTYPE DESCRIPTION**

The goal of this project is to create an interactive website that can help instructor improve the teaching efficiency, and also increase the students’ reading performance. The CLASH prototype will be built as a Single Page Application, or SPA. It will be a highly responsive single webpage that does not reload as webpage content change, and it will be built completely by JavaScript, which is supported by all major browsers. The traditional SPA program includes a JavaScript user interface, a JavaScript web server, and a NoSQL database such like MongoDB. The CLASH application will be slightly different than traditional SPA, in which we will be using traditional relational database, MySQL, instead of the NoSQL database. The CLASH will still be built with a pure JavaScript user interface and a JavaScript web server to increase responsive.

**Benefits to the user:**

SPA can offer a desktop application look and feel, and users can access the tool without any complex installation. The JavaScript server is also known for its highly responsive feature, it can reduce the waiting time even with high volume of internet traffic. Also, the entire application will be built in a single webpage, so it will be easy for users to see all the functionalities.

**Overview of prototype vs real world product:**

|  |  |  |
| --- | --- | --- |
| **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 Reader. | Ability to view documents with slashes inserted and SLASH Reader. |

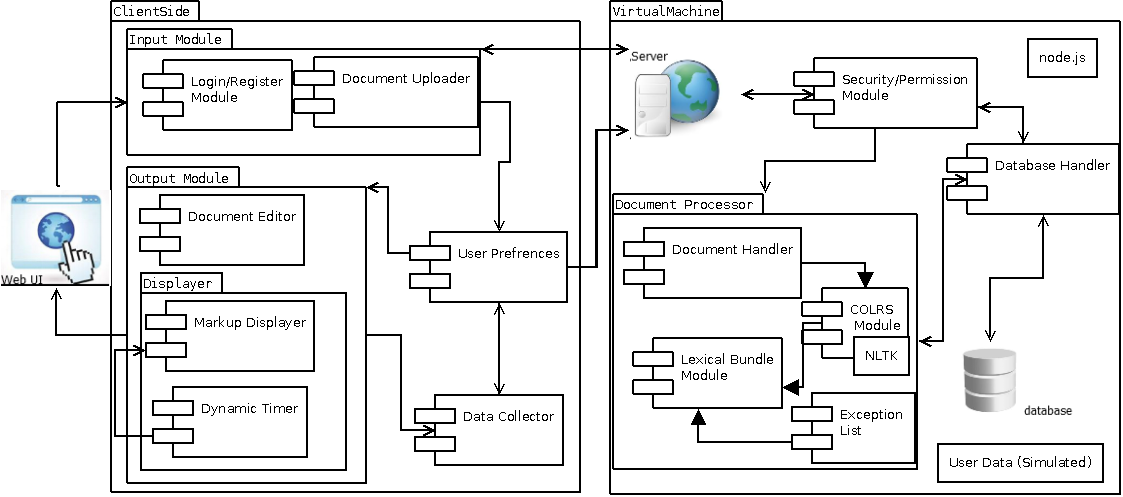
**Overview of simulated areas and models:**

The CLASH prototype will not be able to access any student enrollment data from Old Dominion, so instead, we will be simulated the students and instructors manually. The instructor will be allow to add students from his/her class into the prototype as test cases, and they will be allow to access most of the functionalities CLASH provided. Also, reading materials and student tracking datas will be simulated for test purpose.

## 4.1 Hardware and Software Prototype Architecture

**Hardware.** There will not be any hardware requirement other than a running web server. On the user’s side, the only hardware need will be an Internet capable computer.

**Software.** There will be some software requirement for the server, but all of the software will be open source project or liberally licensed software. Our server will be running Ubuntu 14.04 LTS operating system, and the main http server will be the Nginx and the Nodejs, which will handle incoming network request from students and instructors. The COLRS module will be written in Python with the open source NLTK language processing toolkit, and Slash module will be a pure Java application that break document into lexical bundles according to the input from COLRS. We will be using MySQL database on the server to manage user authentications and document storage, the database will also include miscellaneous data like tracking data and session information.



**Prototype MFCD**

## 4.2 Prototype Features and Capabilities

The CLASH prototype application will have the ability to identify parts of speech and break the document into digestible lexical bundles. In addition, the prototype will provide a speed reader to display one lexical bundle at a time. The prototype will provide functionality to forward or backward lexical bundles, and adjuster the display speed on the fly. Also, users will be able to send feedback about the accuracy of colorization and lexical bundles.

A working prototype will allow us to test the product. There will be some student test group from the Old Dominion University’s English Language Center who will use this product for few week, and we could compare their reading performance before and after using the CLASH. The result can be served as a proofs of concepts of the prototype.

## 4.3 Prototype Development Challenges

There are several challenges during CLASH development. The CLASH is a class project, so most of the team member need to find a time slot in between work, it become hard for all team member to sit together and collaborate with each other. The second challenge was the language complexity, we are running JavaScript as backbone on the server, but we have to use NLTK, which is a Python library, to identify the parts of speech; also, we need to use Java program to actually break document into lexical bundles, and putting different programming language together ant try to integrate them will be harder than using a single programming language overall. The last challenge will be the deadline, we need to build and make it workable before most of other team, so we can have time for Professor Raver-Lampman to actually test the tool and see how lexical bundle reading help students’ reading performance.

# **Glossary**

**CLASH** - Color Lexical Analysis algorithm and Slash Handler

**COLRS** – Colored Organized Lexical Recognition Software

**ELC** – English Learning Center

**ESL** – English as second language

**IBT** – International benchmark test

**JSON** – JavaScript Object Notation

**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) for the Python programming language.

**Node.js** – an open source, cross-platform runtime environment for server-side and networking applications.

**POS** – Parts of Speech

**SPA** – single page application, is a highly responsive web application that fits on a single page and does not reload as the web page changes states.

**TOEFL** – Test of English as a Foreign Language

**Ubuntu** – a Debian-based Linux operating system.

**VM** – Virtual Machine

# **References**

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

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.

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