Lab I - CLASH Product Description CS411

Artem Fisan

Professor Jannet Brunelle

Professor Hill Price

Old Dominion University

**Table of Contents**

1 Introduction 3

1.1 Key Product Features and Capabilities 3

1.2 Major Components (Hardware/Software) 3

2 Identification of Case Study 3

3 CLASH Product Prototype Description 5

3.1 Prototype Architecture (Hardware/Software) 6

3.2 Prototype Features and Capabilities 4

3.3 Prototype Development Challenges 5

Glossary 11

References 12

**1. Introduction**

CLASH stands for Color Lexical Analysis Slash Handler. It is a single page web application that is made up of three modules, COLRS, SLASH, and Slash Player. The COLRS module ‘colorizes’ each part of speech (P.O.S) in a text document with a particular color to increase comprehension of sentence structure and grammar. The ‘Slash’ module parses text into lexical bundles (a group of words that occur repeatedly together, or present one single thought) to increase reading speed. Though our program could be useful in a number of settings, we are primarily concerned with English as a Second Language (ESL) students.

Based on state-reported data, in 2004, it is estimated that 4,999,481 ESL students were enrolled in public schools. Despite those numbers 15% of these ESL students had no special resources or programs to help them learn English. In 2001, in the states that tested ESL students in reading comprehension, only 18.7 percent of ESLs were assessed as being at or above the normal reading level. That same year, 10% of ESLs in grades 7-12 were retained. In February 2001, it was reported that ESLs had dropout rates up to four times that of their native English-speaking peers. Here at ODU, we have an entire department focused on addressing this problem, but while they work diligently to help ESL students, the processes of teaching reading and grammar are outdated. (\*\*McKeon)

Currently, the process is simple, the professor writes a sentence on the board, then circles, or marks each part of speech. Not only is this time consuming, it limits the amount and size of examples that can be given. Psychologically, it has been proven that color impacts learning, they relieve eye fatigue, increases information retention, increase productivity and accuracy, and support developmental processes. (\*\*http://sdpl.coe.uga.edu/HTML/W305.pdf) Extrapolating from that information the ‘COLRS’ module should help students identify P.O.S.(Part of Speech) and increase their potential for learning grammar. For increasing reading speed and comprehension, reading assignments are given, or students are directed to sites like Spreeder (\*\*http://www.**spreeder**.com). These sites don’t help comprehension due to their focus on speed; they teach students to read word for word vs. in lexical bundles. It has been shown that those who learn lexical bundles read faster and perform better in word and sentence recall experiments.(\*\*Tremblay, Derwing, Libben, Westbury)

**1.1 Key Product Features and Capabilities**

CLASH features can be accessed using a standard web browsers (GoogleChrome, Mozila FireFox, Opera) via internet connection. User interface of Slash Player will provide to users control of the speed at which lexical bundles are displayed. Parts of Speech will be colorized and view in COLRS module. The prototype will have password controlled user logins, that consists of three different types of user roles; Administrator, Instructor, and Student. Administrator will be able to add and remove Instructor and Student users, as well as Instructor capabilities. The Instructor will be able to add and remove Student users, upload documents, edit slashed documents, and select the documents available to be viewed by the Students. The Student will be able to select which module they would like to view, control POS colorized displayed by grammar (noun, verb, pronoun, etc.), control reading speed, and which available document to view. The application allows for the Instructor to view activity data for the Student users such as time of usage and time spent reading. This activity data will include the student’s current reading speed, and the amount of time spent on the site. CLASH is unique because it displays parts of speech and lexical bundles. The Slash Player module will display a lexical bundle one at a time. There is also the ability to pause and change the display speed.

**1.2 Major Hardware and Software Components**

The hardware required by CLASH to work is any machine that can be used for programing in JavaScript and HTML, as well as a server. The software required is a database on the server end, an IDE (NetBeans, Notepad ++, CodeBlock, etc.). The server will store the website and database as well as documents uploaded by users. The database will contain the following information:

|  |
| --- |
| Corls Program |
| Slash Program |
| Slash Reader program |
| User account information |
| User credentials |
| Tracking Data |
| Original Documents/Parsed Document |
| Exception List |

The COLRS Module uses open source Natural Language Processing (NLP) to tokenize and parse the inputted document, it will create the POS tags for the document. The different POS tags will be displayed in the selected color, and the user will be able to select which POS to be displayed. The SLASH Module will take an inputted document, whether copy and paste or uploaded document, and use the results of the COLRS Module to break the sentences into lexical bundles, of 3 to 5 words. There is no rule without exceptions, therefore an exception list was created to improve the accuracy of the product. The Slash Player will receive input from the user and display a lexical bundle at a time, user will be able to choose the reading speed at which they will be presented.

**2. IDENTIFICATION OF CASE STUDY**

There are a lot of international students attending Old Dominion University to pursue degrees in 124 Undergraduate programs, 54 Master’s programs, and 42 Doctoral programs \*\*http://www.odu.edu/academics. International students are required to prove a passing score in English proficiency. This is shown with tests such as TOEFL and iBT, the scores required to pass are TOEFL scores between 500 and 550, and iBT 61-79. \*https://www.odu.edu/content/odu/admission/proficiency.html. Students that do not pass, are directed to the English Language Center to participate in the Bridge Program. Greg Raver-Lampman, an instructor at Old Dominion University English Language Center, is one of multiple teachers that are attempting to improve education techniques with modern technology. The current teaching process in the classroom is to write paragraphs that later become papers, read articles and books, listen and comprehend conversations from videos, and learn how to listen to discussions. Professor Raver-Lampman requested help from the Computer Science Senior Design class, which resulted in the creation of CLASH. This web application will be primarily created to help students at the ELC to improve their language skills.

**3. CLASH Product Prototype Description**

The CLASH prototype will be built as a modified version of a Single Page Application (SPA). A SPA is a highly responsive web application that does not reload. Traditionally, SPAs are built completely in. Traditional SPA stack includes a JavaScript built user-interface, a JavaScript friendly application and web server, and a NoSQL database. CLASH will not use a NoSQL database, but a traditional relational database, and Node.js as the web and application server. A SPA is an ideal platform for CLASH because it offers the abilities of a desktop application, but the representation on the World Wide Web (WWW). All functionality will be accessible through menus on the web page.

CLASH will have a user management module that will allow for Administrators and Instructors to add and remove users as needed. Administrators will have access to add and remove Instructors and Students. Instructors will be able to add Student users on a one-by-one basis and add users in bulk with an ODU enrollment file.

**3.1 Hardware and Software Prototype Architecture**

The hardware required for the prototype system will initially be a laptop owned by the application developers. Upon receipt of a Virtual Machine (VM) (provided by the ODU Computer Science Department) the system will be migrated to the VM.

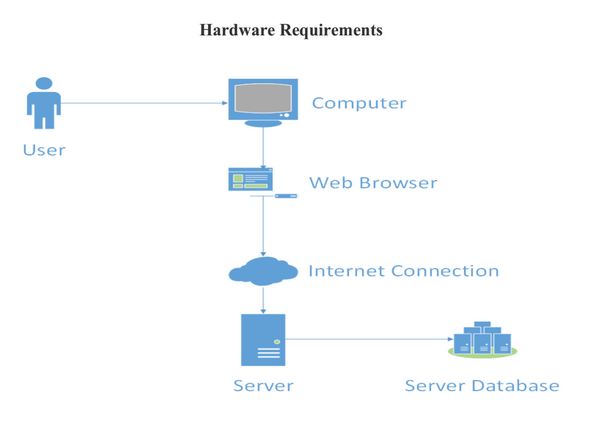
 **Software.** The software required to prototype this system will be a collection of open source or liberally licensed software to permit distribution or potential commercialization of the software package. Our prototype will consist of the following packages installed under Ubuntu 14.04 LTS. The web and application server will be Node.js. This software will provide the http server to allow incoming network connections from users (the students and instructors), as well as the application handling text processing of documents. The Node.js server will also interact with the MySQL database server running on the same machine, and Natural Language Toolkit. The configuration and integration of these separate components is the heart of this project. 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 1.2

**3.2 Prototype Features and Capabilities**

The CLASH program prototype will have the ability of identification specific parts of speech by coloring the text and will also identify lexical bundles through the slash insertions. In Slash Player users will be able to display bundles one at a time with features of increasing, decreasing, or pausing the lexical bundle stream. There is no product similar to CLASH existing, this will demonstrates our success.

Determined risks are to be mitigated through the involvement of the mentor throughout development of the product for ease of use, accuracy of core features and functions and the integration of teams building the modules.

Building a working prototype with these core features and functions will result in the customer being able to test the product in a real world situation against a control group, further assisting the development of the product. Completing these proofs of concepts the program prototype will demonstrate the feasibility of such a system and together with the prototype testing group, the utility of using the product in an academic environment thus giving the customer the desired outcomes.

**3.3 Prototype Development Challenges**

There are multiple challenges that CLASH may face on its developmental journey. One of them is computational time of the exception list. By adding more exceptions to the list, computational time increases due to the increase of size of the list itself.\* Another challenge is missing attributes; meaning features are not working properly, not working at all, or missing completely. CLASH, being unique and very useful, will have minor hiccups that have to be taken care of when they show up. It is impossible to write a program without minor bugs, but those issues will be discovered by actual testing or as they occur.

English is a challenging language and with that comes errors. One particular error that will possibly arise is the incorrect identification of parts of speech and causing incorrect placement of slashes. CLASH aims to provide the user the ability to control the speed at which the lexical bundle will show with the “Slash” feature. There is a possibility of having difficulty getting the algorithm that will control the speed to do the proper task. CLASH is meant to be simple for use, meaning having a simple interface. This is to make the product accessible to users with various computer skills levels. There is that possibility that what we believe to be simple is not simple which will in the long run affect the appeal it has for certain users. Furthermore, another challenge that will arise with time is storage capacity. Users will populate our database with their documents and 10GB will not be enough for long term use. Overall, our prototype is going to have a few challenges but CLASH will one of a kind (Lab 1).

**Glossary**

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

**COLRS** – Colored Organized Lexical Recognition Software

ELL – English Language Learners

**ELC** – English Learning Center

**ESL** – English as second language

GB – Gigabyte

GUI – Graphic User Interface

HMTL – HyperText Markup Language

**IBT** – International benchmark test

JS - JavaScript

**JSON** – JavaScript Object Notation

LB – Lexical Bundle

**Lexical Bundle** – a group of words that occur repeatedly together within the same register

NLP – Natural Language Processing

ODU – Old Dominion University

**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.