Lab 2 – Blackboard Archive Extractor Product Specification Outline

Team Crystal

Dominic Nguyen

CS411W

Professors Thomas J. Kennedy

March 1, 2017

Version 1
Table of Contents

1. INTRODUCTION 3
   1.1 PURPOSE 5
   1.2 SCOPE 6
   1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS 7
   1.4 REFERENCES 7
   1.5 OVERVIEW 8

2. GENERAL DESCRIPTION 8
   2.1 PROTOTYPE ARCHITECTURE DESCRIPTION 9
   2.2 PROTOTYPE FUNCTIONAL DESCRIPTION 10

List of Figures

Figure 1: Current Manual Process Flow ................................................................. 4
Figure 2: Automated Process Flow ......................................................................... 4
Figure 3: Major Functional Component ................................................................. 6
Figure 4: Example of the GUI .............................................................................. 10

List of Tables

Table 1: Real World Product with Prototype ......................................................... 9
1. Introduction

According to Corcoran (2014), approximately 17,000 schools were using Blackboard, in over 100 different countries. Blackboard is an online tool used by schools, colleges, and universities to organize course materials. This tool allows professors to distribute course materials such as syllabi, notes, and assignments to students over the Internet.

This would not be useful for instructors who are required to present their course materials to a Review Board. According to the Old Dominion University Faculty Handbook, lecturers are reviewed every year, senior lecturers are reviewed every three years, and tenured professors are reviewed every five years. Depending on one’s status with the University, lecturers rely on Blackboard to create their portfolio that will then be evaluated. However, there are various obstacles that instructors encounter while using Blackboard as a teaching tool for reviews. According to the ODU policy, courses on Blackboard are limited to being stored with a maximum of 2 years. Professors are forced to either move their content to an empty course or export their course as a Blackboard archive file. A tenured professor would have to move their materials at least twice between evaluations. If the lecturer opted to create a Blackboard archive of their course, they would have to manually extract their content page-by-page. They would also have to manually format the folder structure to allow it to be viewable. This process would have to be repeated for each course archive as illustrated in Figure 1. This proposes the issue of professors having difficulty retrieving their course content after being exported to a Blackboard Course Archive.
With the Blackboard Archive Extractor, its goal is to automate this process by going through various algorithms to make the process automatic and viewable as depicted in Figure 2. This will provide users with a solution that takes less time and less effort than the current process.
1.1 Purpose

Blackboard Archive Extractor is an application designed to allow teachers and professors to load a Blackboard course archive and explore its contents. The application will be able to handle selected Blackboard contents such as blogs, modules, documents, and exams. The application will then create a directory containing all the extracted and processed contents from the Blackboard Course Archive. All materials in the archive will be viewable and accessible through an HTML page for review committees to observe or personal use. Blackboard Archive Extractor will be a desktop application that can be executed from either a command line interface or a graphical user interface. The application will assist lecturers, instructors, and professors organize and maintain materials that are hosted on Blackboard.
1.2 Scope

The Blackboard Archive Extractor will consist of three major components including parsing, linking, and reformatting. The major functional components are as specified in Figure 3. Blackboard Archive Extractor will include a user interface, algorithms to recreate the course, and an output that will be viewable. The application will be desktop software available for Windows 7 through Windows 10. There will be an input and output that will take in the Blackboard archive and create an output that will be viewable. The software will allow the user to drag and drop the archive into the application or browse the computer for the file. The user will then have the option to choose where to save their reformatted Blackboard file. Upon completion, the application will create a navigable set of HTML documents that will display a webpage with the user’s materials from a Blackboard course.
1.3 Definitions, Acronyms, and Abbreviations

**Blackboard Learn** – is a virtual learning environment

**Blackboard Archive** – a ZIP file of the course content and user interactions

**Blackboard Archive Extractor** – a tool that extracts blackboard archives into an easily navigable set of HTML documents

**GUI** – Graphical User Interface is a type of user interface that allows users to interact visually

**IMS Course Cartridge** – is a specification that describes format for creating and sharing primarily educational digital content

**Old Dominion University (ODU)** – is a public co-educational research university located in Norfolk, Virginia

**Root Index** – where the website files are stored

**UI** – User Interface is where a user and system interact

**URL** – Uniform Resource Locator a web address referencing a web resource, specifying location

**UX** – User Experience is the overall experience when using a product

**ZIP file** – is an archive file format that supports lossless data compression

1.4 References


3. Lab I

1.5 Overview

This product specification provides the hardware and software configuration, capabilities and features of the Blackboard Archive Extractor prototype. This prototype will not use of external interfaces. The information provided in the remaining sections of this document includes a detailed description of the hardware, software, and architecture of the Blackboard Archive Extractor prototype. This specification also describes the architecture design of the prototype and how that design leads to the functional capabilities of the Blackboard Archive Extractor. The specification provides a precise list of requirements which the prototype must satisfy, located in Lab 2 Section 3.1 located in a separate document.

2. General Description

The Blackboard Archive Extractor is a software designed to target educators that use Blackboard. The user will use the application through CLI or GUI to upload their Blackboard archive. The Blackboard Archive Extractor tool is a software containing algorithms that will parse and organize contents inside Blackboard archives. The contents are restructured into a viewable directory tree comparable to the Blackboard website. The newly created archive is saved onto the hard drive and a web content is created. With these features, the Blackboard
Archive Extractor’s goal is to provide educators a method of creating a teaching portfolio automatically.

2.1 Prototype Architecture Description

Blackboard Archive Extractor is set to nearly implement every major functional component and algorithms required for the final product as illustrated in Figure 4.

<table>
<thead>
<tr>
<th>Features</th>
<th>Real World Product</th>
<th>Prototype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Can extract all content from an archive</td>
<td>Fully Functional</td>
<td>Partially Functional</td>
</tr>
<tr>
<td>Saves archive on hard disk</td>
<td>Fully Functional</td>
<td>Fully Functional</td>
</tr>
<tr>
<td>Checks every URL that points to an outside website</td>
<td>Fully Functional</td>
<td>Partially Functional</td>
</tr>
<tr>
<td>Execute application from command line interface</td>
<td>Fully Functional</td>
<td>Fully Functional</td>
</tr>
<tr>
<td>Execute program from a GUI</td>
<td>Fully Functional</td>
<td>Fully Functional</td>
</tr>
<tr>
<td>Doesn’t break on future Blackboard versions</td>
<td>Fully Functional</td>
<td>Partially Functional</td>
</tr>
<tr>
<td>Executes from a web application</td>
<td>Fully Functional</td>
<td>Eliminated</td>
</tr>
<tr>
<td>Creates web content from archive</td>
<td>Fully Functional</td>
<td>Fully Functional</td>
</tr>
<tr>
<td>Responsive website design</td>
<td>Fully Functional</td>
<td>Partially Functional</td>
</tr>
</tbody>
</table>

Table 4: Real World Product with Prototype
The application will be distributed through download from the website. The download will be hosted on a private server by ODU Computer Science department. The software will extract all content from the Blackboard archive. It will go through a linking and formatting algorithm to parse the archive. The prototype will check every uniform resource locator (URL) that points to an outside website. The Blackboard Archive Extractor will execute the program from the command line interface (CLI). It will also have the option to execute from a graphical user interface as illustrated in Figure 5. The program will accept the location of the archive. It can also receive the location by dragging and dropping the archive into the application. It will then save the new archive onto the hard drive in the specified location. The software will create a web content from the archive and have a responsive website design. This website will host all the content in the archive and will be viewable to those with the proper authorization. Lastly, this prototype will be compatible with future Blackboard versions.

2.2 Prototype Functional Description

The user must download the Blackboard Archive Extractor application onto their machine from the hosted website. It is also required to have a Blackboard archive that needs extracting. Once the user has obtained the two items, the user has the option to execute the program from either GUI or CLI.
For the GUI, there will be an input and output object accepting the location of the archive and where the extracted archive will be located. The application will also accept a drag and drop method onto the application as an alternative action to specify the Blackboard archive. When the objects are filled with the necessary locations, the user will need to perform the extract button to start the extraction. When it is complete, the user will be notified. A directory will be created where the location was specified containing all the contents. In effect to this, a fully responsive web design is created through this directory. The user will be able to view their materials through the web browser and allowing other viewers, like the review committee, to examine their materials used throughout the course.