Milestones- Blackboard Extractor

CS 410 - Team Crystal
November 17, 2016

Team Crystal Mentor

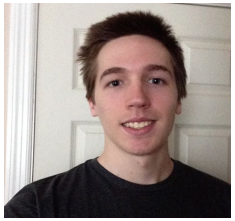
Dr. Steven Zeil

- Associate Professor, Old Dominion University
- Ph. D. in Computer Science, Ohio State University
- Extensive research in software testing, reliability measures, distance learning and digital libraries

Team Crystal

Dominic Nguyen

Software Engineer

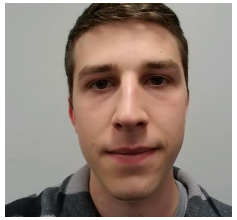


Grant Atkins

Software Engineer/Test Analysis

Austin Tillery

Software Engineer



Devin Haslam

Software Engineer

Merly Mathis

UI/UX Designer

Tristan Pressley

Project Manager/Web Master

Christopher Soffos

Systems Engineer

Outline

1. Problem Statement
2. Manual Extraction of Blackboard Archives
3. The Current Process Flow
4. Plans for Our Solution
5. Target Customers
6. Revised Process Flow
7. Major Components
8. Structure of a Blackboard Archive
9. Software Milestones
10. Software Milestones: Linking Method
11. Software Milestones: Formating
12. Software Milestones: User Interface
13. Software Milestones: Testing
14. Former Competition
15. Risk Matrix
16. Risk Assessment: Customer
17. Risk Assessment: Technical
18. Conclusion
19. References

Extracting Blackboard Archives

Our Problem Statement

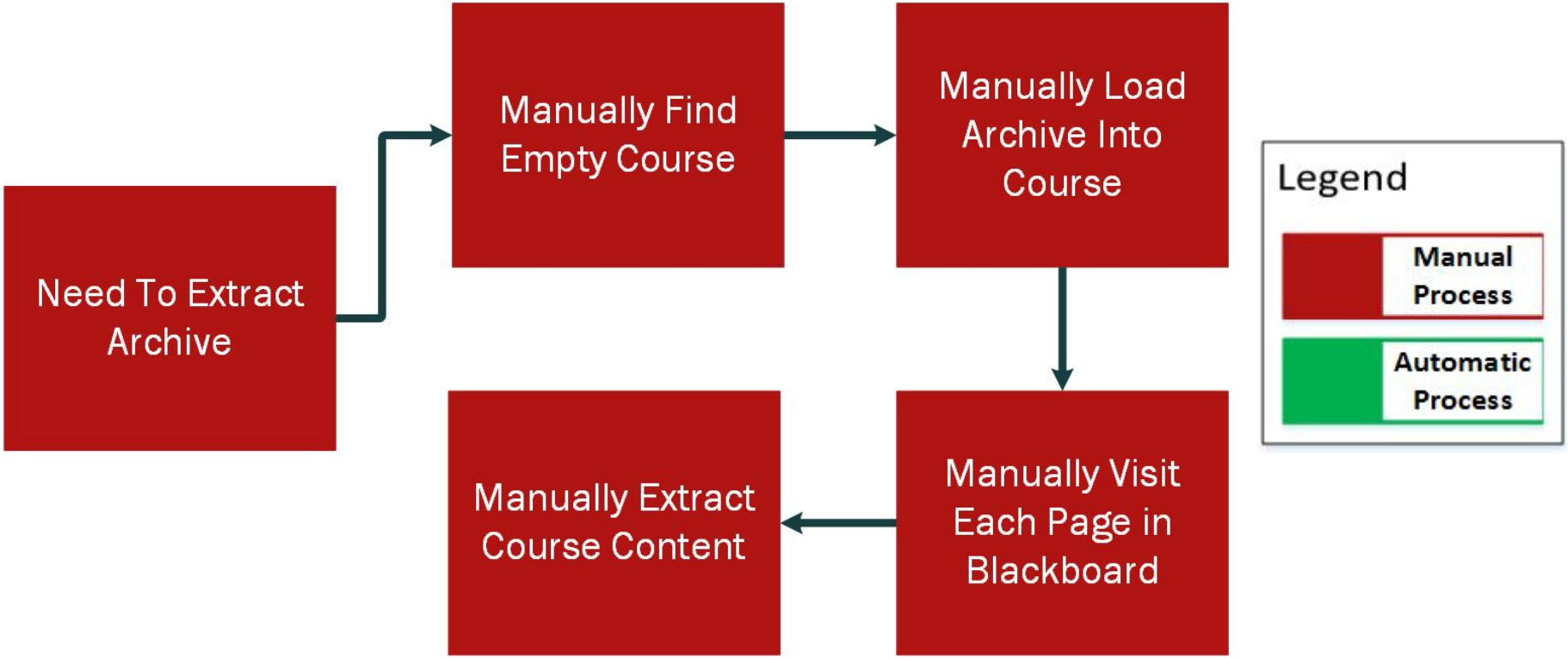
Professors have difficulty retrieving course content after it has been exported to a Blackboard Course Archive.

Manually Extracting Blackboard Archives

Currently, extracting a Blackboard archive requires a professor to:

- Find an unused Blackboard course into which he or she can load the archive.
- If there is no unused Blackboard course, it is not possible to extract the content.
- Manually download course content, page-by-page.
- Manually format folder structure.
- This process must be repeated for each course archive.

Current Manual Process Flow



Plans For Our Solution

Our solution will be an application which allows a user to load a Blackboard Course Archive and explore the contents of the archive.

- This application will be able to handle all types of blackboard content including blog posts, module items, pdfs, *etc*.
- The application will create a directory containing all of the extracted and processed contents from the Blackboard Course Archive.

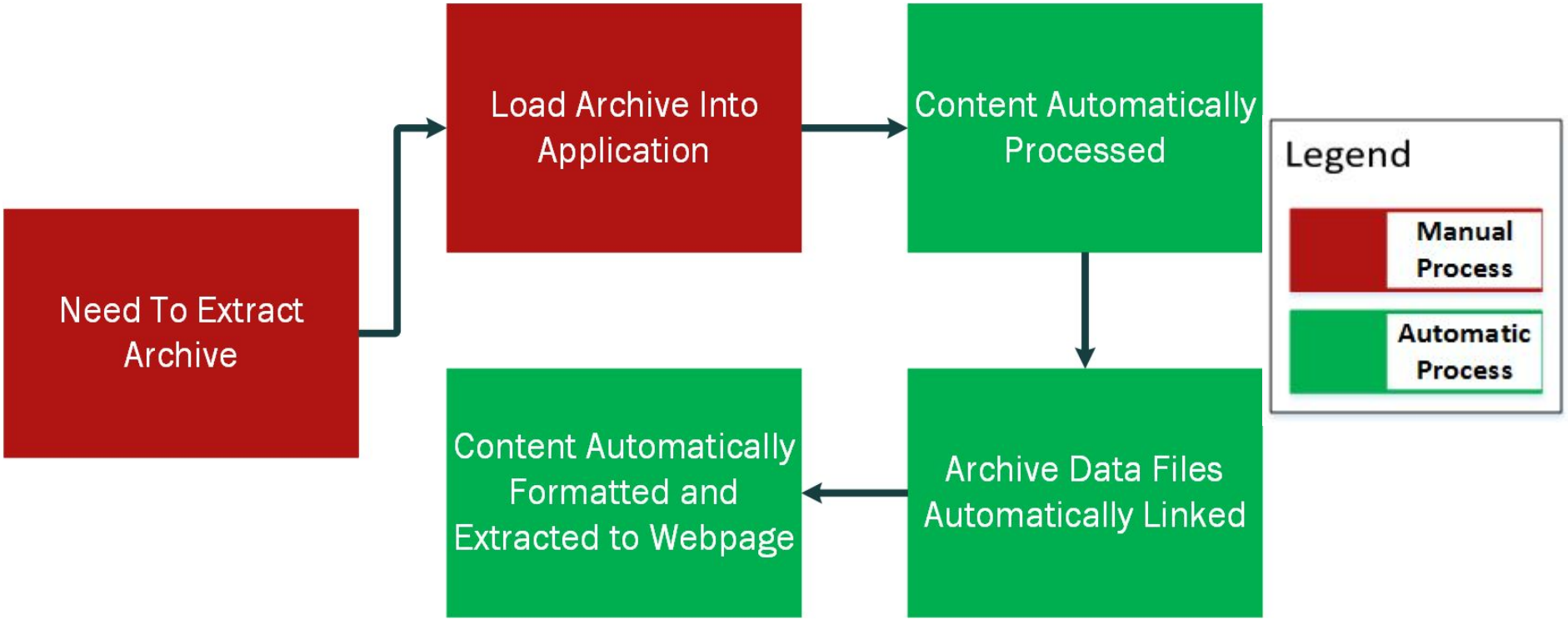
Goals and Objectives

- To automate the process for extracting blackboard course archives by implementing software that will process, link and format the data automatically.
- To provide users with a solution that takes less time and less effort than the current process.

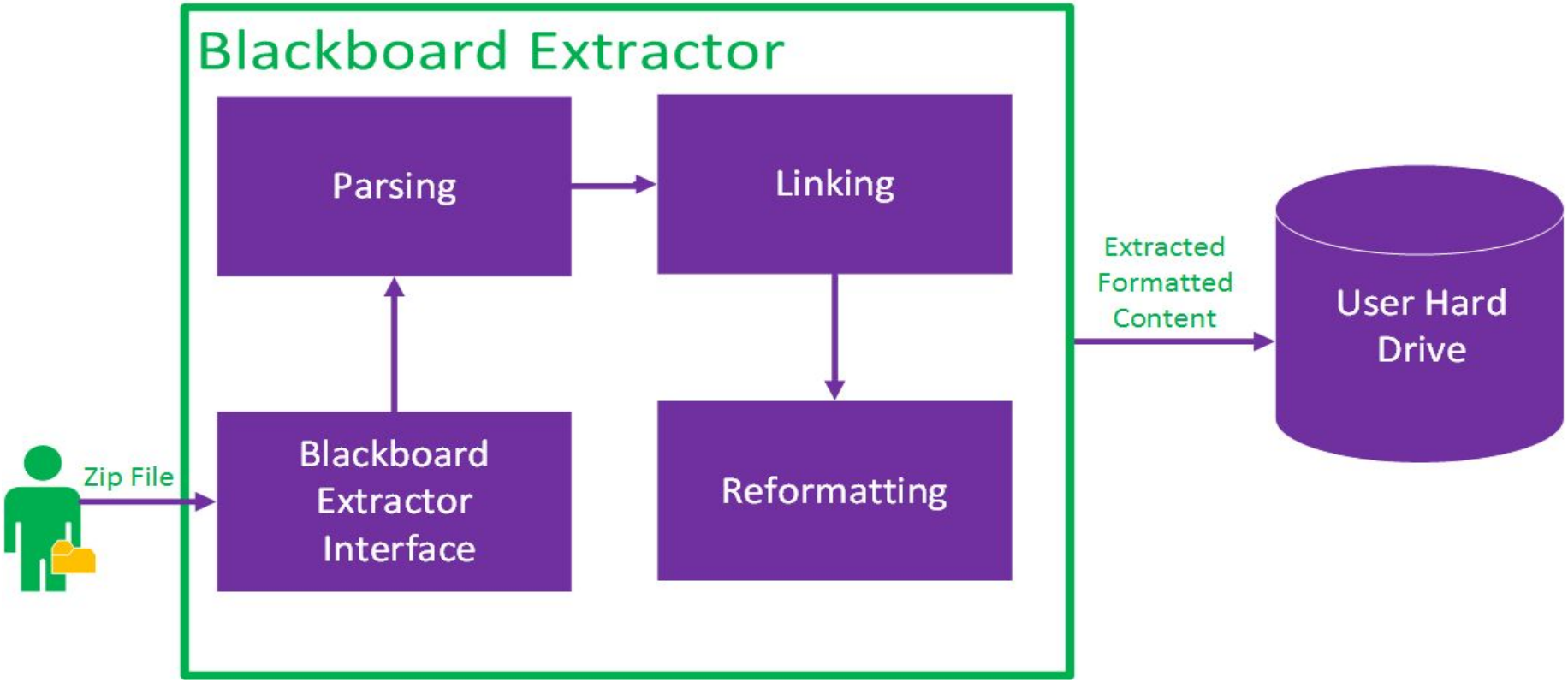
Target Customers

Our initial focus will be educational institutions currently using the Blackboard course management system.

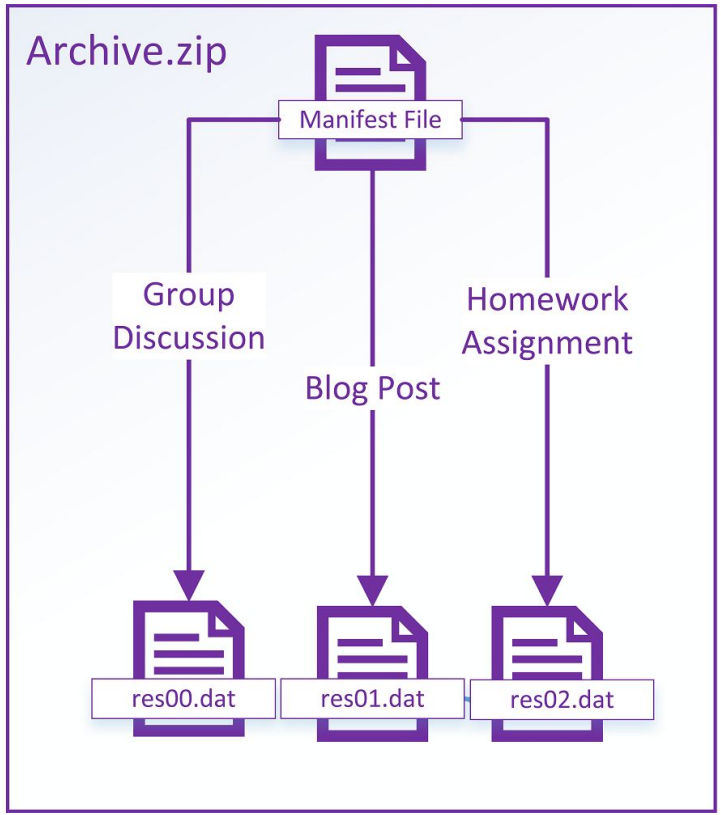
Revised Automated Process Flow



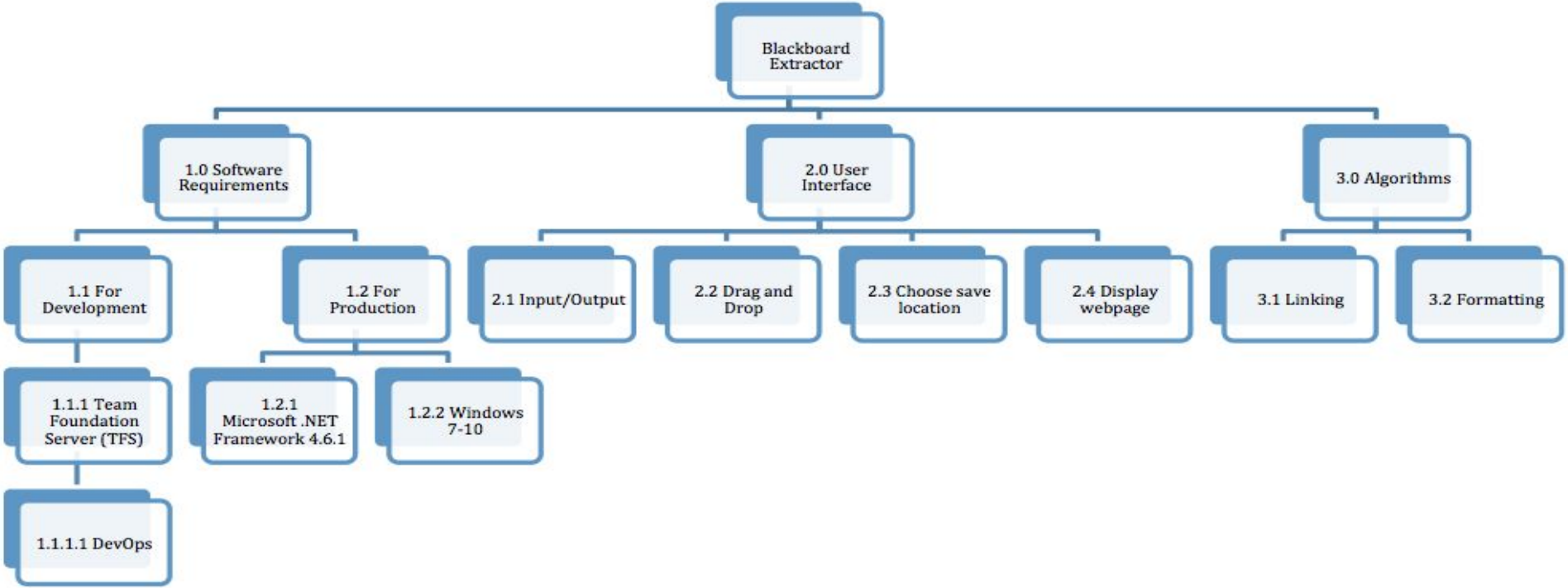
Major Functional Components



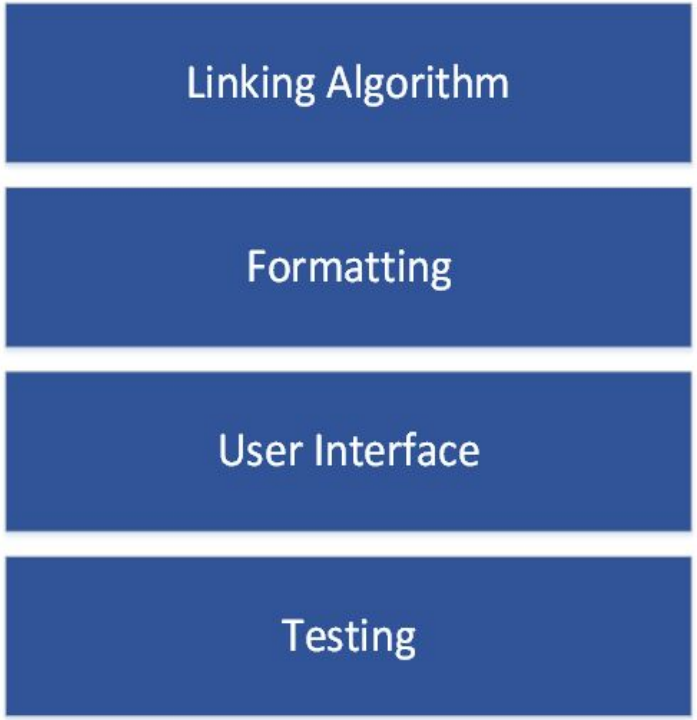
Structure of Blackboard Archive



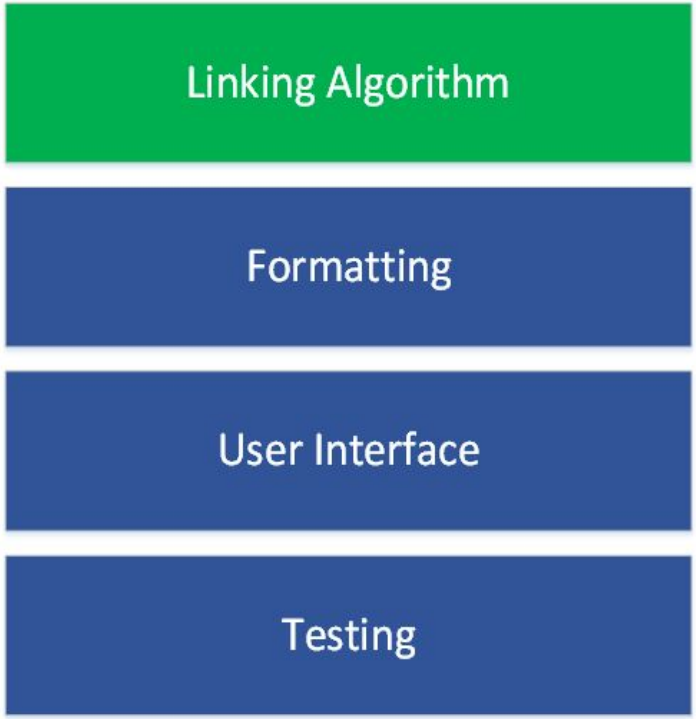
Work Breakdown Structure



Software Milestones

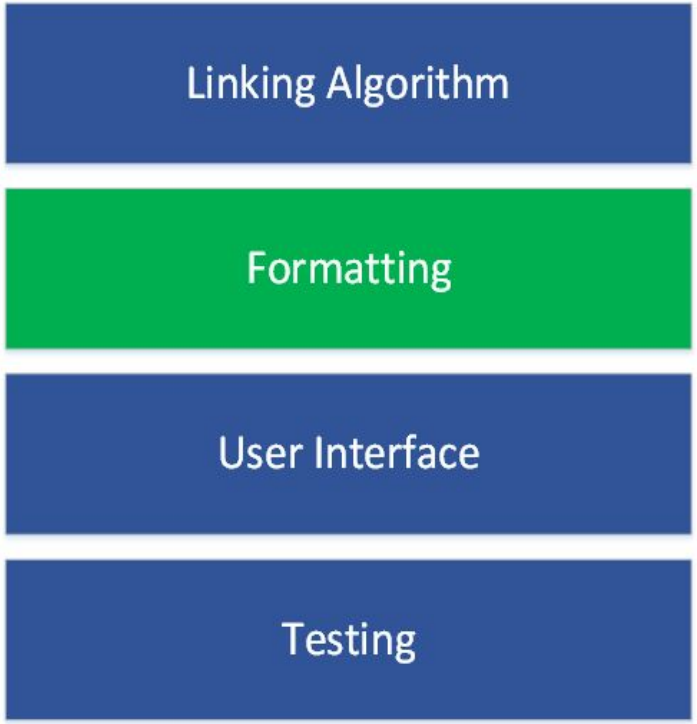


Software Milestones: Linking Algorithm



- Interprets Manifest file and identifies .dat files
- Builds links between files that are related to each other (i.e. parent/child, hyperlinks)

Software Milestones: Formating



- Uses the information interpreted by the linking algorithm
- Builds a file structure containing all of the contents, the contents are indexed by HTML files
- Entire structure of the archive will be viewable in the HTML pages

Software Milestones: User Interface

Linking Algorithm

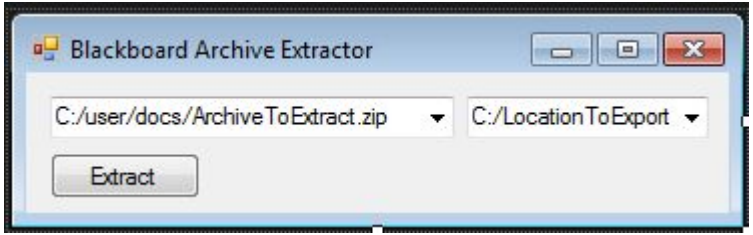
Formatting

User Interface

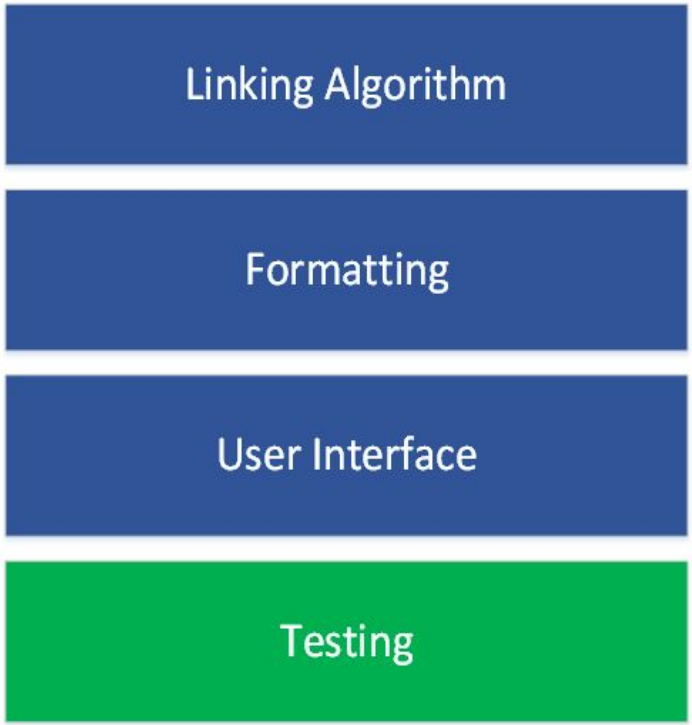
Testing

- Creates simplified Input / Output
- User can drag and drop a Blackboard archive to extract
- User can chose a location to save the Blackboard archive
- Can automatically open the index file to display the webpage

Example of GUI-



Software Milestones: Testing



Aspects to test:

- Linking method
- Formatting of archived course
- Effectiveness of GUI and display

Alpha Test

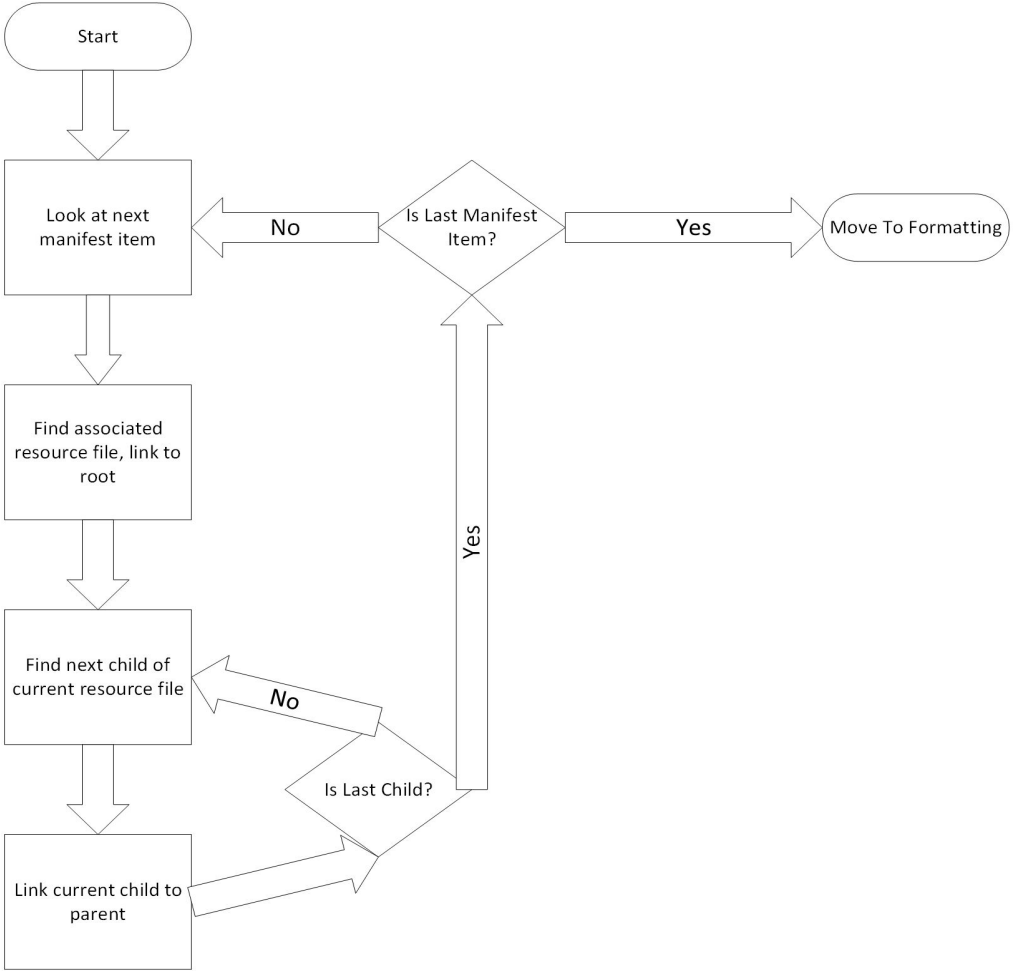
- Closed testing conducted by team Crystal
- The linking method will be the focus of the alpha test

Beta Test

- Open testing conducted by ODU faculty
- The formatting and display will be the focus of the beta test

Linking Algorithm

DRAFT VERSION



Formatting Algorithm

Draft version

Software Requirements

For Development

1. Visual Studio 2015
2. Gitlab (Might be slightly easier if we go with TFS)
3. Devops stuff (CI/CD, Build,Config,Deployment management, if we use TFS we can use Azure for all of this)

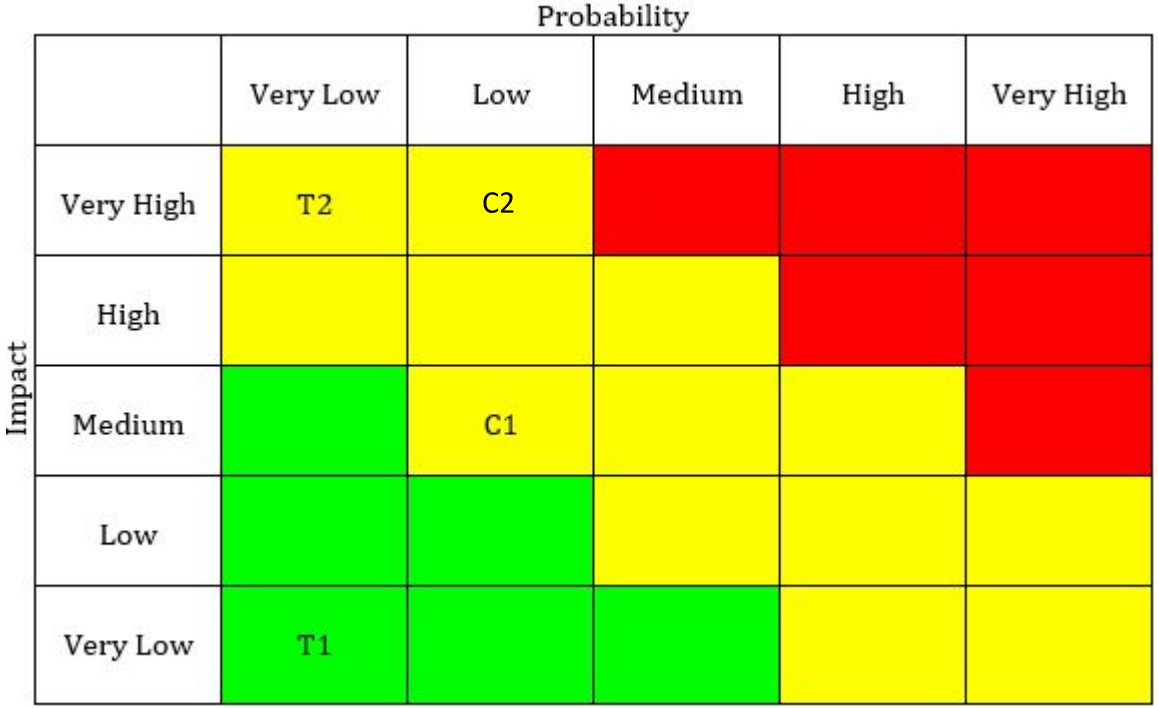
For Production

1. Microsoft .NET Framework 4.6.1
2. Windows 7, 8, 8.1, 10

Former Competition - bFree Blackboard Extractor

	Our Blackboard extractor	bFree Blackboard Extractor[3]
Extracts archives and converts to webpage	X	X
Runs a java web applet		X
Runs from an executable	X	
Modern Webpage Design	X	
Documentation included	X	
Supports Latest Blackboard Version 19	X	

Risk Matrix



Customer Risks

C1. Customer dislikes UX/UI

C2. University stops using BB

Technical Risks

T1. Security permission issues

T2. BB changes archive style

Risk Assessment: Customer

C1. Customer Dislikes UI/UX

Probability: low *Impact: medium*

If professors do not like the UI/UX, they will be hesitant to use the software. This can be mitigated by offering more than one interface format.

C2. Old Dominion Stops Using blackboard

Probability: low *Impact: very high*

If Old Dominion stops using blackboard, our software will be useless to the University. However, if our software is used at all schools using blackboard, the impact will not be as high.

Risk Assessment: Technical

T1. Security Permission Issues

Probability: very low *Impact: very low*

If there is unauthorized access to a user's account, we must notify the user and direct them to the system administrator.

T2. Blackboard changes archive style

Probability: very low *Impact: very high*

If Blackboard were to drastically change its archive style in a new update, our software would have to be updated. We can avoid this risk by only supporting one version of blackboard.

Conclusion

Professors have difficulty retrieving course materials from blackboard. Our blackboard extractor will help all professors using blackboard by automating and simplifying much of the archive extraction process. This includes our milestones which require a linking algorithm, formatting, user interface, and testing. We also have acknowledge the many different risks and have plans to address these risks.

Questions?

References

1. Corcoran, B. (2014, July 23). Blackboard's Jay Bhatt Strikes Up the Brass Band. Retrieved October 19, 2016, from https://www.edsurge.com/news/2014-07-23-blackboard-s-jay-bhatt-strikes-up-the-brass-band?elq=29068c31188e4150bde7dd6db8a3dab7&elqCampaignId=7599
2. ODU Faculty Handbook. (2005, December 21). Retrieved October 26, 2016, from http://ww2.odu.edu/ao/facultyhandbook/index.php?page=ch07s01.html
3. BFree: Extract Blackboard content. (n.d.). Retrieved October 19, 2016, from https://its2.unc.edu/tl/tli/bFree/index.html
4. Blackboard | Reimagine Education | Education Technology. (n.d.). Retrieved October 19, 2016, from http://www.blackboard.com/