Project Stratus
Sprint 3 Deliverable

Team P
Kyle Van Landingham, Alec Schule, Alec Fong, Andrew Zolintakis, Noah Bergman

Table of Contents

1. Executive Summary

This document describes the development process followed by team P through the third and final sprint of the projet. This includes a review and retrospective of the second sprint, focusing on how the team planned to address the issues that arose during sprint 2 and the sprint 2 demonstration of the system. Follow that are the product backlog and the sprint backlog sections that show which requirements the team planned on completing during sprint 3 of the project. Finally this document describes evidence that show the team followed the extereme programming practices and scrum requirements for the project.

2. Overview

2.1 Purpose

The purpose of this document is to track and document the work planned and completed in the third sprint of the project.

2.2 Intended Audience

The intended audience for this document is the product owners and the development team.

2.3 References

Reference	Relevant Information
GitHub Repository	https://github.com/kvanland/csci310TeamP
Product Backlog Excel Sheet	https://docs.google.com/spreadsheets/d/1IrTDg63ZMwNC10liWfWb5Cs50-KmELNTij1JuijRXHE/edit?usp=sharing

2.4 Definitions

Term	Definition
Product Backlog	A prioritized list of features or requirements for the system to have some time during development.
Sprint Backlog	A sublist of the Product Backlog that contains the features or requirements that the developers will tackle during the current sprint.

3. Sprint Review

- Articles sorted by frequency default
- Article list test only checks for table

Should test for everything frequency, conference, etc (including correct information)

- Bibtex test
- For conference list, clicked word must be in article
- Need article subset
- Export article list
- Need download link
- Download wc image

Sprint Planning/Retrospective Meeting 4/19/17

4. Sprint Retrospective

The group discussed that we should concentrate on a more sustainable pace during the sprint which can be accomplished through consistent communication and scheduling for pair programming sessions. The group also emphasized the importance of simple design so the team can effectively and efficiently adhere to the collective code ownership XP practice requirement. A long discusssed topic during the retrospecitive was the need to communicate on slack consistently between scrums and during pair programming sessions since the team pinpointed a lack of communication as a large factor in the slow development of the system.

5. Product Backlog

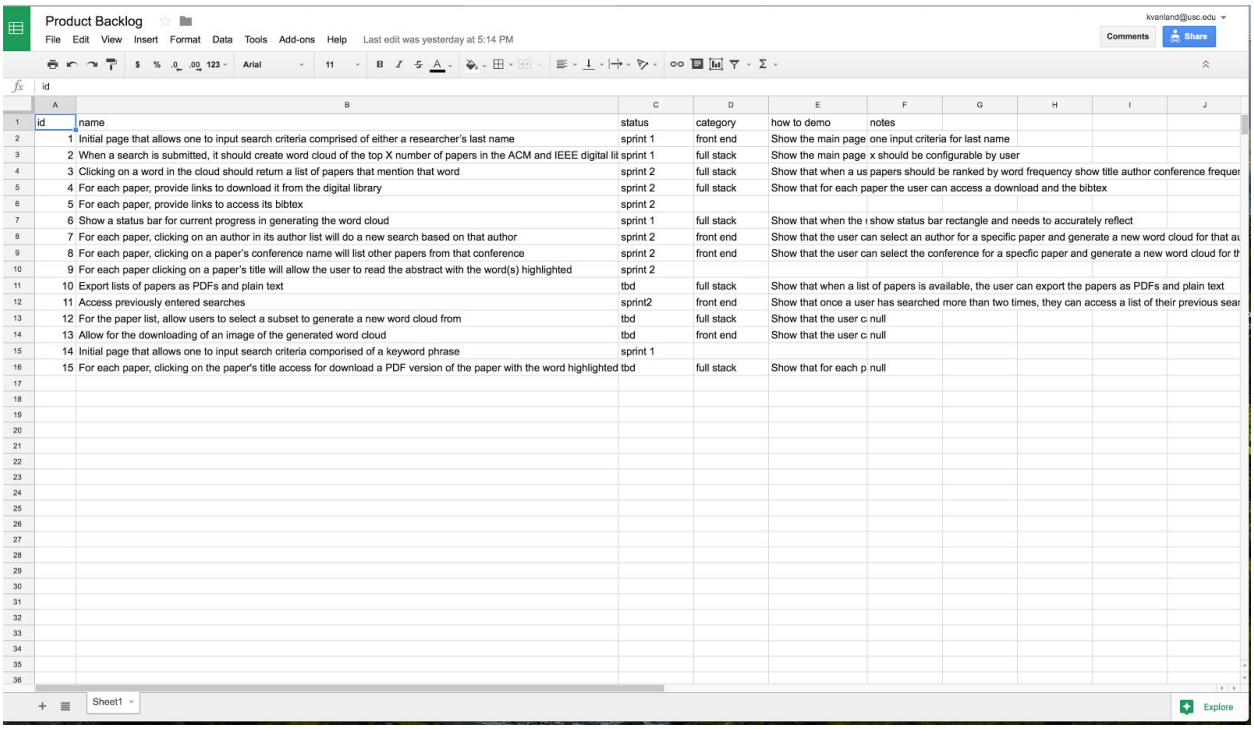
The ordering of the Backlog reflects the priority of each feature where number one is the highest priority and fifteen is the lowest priority.

Project description: Web application to display word clouds associated with researchers' papers that have been published in the ACM or IEEE digital libraries and allow navigation to inspect papers and authors.

1. Initial page that allows one to input search criteria comprised of either a researcher's last name.
2. When a search is submitted, it should create word cloud of the top X number of papers in the ACM and IEEE digital library that match the provided criteria. (X is configurable by the user)
3. Clicking on a word in the cloud should return a list of papers that mention that word
4. For each paper, provide link to download it from the digital library
5. For each paper, provide link to access its bibtex
6. Show a status bar for current progress in generating the word cloud (needs to reflect actual progress)
7. For each paper, clicking on an author in its author list will do a new search based on that author
8. For each paper, clicking on a paper's conference name will list other papers from that conference.
9. For each paper clicking on the paper's title will allow the user to read the abstract with the word highlighted.
10. Export lists of papers as PDFs and plain text
11. Access previously entered searches
12. For the paper list, allow users to select a subset to generate a new word cloud from
13. Allow for the downloading of an image of the generated word cloud
14. Initial page that allows one to input search criteria comprised of a keyword phrase

15. For each paper, clicking on the paper's title allows for download of a PDF version of the article with the word highlighted

Image of Product Backlog within an excel format. Link in the reference table.



Product Backlog

id	name	status	category	how to demo	notes
1	Initial page that allows one to input search criteria comprised of either a researcher's last name	sprint 1	front end	Show the main page	one input criteria for last name
2	When a search is submitted, it should create word cloud of the top X number of papers in the ACM and IEEE digital lil	sprint 1	full stack	Show the main page	x should be configurable by user
3	Clicking on a word in the cloud should return a list of papers that mention that word	sprint 2	full stack	Show that when a us	papers should be ranked by word frequency show title author conference frequer
4	For each paper, provide links to download it from the digital library	sprint 2	full stack	Show that for each	paper the user can access a download and the bibtex
5	For each paper, provide links to access its bibtex	sprint 2			
6	Show a status bar for current progress in generating the word cloud	sprint 1	full stack	Show that when the s	how status bar rectangle and needs to accurately reflect
7	For each paper, clicking on an author in its author list will do a new search based on that author	sprint 2	front end	Show that the user	can select an author for a specific paper and generate a new word cloud for that au
8	For each paper, clicking on a paper's conference name will list other papers from that conference	sprint 2	front end	Show that the user	can select the conference for a specfic paper and generate a new word cloud for th
9	For each paper clicking on a paper's title will allow the user to read the abstract with the word(s) highlighted	sprint 2			
10	Export lists of papers as PDFs and plain text	tbd	full stack	Show that when a	list of papers is available, the user can export the papers as PDFs and plain text
11	Access previously entered searches	sprint2	front end	Show that once a	user has searched more than two times, they can access a list of their previous sear
12	For the paper list, allow users to select a subset to generate a new word cloud from	tbd	full stack	Show that the user c	null
13	Allow for the downloading of an image of the generated word cloud	tbd	front end	Show that the user c	null
14	Initial page that allows one to input search criteria comporised of a keyword phrase	sprint 1			
15	For each paper, clicking on the paper's title access for download a PDF version of the paper with the word highlighted	tbd	full stack	Show that for each p	null

6. Sprint Backlog
    1. Fix any and all issues from Sprint 2
    2. Parse PDF instead of Abstract (From Sprint 2)
    3. Display Download Link (From Sprint 2)
    4. Export lists of papers as PDFs and plain text
    5. For the paper list, allow users to select a subset to generate a new word cloud from
    6. Allow for the downloading of an image of the generated word cloud
    7. For each paper, clicking on the paper's title access for download a PDF version of the paper with the word highlighted in the PDF

Justification for Sprint Backlog

During the Sprint Planning meeting documented above, the team assumed that all remaining Product Backlog items were to be added to the Sprint Backlog for the third and final sprint. The first three items on the sprint backlog were added due to lack of completing the Sprint Backlog from Sprint 2. These items were listed as the highest priority because they were meant to be completed in the last sprint. The other items on the list that were taken from the Product Backlog are listed in order of a mix of importance and ease of implementation.

7. XP Evidence

7.1 Test Driven Development

Feature: Generate word cloud from subset of articles (backend)
Test Commit: `aefe4cd3ec6efac90cbd765017344a10faa07f5c 4/20/17`
Feature Commit: `aefe4cd3ec6efac90cbd765017344a10faa07f5c 4/20/17`

Feature: Export article list as plain text (frontend)
Test Commit: `aefe4cd3ec6efac90cbd765017344a10faa07f5c 4/20/17`
Feature Commit: `aefe4cd3ec6efac90cbd765017344a10faa07f5c 4/20/17`

Feature: Export article list as pdf (frontend)
Test Commit: `3bbedb34b1ebfb31207edacdb124309157815e16 4/21/17`
Feature Commit: `3bbedb34b1ebfb31207edacdb124309157815e16 4/21/17`

Feature: Bibtex
Test Commit: `aefe4cd3ec6efac90cbd765017344a10faa07f5c 4/21/17`
Feature Commit: `aefe4cd3ec6efac90cbd765017344a10faa07f5c 4/21/17`

Feature: Download Link
Test Commit: `aefe4cd3ec6efac90cbd765017344a10faa07f5c 4/21/17`
Feature Commit: `aefe4cd3ec6efac90cbd765017344a10faa07f5c 4/21/17`

Feature: Create Subset WC (frontend)
Test Commit: `b89e0824fd5aa10f3e172aff7644b75cfd34ffae 4/23/17`
Feature Commit: `81e6cc73502c918154c51fe6e681518f215a717c 4/23/17`

Feature: Download Image of WordCloud
Test Commit: `b0fed0afb5bce5bbbc0a75bb613fdb38fc59a53e 4/27/17`

Feature Commit: `b0fed0afb5bce5bbbc0a75bb613fdb38fc59a53e 4/27/17`

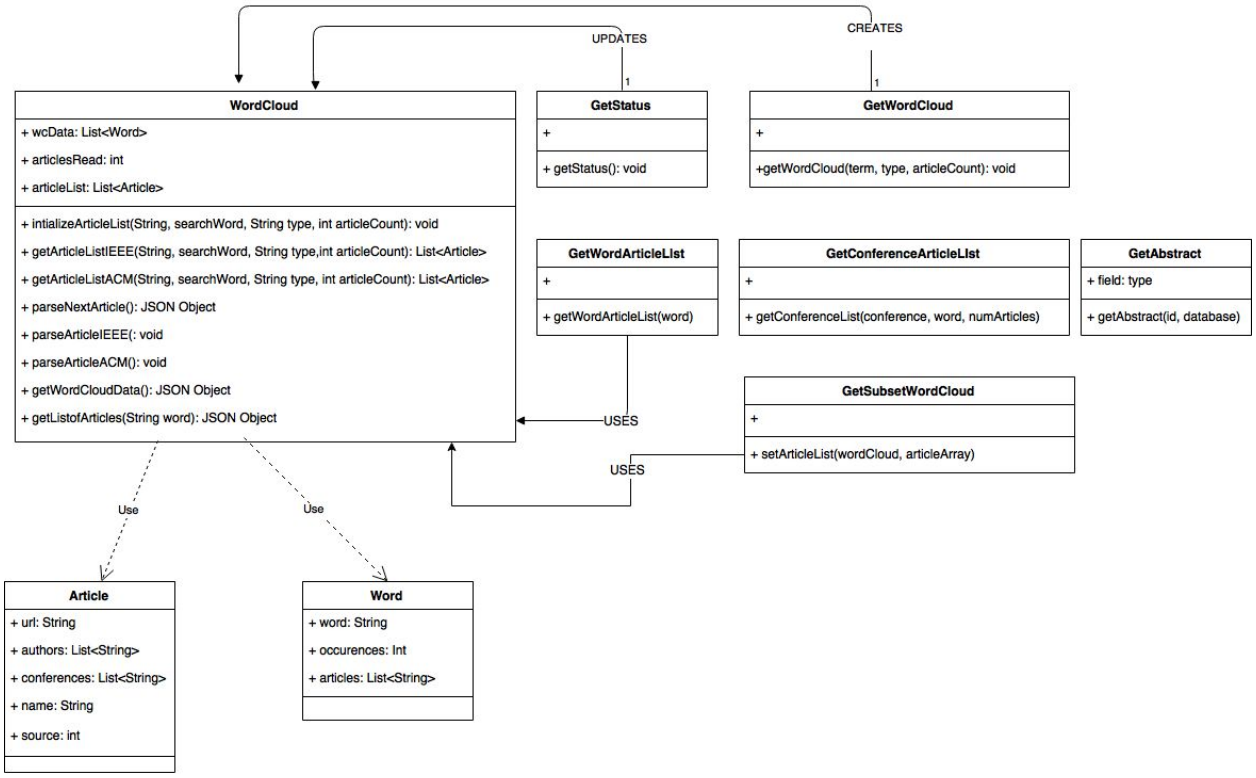
Feature: Generate PDF with word highlighted
Test Commit: n/a
Feature Commit: n/a

7.2 Pair Programming

To view evidence of pair programming during this sprint navigate to sprint3_documentation/pairProgramming in the gihub repository listed in the references section of this document. Inside each folder within that directory is a pciture of the pair programming team with the test that they ran after their session. The folders are intended to be created in the same commit as the code that was tested was commited but it is possible a few commits of the pair programming evidence were commited after the code was commited that is associated with the pair programming session but those would be no more than a day later in that case.

7.3 Simple Design

The system is designed to be simple in the sense that all the logic for the system is mostly centered within the php backend scripts. All the scripts that are called from the frontend via AJAX access data from a shared session object called WordCloud. WordCloud has all the data related to word cloud and any script that needs to access or manipulate that data has access to that data via the shared session instance of WordCloud. For each function the frontend there are corresponding php backend scripts that provide and manipulate the data for that specific function. There are no general or multifunctino scripts that have ambigous functionality, there are only well defined scripts that accomplish specific functions. This diesng is simple because it is easy to access the word cloud data from any script and all the data related to the system is centered within the word cloud object.



The class diagram represents the logic behind the backend part of the system. The WordCloud class represents a global shared object between all the php files which contains all the information needed to generate a word cloud and the article list associated with that word cloud. The reason for a lack of front end diagram is that the majority of the computing logic resides in the backend section of the system. This represents simple design because it centralizes the logic of the system which simplifies how the developers can achieve the same functionality of the system with logic in the backend versus implementing significant logic in both the frontend and the backend parts of the system.

The interaction between the frontend and the backend that is used in the system developed during this sprint is limited to AJAX calls to GetStatus, GetWordCloud, GetWordArticleList, GetConferenceArticleList, GetAbstract, and, GetSubsetWordCloud. The parameters of the methods in the class diagram. Out of these six endpoints, GetWordCloud, GetWordArticleList, GetSubsetWordCloud, and GetStatus use the global object wordcloud. This is why the classes have an arrow to the WordCloud class in the diagram. GetConferenceArticleList and GetAbstract make calls the third-party APIs to get their data.

7.4 Sustainable Pace

The visual of the burndown chart shows that the team attempted to stay on a sustainable pace during the third and final sprint of the project. There are frequent screenshots of the burndown chart in sprint3_documentation/burndownChart and screen shots of the respective task chart in sprint3_documentation/taskChart. It is noted that the burndown chart shows a steep

decline on the last day of the sprint. Two factors contributed to this otherwise sustainable pace set by the team, the first of those being that a lot of the tasks being open ended such as “update backend tests that are outdated” and the second of which was the relative unimportance of certain task compared to having a smooth running and feature full system. These do not necessarilly excuse the lack of sustainable pace at the end of the sprint but do however offer some justification for the noticable change seen in the burndown chart.

7.5 Collective Code Ownership

This XP practice is easily documented in the pair programming logs. For the second sprint Kyle Van Landingham and Andrew Zolintakis worked on backend php scripts. Alec Fong and Noah Bergman mainly worked on frontend code. Alec Schule worked mostly on black box tests and did miscellaneous work. Looking at he Pair Programming section of the documentation it can be noted that most of the pair programming sessions took place between two team members that worked on different parts of the program such as Noah and Kyle or both Alecs. Those pair programming groups that did work on the same part of the system, i.e. worked on backend both in sprint 2 and sprint 3, those groups worked solely on new features of those sections of the system. These pair programming groups are in compliance with the clarificatino asked in piazza post 320.

7.6 Refactoring

Refactoring was done for each commit that contributed a good amount of code to the system. Refactoring the code in the system is meant improve readability or simplify the code structure. Documentation for refactoring can be found in sprint3_documentation/refactoring where there is a text document named log.txt which contains a log of the refactoring done by each pair programming group during each session that was large enough to accommodate a refactoring.

7.7 Continuous Testing

Continous testing was adhered to during the sprint by running the entire test suite after each significant pair programming session and submitting the results of the test with the code that was tested. These test results can be found in sprint3_documentation/pairProgramming and within each folder in that directory there are files designated as the test results for that commit. Note that certain folders commited earlier in the sprint lack the test outputs either due to neglect or small commits, either way it is noted and was corrected throughout the sprint.

7.8 Create a test for every bug found

A log of creating a test for every bug found during the sprint can be found in sprint3_documentation/testsForBugs.txt. It is noted that there were very few tests for bugs found, a possible explanation for this is that many bugs were found during the running of tests and the

test suite was adapted to check for those bugs but were not documented due to preoccupation with getting the test suite to run correctly and the tests for those bugs were just modified not added into the test suite. It is possible to observe the code changes in the different commits however due to time constraints the team was not able to track each change of the test suite that accommodated for specific bugs found while testing. It is not expected that the grades go through the repository themselves either, this is just an attempt at explaining the low number of test cases listed in the testForBugs.txt document.

8. Scrum Evidence

Pictures of the stand up meets can be found in the sprint3_documentation
4/20/17
What have you accomplished since the last meeting?
Kyle: Andrew and I completed exporting articles list as plain text and started making a word cloud from a subset of articles in the backend.
Andrew: Completed exporting article list as pdf and partially implemented back end for an article subset word cloud
Noah: Worked on refactoring php code with alec
Alec F: Refactored code to be modular and more readable
Alec S: Nothing
What will you accomplish before the next meeting?
Kyle: I will work on exporting the article list as a pdf
Andrew: I will export the article list as a pdf
Noah: I want to start working on generating the subset WC the frontend
Alec F: instructions to integrate into greater codebase
Alec S: Pair program parsing PDFs with Alec F
Are there any obstacles that will prevent you from accomplishing your goals?
Kyle: None.
Andrew: Obstacles depend on how much documentation we can find on this
Noah: None
Alec F: None.
Alec S: None.

4/21/17
What have you accomplished since the last meeting?
Kyle: Andrew and I completed exporting the article list to pdf
Andrew: Completed exporting article list to pdf
Noah: Started work on frontend subset WC
Alec F: Ability to integrate code into greater codebase
Alec S: Pair programmed parsing PDFs with Alec F

What will you accomplish before the next meeting?
Kyle: Subset of articles word cloud
Andrew: Subset of articles word cloud
Noah: Finish frontend subset WC
Alec F: parse pdf's
Alec S: Keep working on PDF parsing
Are there any obstacles that will prevent you from accomplishing your goals?
Kyle: None
Andrew: Sending a lot of information through ajax
Noah: Backend may not be completely done for the subset WC
Alec F: none
Alec S: Shibboleth is difficult to work with

4/23/17
What have you accomplished since the last meeting?
Kyle: finished creating a wordcloud from a subset of articles
Andrew:Completed subset of articles word cloud
Noah: Finished frontend subset WC
Alec F: Attempted to integrate parse pdf with codebase
Alec S: Debugged code to parse pdfs
What will you accomplish before the next meeting?
Kyle: Help with black box tests
Andrew: For conference articles, include only articles with the specified word
Noah: Download WC image
Alec F: Re-test with correct link and find true bug of pdfParser for ACM.
Alec S: Finish getting PDF parse code working
Are there any obstacles that will prevent you from accomplishing your goals?
Kyle: no
Andrew: This could take a long time if few articles contain the word
Noah: Don't know if canvg will work
Alec F: Dependency issues with mechanize
Alec S: Issues with apache permissions and external package version differences

4/25/17
What have you accomplished since the last meeting?
Kyle: worked with noah on downloading word cloud as image and andrew on export as pdf
Andrew: Worked on conference article list
Noah: Progress on downloading WC image

Alec F: Scripts run well with back end
Alec S: Made a little progress on gettings pdf parsing to work
What will you accomplish before the next meeting?
Kyle: get the wordcloud to download as an image
Andrew: Implement all necessary white box tests
Noah: finish WC image
Alec F: Find solution to highlighting pdfs
Alec S: Finish PDF parsing
Are there any obstacles that will prevent you from accomplishing your goals?
Kyle: no
Andrew: none
Noah: Pretty busy working on another project
Alec F: there seems to lack of out of the box pdfs
Alec S: Very busy

4/26/17
What have you accomplished since the last meeting?
Kyle: worked with noah on downloading word cloud as image and andrew on export as pdf
Andrew: Worked on conference article list
Noah: Still working on downloading WC image
Alec F: Scripts r
Alec S: Made a little progress on gettings pdf parsing to work
What will you accomplish before the next meeting?
Kyle: get the wordcloud to download as an image
Andrew: Implement all necessary white box tests
Noah: work with Kyle to get WC to download as image
Alec F: get PDF with highlighted words
Alec S: stream download of pdf from backend instead of link to acm or ieee
Are there any obstacles that will prevent you from accomplishing your goals?
Kyle: no
Andrew: session variables might mess up
Noah: no
Alec F: a lot of research is going to be needed to implement this feature
Alec S: No

4/27/17
What have you accomplished since the last meeting?
Kyle: got wordcloud image to download
Andrew: working on whitebox still

Noah: got WC download as image
Alec F: working on downloading PDF with highlighted words
Alec S: finished download link
What will you accomplish before the next meeting?
Kyle: n/a
Andrew: n/a
Noah: n/a
Alec F: n/a
Alec S: n/a
Are there any obstacles that will prevent you from accomplishing your goals?
Kyle: n/a
Andrew: n/a
Noah: n/a
Alec F: n/a
Alec S: n/a