

# **PHASE 4 REPORT**

## **Team 01: NewsFlash**

### **Team Members:**

*Nicholas Kazlauskas; Dickson Wong  
Miranda Xu; Amna Iqbal; Chandni Sehgal*

## Final State of the Project

For the final iteration of the project, changes were made to the plan to accommodate new features:

- a. Crawling on Twitter – this encompasses a few user stories (see release plan)
- b. Crawling an entire website for articles

Furthermore, integration of features implemented from the previous phases was completed.

- a. Watch lists of RSS, Articles, Websites
- b. Visualizations

Other features had been worked on, but not fully integrated into the UI:

- a. Exporting WARC
- b. A login system
- c. This also however, includes the Twitter API

We ran into similar issues that we did in the past:

1. As mentioned above, some features were not implemented into the UI. In terms of the login system and Twitter related functionality, these were simply not complete and could not be integrated with the rest of the system.
2. This was due to time constraints, as we have faced in the past – largely due to other commitments the team members have.

Looking back at the original release plan, the original plan was not followed at all for many reasons.

1. The plan was completely rewritten to prioritize the customer after the first phase.
2. During phase 2, iteration 2 did not meet very many of the goals due to other commitments for the team members. This resulted in shifting the completion of requirements for subsequent iterations.
3. Furthermore, as the project evolved, more requirements were realized:
  - a. Watch-list, RSS functionalities were realized after the second iteration.
  - b. Twitter functionality was needed after the third iteration.

However, only the order in which the requirements were completed was largely affected. In terms of actual functionalities, many of the requirements were completed.

- a. Database requirements were completed
- b. Crawling functionality was completed
- c. Visual representations were completed
- d. A UI to work with all of this

On the other hand, many of the requirements that were later added (some part of the original – the low priority ones) were not completed.

- a. Crawling Twitter was not completely finished and thus not implemented into the UI
- b. The login system was not integrated with the UI
- c. Data exports – WARC were not implemented into the UI
- d. Other user personal functionalities (such as restricting use) were not implemented at all

In retrospect, one major problem was the time constraint that the team members had; although many of the features were actually implemented – new features added into the system meant that changes and fixes to the system were constantly needed.

For example, during the last iteration, Twitter functionality was needed; however, the database did not support this at all – sources that represented single Twitter accounts could not be added into the database.

On the other hand, during the first iteration, the team had completed a working version of the database – that would allow for additions of the main items (sources, articles, etc); however, we did not expect so many issues would appear with UI (the database could not handle different types of errors) and allowing other classes for formation of easy queries (eg. visualizations getting the needed information was not particularly easy at first).

This shows perhaps that the team does not have the best planning skills for future development – although following the agile principles may mean that we do not focus on planning. If the team were to do it again, we would spend more time (although not too much time – according to agile) on planning for future iterations.

One of the best decisions was to completely change the way database worked – during iterations 2 and 3. Previously, making the correct queries from it was not entirely easy – get functions would need to be called by passing in queries (strings). It was changed to be more intuitive – for example, `get_references` could now be called with the actual source and article ids to get the related references. This meant that all the team members could use this database easily.

Perhaps one of the worst developments was the addition of the Twitter related stories – not that they were not required, but the team set out to do too much for this iteration (in general, that seemed to be an issue). The team did not have enough time to completely correct the issues from previous iterations and implement too many features during the last iteration. If possible, we could have restricted adding only basic Twitter functionality, such as adding a certain Tweet to the database and adding accounts as sources.

If the team had 3 more weeks, we would complete implementation of Twitter into the application and complete testing of all our features. If the team had 10 more weeks, we could implement the features we had removed completely from the plan and those that were not completed in time.

## Final Release Plan

== Cost ==

Unit cost –refers to one member of the team working on the project for a day.

We have 5 members, so in a day, we have 5 units of work that can be completed. But we are not working every day in the week, so we can expect to only have 2 days of the week to work on the project.

For example, in a given week:

5 units per day \* 2 days per week = 10 units of work per week

=====

### Iteration 1 (2 weeks) - Release Date: 15/10/2014

This iteration takes into account the implementation of the essential functionalities needed by the business. These include:

- The search for references to sources within an article and then retaining a list of references that are made inside articles.

This iteration also focuses on the implementation of:

- Keeping track of a list of sources, articles, and keywords.
- Adding the list of sources, articles and keywords to the system,

#### 10.

As John (a researcher), I want to keep track of a list of references made inside an article, so that I am able to analyze the different sources that influence the article.

Priority: 10

Cost: 1

#### 14A.

As John (a researcher), I want to search (for references and sources) within the article, so that I am able to research it more thoroughly.

Priority: 10

Cost: 5

**1.**

As John (a researcher), I want to keep a list of articles (with appropriate tags), so that I can research the influence of certain articles and sources.

Priority: 9

Cost: 1

**2.**

As John (a researcher), I want to keep a list of sources (with appropriate keywords), so that I can research the influence of sources.

Priority: 9

Cost: 1

**3.**

As John (a researcher), I want to save changes to the list of articles and sources, so that I can retain the old data and continue from there.

Priority: 9

Cost: 1

**4.**

As John (a researcher), I want to add a source to the application, so that I can research the influence of sources.

Priority: 9

Cost: 1

**5.**

As John (a researcher), I want to remove a source from the application, so that I can manage my sources.

Priority: 9

Cost: 1

**7.**

As John (a researcher), I want to add an article to the application, so that I can research the influence of sources in that article.

Priority: 9

Cost: 1

**8.**

As John (a researcher), I want to remove an article from the application, so that I can manage the articles that interest me.

Priority: 9

Cost: 1

**15.**

As John (a researcher), I want a display of the list of references a source makes to other sources, so that I can research the influence of different sources and their relation to one another.

Priority: 9

Cost: 2

**11.**

As John (a researcher), I want to manually add references to an article in the application, so that the software can help me research the influence of sources.

Priority: 8

Cost: 1

**12.**

As John (a researcher), I want to manually remove references from an article in the application, so that the software can help me research the influence of sources.

Priority: 8

Cost: 1

**13.**

As John (a researcher), I want to see a list of articles that a specific article links to, so that I can research the article.

Priority: 8

Cost: 1

**6.**

As John (a researcher), I want to change keywords that are associated with a source, so that I can research the influence of sources.

Priority: 7

Cost: 1

*The total cost for this iteration is 20.*

=====

## **Iteration 2 (2 weeks) –End Date: 30-10-2014**

The second release will focus on the implementation of:

- Association of tags to an article.
- Graph visualizations of sources and associated references to one another (including both network graphs and 2d plots)

In addition, this iteration will cater to:

- Recursive reference searching inside an article
- Sorting by category

### **14B.**

As John (a researcher), I want to search (for references and sources) within links of an article, so that I am able to research it more thoroughly.

Priority: 10

Cost: 3

### **22.**

As Jessica (a librarian), I want to categorize data by its' date, title, or keywords/tags, so that I am able to retrieve certain pieces faster.

Priority: 8

Cost: 2

### **16.**

As John (a researcher), I want a graph visualization of references from a source to other sources, so that I can analyze the references that have been made in articles from a certain source.

Priority: 8

Cost: 4

### **18.**

As John (a researcher), I want a graph visualization of references made between different sources, so that I can research the relationships between sources.

Priority: 8

Cost: 4

**20.**

As John (a researcher), I want to see graphical representations (2D line plots) of the number of references made to sources, so that I can analyze the changes to the popularity of sources over time.

Priority: 8

Cost: 4

**9.**

As John (a researcher), I want to change the tags of an article in the application, so that I can research the influence of sources in this article (and to other articles).

Priority: 7

Cost: 1

The planned cost of this iteration is 20.

=====

**Adjustments made:**

User stories 14B, 16, 18, 20 are moved to iteration 3 (see iteration 2 plan and its re-estimations).



### **Iteration 3 (2 weeks) - Release Date: 14-11-2014**

Iteration 3 is geared towards completing the visualizations and updating entries inside the database with RSS feeds and article watch list capabilities. Furthermore, the implementation of various data exports are planned to be completed.

This includes:

- Graph visualizations and plots representations. These pertain to different sources, articles, and the references.
- Watch list of articles
- RSS Feed from websites
- XML and WARC exports

#### **20A.**

As John (a researcher), I want to see graphical representations (bar graph) of the number of references made to a target list of sources, so that I can analyze the changes to the popularity of sources over time.

Priority: 7

Cost: 4

#### **30.**

As John (a researcher), I want to specify the articles that I can get updates on (watch list), so that my research is relevant.

Priority: 7

Cost: 1

#### **31 .**

As John (a researcher), I want to receive updates on the articles that are on my watch list, so that my research is relevant.

Priority: 7

Cost: 2

**19.**

As John (a researcher), I want to see the graphical representations (network connections) of related articles and sources, so that I can better my understanding of the big picture of the public sphere.

Priority: 7

Cost: 3

**35.**

As John (a researcher), I want to add a website to automatically get articles from the RSS feed, so that I my research and information is more complete.

Priority: 7

Cost:4

**32.**

As Jessica (a librarian), I want to be able to store archives of articles and related information in a WARC format, so that the data will be accessible in the future on any system.

Priority: 6

Cost: 3

**23.**

As Jessica (a librarian), I want to be able to export data in a compatible format (e.g. XML) to certain other software, so that the data will be accessible in the future on any system.

Priority: 7

Cost: 3

=====

### **Adjustments made:**

From iteration 2, user stories 14B, 16, 18, 20 are completed during this iteration instead.

User stories 20A, 31, 32, 23 are moved to iteration 4. (see iteration 3 plan and its adjustments)

#### **Iteration 4 (2 weeks) - Release Date: 28/11/2014**

The last release will involve implementing Twitter related functionalities – such as Tweet crawling. Previous features will be improved on: crawling on entire websites, WARC functionality, and adding a log in system.

**44.**

As John (a researcher), I want to crawl an entire website for articles, so that my research becomes easier.

Priority: 8

Cost: 3

**40.**

As John (a researcher), I want to add twitter accounts as sources of information, so that I can research accounts that are of interest to me.

Priority: 7

Cost: 2

**41.**

As John (a researcher), I want to add a tweet to the database as an article if it links to an existing article, so that it can be used in my research.

Priority: 7

Cost: 2

**42.**

As John (a researcher), I want to search for references to other twitter accounts and find links within a tweet, so that I understand the relationships between different sources on Twitter.

Priority: 7

Cost: 3

**43.**

As John (a researcher), I want to be able to crawl Tweets (find links recursively), so that I can understand the relationships between different sources on Twitter.

Priority: 7

Cost: 2

**24.**

As Jessica (a librarian), I want a login screen for the authentication and authorization of users so that the data access is restricted and secure.

Priority: 5

Cost: 2

=====

**Adjustments made:**

User stories 31, 32, 23 were moved to this iteration.

Integration of WARC, Twitter, and a login system were not completed.

Stories planned originally were removed from the plan.

## **User Stories not implemented/worked on**

**27.**

As Candice (an educator), I want to export the data (sources, articles, visualizations) through a web interface, so that the data is more readily available to larger groups of people.

Priority: 6

Cost: 4

**28.**

As Candice (an educator), I want to import the data (sources, articles, visualizations) through a web interface, so that the data is more readily available to larger groups of people.

Priority: 6

Cost: 4

**26.**

As Candice (an educator), I want the allowance for student level accounts to only viewing and accessing of visualizations and queries, so that there will be a prevention of unauthorized changes to the database.

Priority: 3

Cost: 2

## Iteration Plan for Iteration 4

The main features addressed during this iteration are:

1. Integrating the visualizations in the UI
2. Integrating RSS and in general, watch list capabilities into the UI
3. Crawling an entire website (with a limit) for articles
4. Twitter related functionality
5. A login system.

===== Improvements/Changes for previous functionalities planned =====

Stories 16, 18, 20, 20A, 19 (Visualizations)

1. Reformatting code related to network visualizations and plot visualizations: 3 – Dickson
2. Integration into UI: 1 – Nick
3. Adding more tests: 1 – Chandni

Stories 35 (RSS), 30, 31 (Watchlist)

1. Changing database to watch any URL: 1 – Nick
2. Integration of watchlist capability on UI: 1 - Nick
3. Updating tests for RSS: 2 – Amna

===== New functionalities/requirements =====

Story 44 – Crawling a website

1. Adding a crawl for a website: 2 – Chandni
2. Testing: 1 – Miranda

Story 40 – Adding Twitter accounts as sources

1. Changing database and source to accommodate this: 1 – Dickson
2. Testing: 1 – Nick

Story 41 – Adding Tweets to the database

1. Testing: 1 – Dickson

Story 42 – Searching for References/Links in Tweets

1. Implementation: 2 – Miranda
2. Testing: 1 – Chandni

Story 43 – Crawling Twitter

1. Implementation: 1 – Miranda
2. Testing: 1 – Chandni

#### Story 24 – Login

1. Implementation: 1 – Chandni
2. Integration: 1 - Nick

Other duties:

The implementation of Twitter related functionality will require changes to the UI. In particular, crawls related to Twitter will require server to call a different crawler other than web. Furthermore, for our implementation, the user will also need to “follow” the twitter accounts that they are interested in and thus, UI will need to be changed to accommodate this. Other required features from the UI will be realized as the Twitter API is developed: 3 - Nick

#### **Project Velocity:**

The expected project velocity for this iteration is 12.5/week.

Over the course of the iteration, other commitments arose for the team members. The login system was started; however, the integration with the UI of that was not completed.

Furthermore, the particular implementation of the Twitter crawling required many changes to the UI that we did not expect. In particular, it required that the user “follow” accounts for which they were interested in to get new articles. This and other related features were not fully implemented and integrated into the system.

The actual project velocity for this iteration was 10/week

## **TaskBoard**

Our Task Board is available via the following Trello Link:

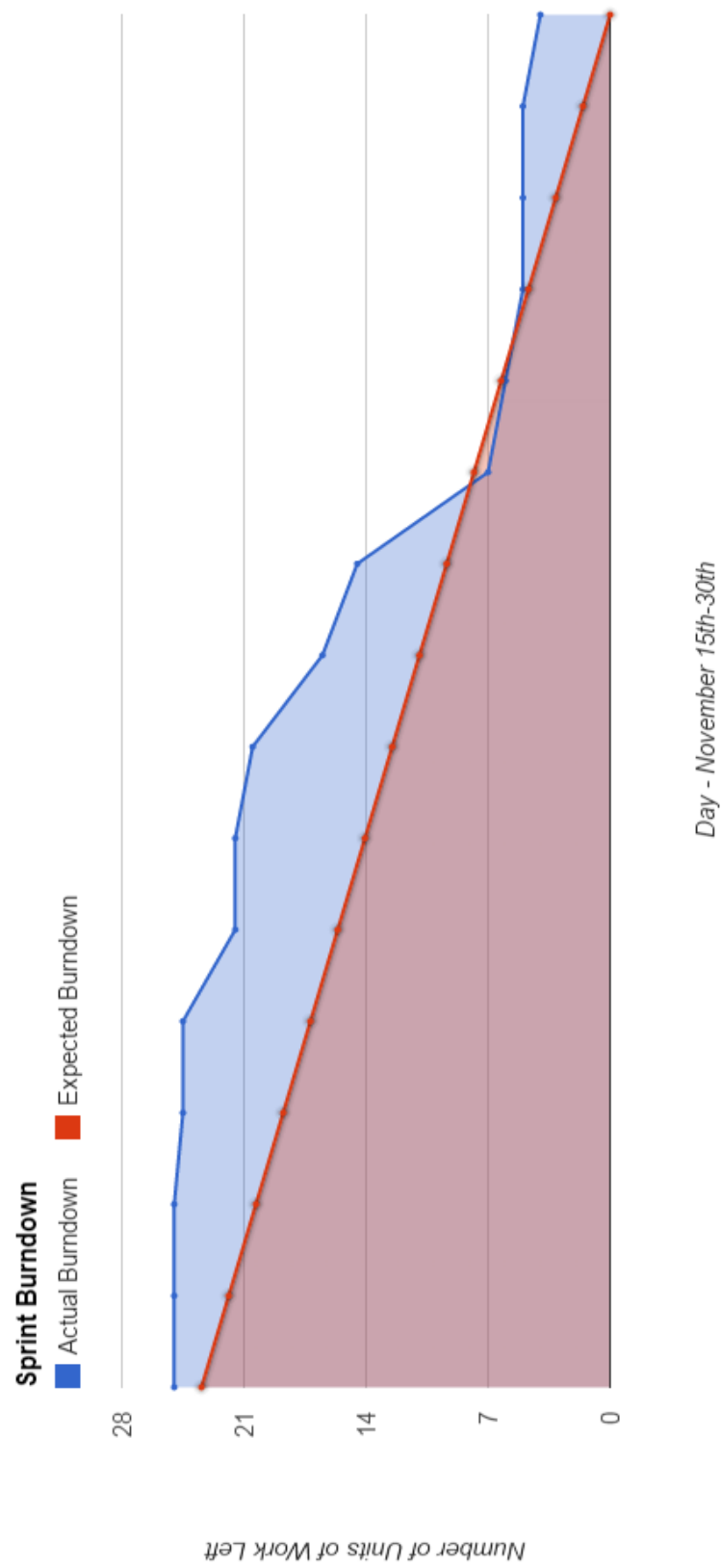
<https://trello.com/b/EWA48O1o/csc01projectteam01>

(The access to our Task Board is through the previously sent system email).

## **Burndown Chart**



User Story	Time (est)	Assigned	Status	Work Done per day per task [units]																								
				Nov	15	16	17	18	19	20	21	22	23	24	25	26	27 - Testing	28 - Testing	29 - Testing	30 - Testing								
16/18/20/20A 19- Visualizations	5	▼	Done									1	2	1														
35. 30. 31 - RSS/Watchlist	4	▼	Done							0.5			1	1	0.5		1											
44 - Website Crawling	3	▼	Done							0.5			1		1													
40 - Add twitter account as sources	2	▼	Done							0.5																		
41 - Adding tweets	1	▼	Done													2												
42 - Reference/link searching in tweets	3	▼	Working													1												
43 - Crawling twitter	2	▼	Working							1						1	1											
24 - Login	2	▼	Incomplete							1						1												
Changing UI to accomodate Twitter	3	▼	Incomplete																									
			Burnout Per Day	0	0	0	0	0.5	0	3	0	1	4	2	7.5	1	1	0	0	1								
			Actual Burndown	25	25	25	25	24.5	24.5	21.5	21.5	20.5	16.5	14.5	7	6	5	5	5	5								
	1.5625	1.5625	Expected Burnout Per Day	1.56	1.56	1.56		1.56	1.56	1.56	1.56	1.56	1.56	1.56	1.56	1.56	1.5625	1.5625	1.5625	1.5625								
			Expected Burndown	23.4	21.8	20.3	18.7	17.1	15.6	14.0	12.5	10.9	9.3	7.8	6.25	4.6875	3.125	1.5625	1.5625	0								



## **Verification**

Verification was done via unit testing, and code reviews.

The code reviews are available under */course/code\_reviews/phase4\_codeReviews*. We used a checklist based inspection, taking care to address all significant elements.

Unit tests can be found in */src*.

## **System Modeling and Design**

Our system follows singleton design pattern throughout. Singleton design pattern occurs when the instantiation of a class has been restricted to one object. Such a class that uses this design in our system would be the Database class.

The Server ensures that all classes that require Database would be able to access this class. For example, Server will be able to pass Database as a reference to NetworkVisual, Plot2D, or web to the functions that need it.

Furthermore, there should be only ever one instance of Database inside the application; Server, at the beginning of run-time, will create only one instance of Database and there is no way to create another instance of it through any function call.

Similar notions can be said about Server.

System Modeling and Design took into account the three most significant scenarios within the system. These can be found in [/course/reports/phase\\_04](/course/reports/phase_04).

### **Improvements from phase 3**

Code reviews:

These are under `course/code_reviews/phase4_codeReviews/Phase 3 Improvements`

Unit testing:

Updated tests can be found within the `src` folder. New tests have been added for visualizations, database, RSS, and crawler/web.