**Content Aware Search Application**

**By**

**Koushik Tammineedi 16121926**

**Jian Wang 12255369**

**Introduction:**

The project is kind of a content aware search engine, when user enters a keyword of his interest; we display the results about the keyword and some similar recommendations depending on the type of the keyboard, from using different API’s, existing services and with some user defined services.

**Overall goal:**

The goal of the project is to develop a mash up mobile app for tracking the topics that user wants to follow. We will update him through email notifications when the app is running.

**Specific objectives:**

1. To develop a mash up Mobile application (possibly for Android OS, iPhone, etc).

2. To get the data related to the topic that the user wants to follow from Facebook, twitter, wiki and also could get the related videos from different API. Try to get the type of keywords from analyzing it with data in Hadoop and data mining using mahout.

3. To create a database of the latitudes and longitudes of the places if anything like events or something going on related to the topic which he’s following in his surroundings, and pointing them on google maps.

4. To Research the option of notifying him through E-mail or by text message when the app is running.

5. To develop and build a responsive User Interface for the ease of use of the software on mobiles, tablets and web.

**Significance:**

Our project is aimed at providing the user the ease of finding/ providing everything- the information about the topics (Such as the history, the trending stuff right now and videos or any research or events going on) that he’s interested in. It saves people’s time as this app makes easier to find everything about the topic at one place. For the testing phase we may need /require initial investment of time by the user.

Sometimes people may like to know about something about a topic of their interest. In traditional way they search in Google. When getting results they need to navigate to each page to know about it, which is time consuming. Our app aims to reduce that process of users navigating to the pages and we display the info relating to the topic from Wiki, Facebook, Twitter and you can also see some pics from Google, Flickr or even videos from youtube, mysapce, metcafe. So technically you have everything in front of you without going to google images, flickr and searching for that or going to youtube to checkout some videos on that.

Users also have the option of sharing the info on Facebook, Twitter and on other social networking sites, and also has the option of inviting people to join in his circle(similar to Google circles or users have the “follow him” like option in twitter).

**Project Background and Related Work:**

There’s no app on it that I found but my Idea is based on many existing individual web application **Pininterest** where people can follow and pin it to their area of interest and now a days there are lot of information on Facebook or Twitter pages on the topic you are interested on. You could watch videos regarding the topic in **youtube** and also you could find events related to that in a different web page. So, my point here everything is there for you but they are separated so that you have to spend a lot of time searching for them and to get to know about it.

**Users & Stakeholders:**

Initially the users will be some students and faculty of the university. They can help the product in testing phase. Their feedback can help us in getting to know whether the product can be deployed on a larger scale or not.

As this is a semester project and part of the coursework, the stakeholders would the faculty and project development team members of the institute.

**Proposed System:**

**Functional requirements**:

**Access module**:

In this module we authorize and authenticate the username and password and it contains the registration page and required validations for username and password to access the system.

**Services Module:**

In this module we have all the services that are required by the website to get the information by accessing internets and we have the services that are already existing and some of them will be user defined.

**API Module:**

In here we have different APIs that we will be using depending on the keyword the user enters. We can have a configuration file that has all the API keys for the API’s that we use.

**Maintenance module:**

We grant the users with permissions to update their contact info and change password through this module. They can see other people’s profile or search for other people and can see what others are following.

**Tracking Module:**

The user will be able to follow the topic that he is interested in and can see who are the other people following the same topic.

**Abstraction (Usability):**

The user has the permission to do one-one personalization and can change the user’s info and can support the tasks of the user such as feedback, navigation and help.

**Maintainability:**

There will be a less affect on the modules by using less no of modules and having less dependencies between composed services..

**Performance:**

Latency is the issue when we are dealing with mobile web development and if the request is refused to be processed the user does not tolerate that. So additional measures should be taken such that the request is properly processed on time and is executed in less than 3 seconds.

**Technological Requirements:**

**Operating System**  : Windows 7 Platform

**Technology**  : Java, JavaScript, JQuery Mobile, Html5

**Browser**  : Firefox, IE 7, Chrome

**Client Side Components**  : AJAX, JavaScript

**Web Server**  : JavaServlet, GlassFish Server

**Programming language** : Java, CSS, Html 5

**Interpreter** : Java Virtual Machine

**Relational Database API** : JDBC, MySQL

**Cloud Systems** : IBM Smart Cloud

**Framework Specification:**

**Mobify.js:** Mobify.js is an open source library for improving responsive sites by providing responsive images, JS/CSS optimization, Adaptive Templating and more.

**Sidetap:** is a lightweight (2k when minifyied and gzipped) and simple framework that allows you to quickly build platform-independent mobile web interfaces. It aims to always deliver the best possible experience for all devices by starting with a basic experience and progressively enhancing it for more capable devices.

Hadoop Framework: MapReduce framework to process big data files

Mahout Framework: Building our defined specific models to classify and cluster big data files (which are from user input or other collected data sources) by using Naïve Bayes, K-Means or other algorithms.

**Specific features:**

**1. Registration and Login :**

We design a login page and registration page, where the students can enter their information and register to the site and then use the login credentials to login into the site. The login credentials that the user provides will be verified with the stored data in the database at backend.

**2. Search:**

We will provide a search box or a list of content where the user can enter or select his topic of interest and depending on the input the results will be displayed on to the user’s page.

**3. Showing results:**

We send the keyword to different API’s and the services that we will be using and collect the results and display it on the page of the user in the respective pages.

**4. Map Reduce Algorithms:**

We may use Map Reduce algorithms as mentioned but not sure to what extent we may implement them.

**System Architecture Diagram :**

****

**User Defined Services:**

Registration and Login Service:

We design the login service in JAVA in order to validate the login of the user. As we store the user details in the database at the backend. We connect to the database from the service to the database through JDBC.

Actors: USER



**Service specifications:**

**Operational description:**

In this the user clicks the “follow” button to follow a particular topic. The topic along with the username is sent to the tracking service and then the service checks for thr username in the database and if user exists we have to store the topic in the database and inturn we can return the other user wo are following this topic if exists any. Once the user clicks on other user links we can direct them to see their profiles and the topics the user. Once the user clicks on topics, he will be directed to wiki page, facebook page, etc.

**Input/output for services :**

Input: Username, topic

Output: Other users following the same topic

**Constraints/exceptions:**

The user cannot modify any contents of other users.

The user has to connect to facebook inorder to network with others

**Priorities :**

The significance of this service is high and the difficulty to design this service is also high.

**Registration Service:**

The user can register into the website before he has permissions to do anything. We will ask the user to enter his personal details such as email, phone, Facebook name etc. If we want to store the social networking links of the user we can provide/authorize the login through Facebook or twitter or Google.

More new Services will be soon added to the document in the respective increments

**Existing API Services:**

Our application will be using some of the existing Services developed by Google, yahoo, Flickr, Wikipedia, FourSquare, etc.

**Name:** FourSquare API

**Description:** It is a kind of service for the mobile, especially smartphones to get the information about the events and you could see whether any of your friends are heading to the event and so on. This all could be done by using FourSquare API services.

**URL:** <https://foursquare.com/>

Name: Google API

Description: Google has many AOI, out of them we mainly use Google Search, Google AJAX Feed, Google Maps, etc

URL: [www.developes.google.com](http://www.developes.google.com)

Name: Weather API

Description: Getting the weather and displaying it on the page of the user based on his location(Longitude and latitude).

URL: <http://openweathermap.org/API>

Name: Twitter API

Description: It’s a social networking site when you can get info of the topic that you search for in the system and also the information about what are the trending topics under discussion.

URL: [www.dev.twitter.com](http://www.dev.twitter.com)

Name: FaceBook API

Description: It is also a social networking site where you will know if anyone that you kmow is interested in the topic or the discussion you are interested in and you can network to them and know them better.

URL: [www.developers.facebook.com](http://www.developers.facebook.com)

Name: Amazon API

Description: We can get the results of some goods that the user may be interested in buying if at all his topic of interested is related to anything that can be bought in sites like amazon.

URL: [www.developer.amazon.com](http://www.developer.amazon.com)

Name: Flickr API

Description: We use this to display the images that the user type in the search box of the system.

URL: [www.flickr.com/services/developer](http://www.flickr.com/services/developer)

Other Services: Yahoo API, Ebay API, etc

**BigData sources:**

* [http://usgovxml.com](http://usgovxml.com/)
* <http://aws.amazon.com/datasets>
* [http://databib.org](http://databib.org/)
* [http://datacite.org](http://datacite.org/)
* [http://figshare.com](http://figshare.com/)
* [http://linkeddata.org](http://linkeddata.org/)
* <http://reddit.com/r/datasets>
* [http://thedatahub.org](http://thedatahub.org/) alias [http://ckan.net](http://ckan.net/)
* [http://quandl.com](http://quandl.com/)
* [Social Network Analysis Interactive Dataset Library](http://www.growmeme.com/overview) (Social Network Datasets)

The “**Tiny Images”** dataset accounts for 227 GB of image data and 57 GB of metadata.

One of the biggest Images dataset is ImageNet.

The **MOBIO** dataset is about 135 GB of video and audio dataThe audio and Video Dataset that has about 135 GB of data

For academic researchers,the yahoo webscope provides several GB’s of data including Flickr image dataset of 85GB .

**Google** has provided a dataset where words are mapped to Wikipedia URLs (i.e., concepts).

Regular huge data dumps are made by FreeBase.

**Wikipedia** made a dataset available which has all the edits done to the articles in wiki.

There is a 1.75 GB data of wiki limks made available by GOOGLE.

[http://imat-relpred.yandex.ru/en...](http://imat-relpred.yandex.ru/en/datasets)  
 <http://www.icwsm.org/2011/data.php>  
 [http://lemurproject.org/clueweb0...](http://lemurproject.org/clueweb09.php/)  
 [http://wiki.freebase.com/wiki/Da...](http://wiki.freebase.com/wiki/Data_dumps)  
 [http://download.freebase.com/dat...](http://download.freebase.com/datadumps/latest)  
 [http://www.kaggle.com/c/wikichal...](http://www.kaggle.com/c/wikichallenge/Data)  
 [http://webscope.sandbox.yahoo.co...](http://webscope.sandbox.yahoo.com/index.php)  
 [http://americannationalcorpus.or...](http://americannationalcorpus.org/OANC/index.html)  
 [http://kddcup.yahoo.com/datasets...](http://kddcup.yahoo.com/datasets.php)  
 [http://horatio.cs.nyu.edu/mit/ti...](http://horatio.cs.nyu.edu/mit/tiny/data/index.html)  
 [https://proteomecommons.org/data...](https://proteomecommons.org/dataset.jsp?i=72639)  
 <http://redd.csail.mit.edu/>  
 [http://www.1000genomes.org/ftpse...](http://www.1000genomes.org/ftpsearch/)  
 <https://www.idiap.ch/dataset/mobio>  
 [http://www-nlp.stanford.edu/pubs...](http://www-nlp.stanford.edu/pubs/crosswikis-data.tar.bz2/)  
 [http://stat-computing.org/dataex...](http://stat-computing.org/dataexpo/2009/the-data.html)  
 [http://blog.archive.org/2012/10/...](http://blog.archive.org/2012/10/26/80-terabytes-of-archived-web-crawl-data-available-for-research/)

**Design of Mobile Client:**

We design the mobile client in HTML5 and use JavaScript, Jquery mobile for the better UI and also use a lot of different frameworks for CSS, which would help us/ aid us in making the web app good looking and user-friendly.

**Features:**

- Advanced forms

- Geolocation API

- Canvas Drawing

- Offline support

- Real time access to Calendar, Email, and Address Book

**Technologies:**

HTML5

JavaScript

JQuery Mobile

CSS3

Java Servlet, Glassfish Web Services

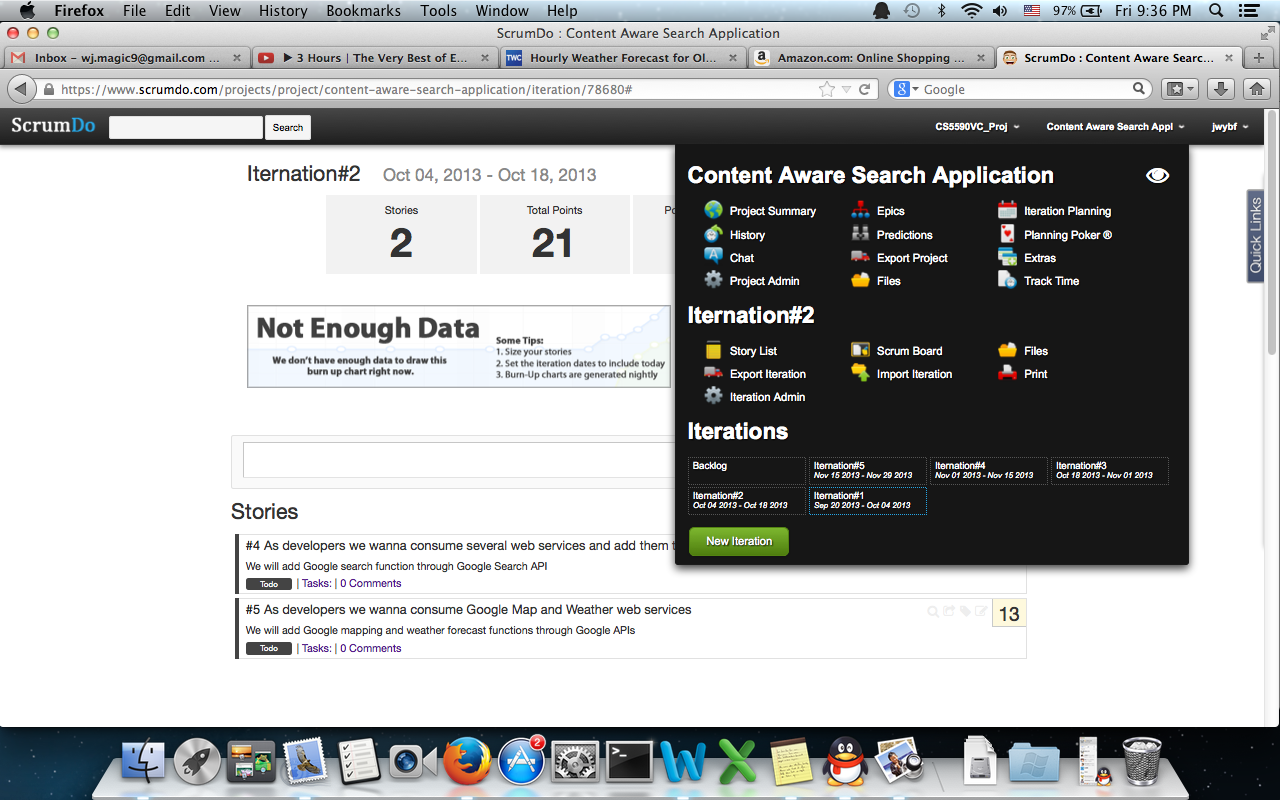
IBM cloud Hadoop systems

Mahout

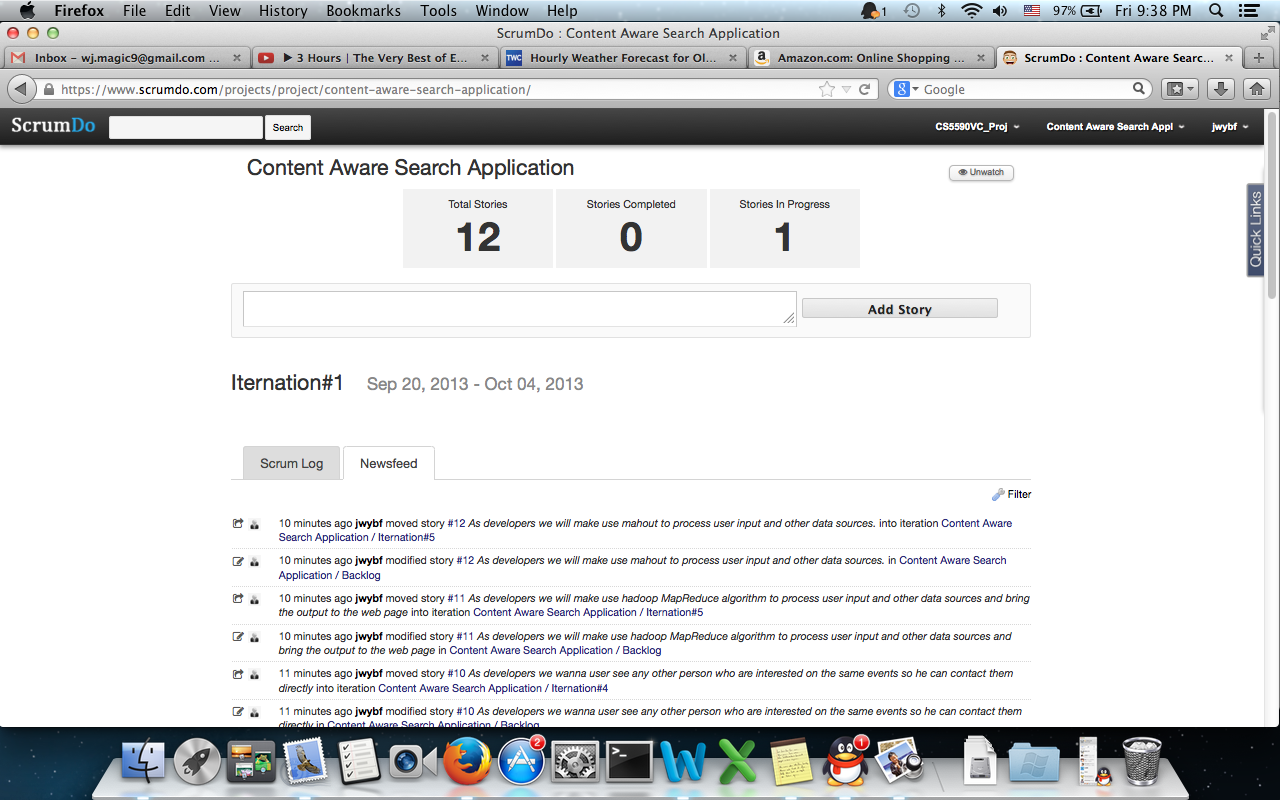
**Project iterations:**

According to our design and planning on this project, we break it to 12 user stories with weighted points, and will implement it within 5 iterations (two weeks as one iteration, starting from 09/20 – 11/29/2013).

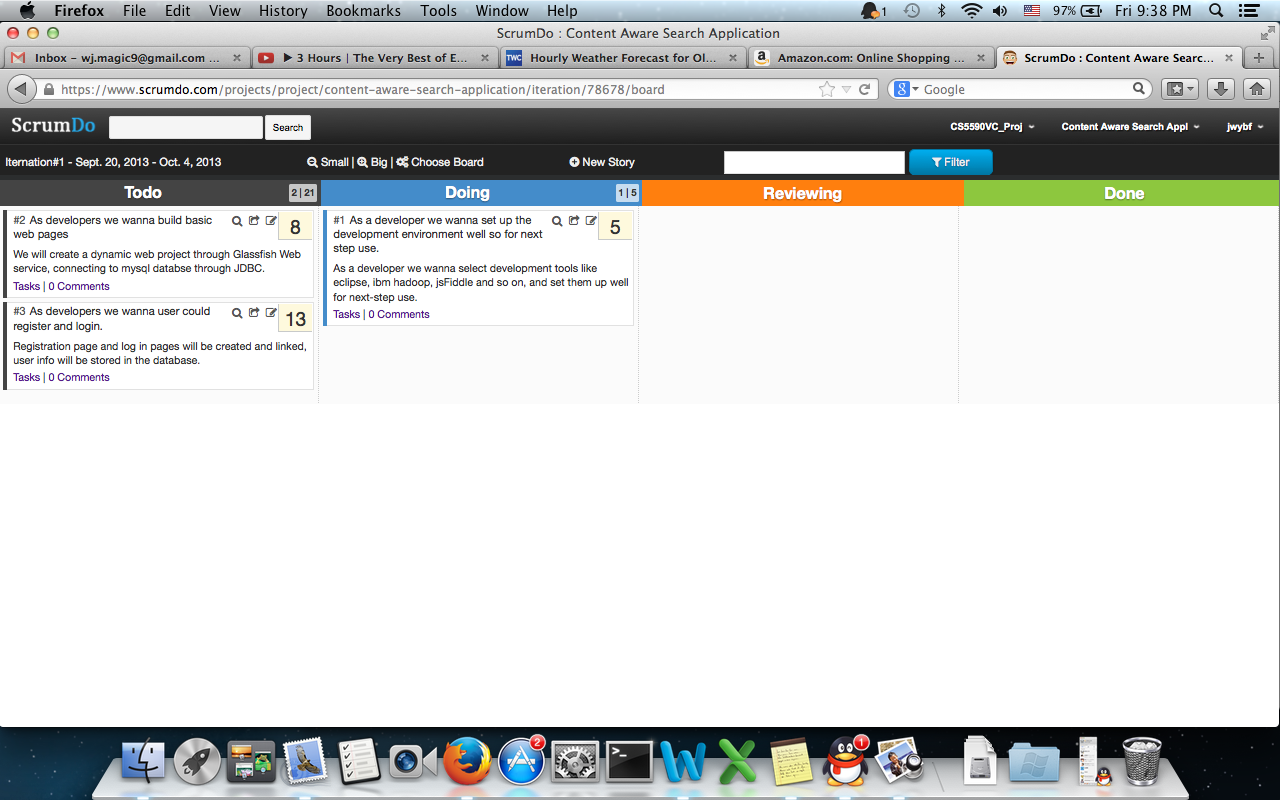
We use ScrumDo tools to keep tracking the status of these user stories.



The below picture shows the details of 12 user stories we created.

****

This is the Scrum Dashborad to show the details of user stories and status of it.

****