

Development Document: BIG NEWS

COMS W4170: User Interface Design

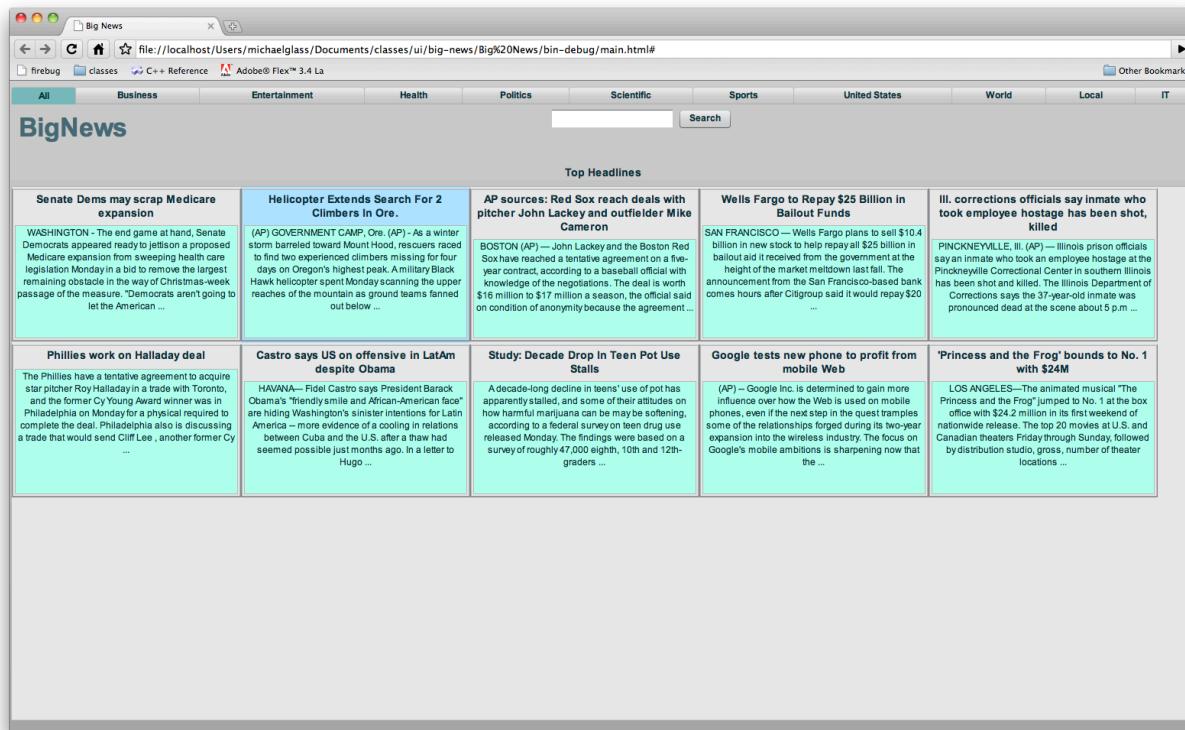
Team: Bow Flex Designers

Team Members

Waseem Ilahi (wki2001@columbia.edu)

Michael Glass (mgg2102@columbia.edu)

David Flerlage (dff2103@columbia.edu)



Overview

This development document contains the high level description of the development process for Big News, implemented by the “Bow Flex Designers”. It will mention the roles of each team member and then go on describing various decisions/steps taken while implementing the system, including the prototyping phase and figuring out the target user groups.

1. Process

The development team consists of three members. The task for actual development was divided such that:

- The entire team sat down and figured out a target user group. Then took a couple of personas from the user group and wrote a use case scenario for each.
- Now that we had the target in sight, we started figuring out a UI for the system, based on the use cases. The lo-fi prototyping technique was used.
- After deciding on a design that will follow the Ten Usability heuristics, the implementation phase started. Now, one member was assigned to research the Bing API, especially the NEWS source type and figure out the proper way to use it with flex.
- Meanwhile the other two members started working on the UI for the system.
- First, very simple UI was created and very basic query returned. After that, improvements were made and the various proposed functionalities were added to the system.
- This document was completed during the final steps of implementation, integration and testing.

2. Target Users

Big News will be a news searcher for people with visual and motor disabilities. As a side objective, we'll also try to make our application usable for people not completely familiar with computers. Perhaps we will split up these feature sets, so that in-program help for new users doesn't get in the way of speed and usability for power users.

Personas:

Persona: Tootsie Glass

Tootsie Glass is not a computer user. She is a great typist, but she can't use a mouse, and is unwilling to learn. She's 94, I don't really blame her. Her grandson has tried and tried again to get her to start using a computer, and she's perpetually resisted. She reads the Times every day, and skims the paper for references to anyone she knows - her son has been in the news recently. Her grandson, Michael, catches her doing this, and sees an opportunity to make a computer application that she might even use.

Use Scenario for Tootsie Glass

Tootsie sits down at her news console, (the application is run full screen, and so the computer appears to only run Big News). She types in "cookies" and sees a slew of articles about cookies in the recent news, displayed as if it was in a newspaper. She arrows down to one of the results, double clicks, and reads an article about thanksgiving and eating. She again types another query, her daughter's name, "Ellen Glass" and finds nothing of interest, however on the side of the screen, the application tells her, "you last searched for cookies", and is reminded to get back to preparing for thanksgiving.

Persona: Joe

Joe, 50 years old, is an airline pilot, but has been on disability leave for several years because of a plane crash he was in, leaving him with visual and motor disabilities. Joe is an avid reader and spends a good deal of his time reading current news. Unfortunately, the print in the newspaper is too small to read, and most websites are too difficult and too time consuming to navigate. Joe wants an easier and quicker way to read the news.

Use Scenario for Joe

Joe starts up "Big News" with the intent of looking up the scores and statistics from the most recent game between the Denver Broncos and the New York Giants. He opens up the program and clicks on the sports category. He looks at the sports headlines, but does not see any information on the game he is interested in. He then decides to execute a search query, specifically, he types in "Giants scores". A bunch of news results come up. He looks through a few of them and finds some interesting information on the Broncos. To this end, he types in "Broncos scores". The first search results that appear are those headlines that were common between his previous searches. Unfortunately, the Giants and Broncos have been playing against each other for years, resulting in some of the news articles being outdated. He decides to sort the results by their date of publish. He clicks the "sort by time" button, reorganizing the data so that the most recent articles are displayed first. He reads a bunch of articles and is very satisfied with his experience, except that he remembers that he forgot to read an article from the previous query ("Giants scores"). He moves the mouse to the right side of the screen to the previous searches box, locates the desired query, and clicks on it. All of the search results from his previous search are displayed. He reads the article he forgot, and then leaves.

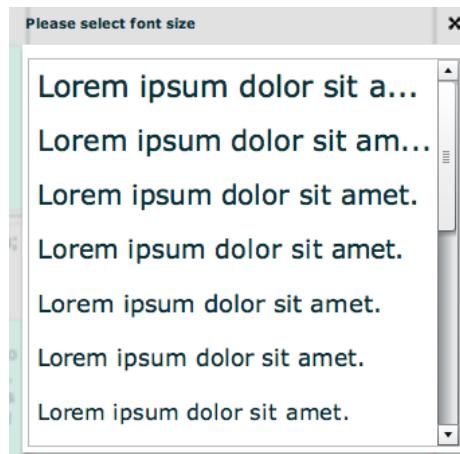
3. Design Decisions

While designing the application and then in the implementation phase, we had to keep in mind the target users and the ten usability heuristics. So, the system was designed following the guidelines set by these heuristics, while keeping in mind the target user group. This section of the document “evaluates” the system, based on the ten heuristics and the user experience as a result of that.

First of all, the requirements provided in the assignment had to be satisfied. We created a search “system”, that allows the user to search the news using Bing API. Of course the user doesn’t know about Bing API and doesn’t care. We also provide the user the opportunity to look at the news; they have searched for recently (in the current session).



The user can choose from ten different topics to search for specific NEWS related to those specific topics. Also, we provide the users the opportunity to adjust the size of the “page”, since our target users group consists of people with the vision problems. To make it easier for all kinds of users, when the program starts, it offers the users the ability to select a text size that is readable for them.

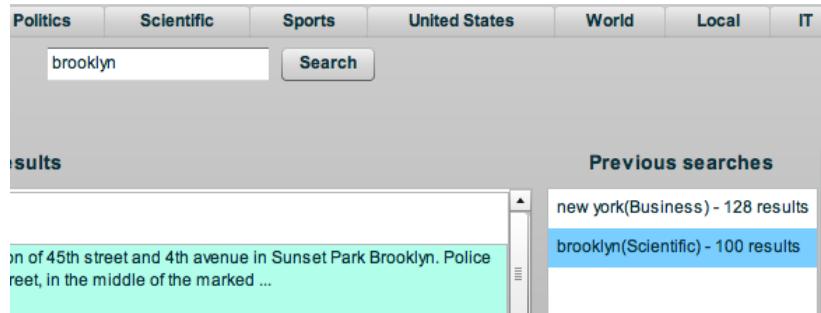


Now the document will list the ten usability heuristics and evaluate the “Big News” system, based on each heuristic.

i. Visibility of system status

“Big News” is a search system. When a user searches for something the results get shown on the screen, or a message comes up saying no results found for this search. Also, when user clicks on one of the links in the history view list, the selected “item” is

highlighted to show the user, which “query” they are looking at, and the query box (where the search term is written) is also populated until manually changed by the user



Whenever the system is busy running accessing the web service, we notify the user with a graphical “spinner” geographically close to the location where they caused the web-service access. When they create a new query or load headlines, we spin near the top of the application near the search bar. When they access a new page from a query, we spin near the pagination buttons.



ii. Match between system and the real world

All the terms used in the “category list” and anywhere else on the system are easy to understand for anyone, who can read and understand English. The headings are clear and the same as used in the news papers and major search engines.

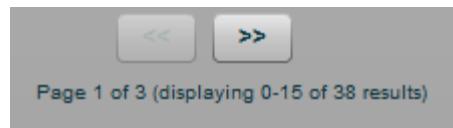


iii. User control and freedom

The entire process for Big News is on “one” page inside the browser. As far as the navigation within the search queries is concerned, the user is provided with the next and previous buttons to maneuver through the search results. Also the “Previous Searches” has a list on previous search results that the user can look at. If the user double



clicks on a “NEWS item”, they navigate away from the “Big NEWS” in a separate window, the original window remains unchanged. After reading the complete story, the user can simply close that window and start looking at the other news inside BigNews. Even with the category selection at the top. The search term remains in the search box, until it is overwritten, so to change the category for search, simply choose one of the eleven options and click search.



iv. Consistency and standards

Big News follows the standards for any search engine. The search button, the headings for each result and the “Headlines” heading all follow the basic standards, for a news search engine. There is no “fancy” terminology used in BigNews. Everything is plain and simple.

v. Error prevention

Because of its simplicity, BigNews makes it very hard for the users to make errors. The possible “errors” are in search terms. BigNews doesn’t autocorrect the terms, for that is not the required feature of the system. Other than that, there is minimal possibility of users committing errors.

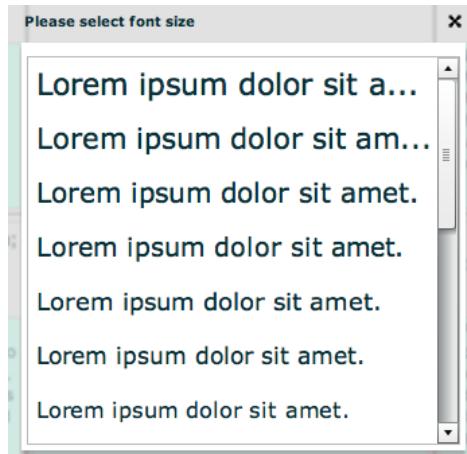
vi. Recognition rather than recall

One beauty Big News is its simplicity. Even in its simplicity, Big News provides a lot of functionality to its users. However, because it is so simple, it allows the users to do whatever they want, without asking them to remember a lot. The possible categories of search are visible to the users. A user can choose any one just by clicking on it. Also the previous searches for the session are all visible on the right side. The user can simply click on any one to see the results for that query. Everything is there in front of the users.

A screenshot of the BigNews search interface. At the top left is the logo "BigNews". To its right is a horizontal menu bar with tabs: All, Business (which is selected and highlighted in blue), Entertainment, Health, Politics, Scientific, Sports, United States, World, Local, and IT. Below the menu is a search bar containing the text "Brooklyn" and a "Search" button. On the far right of the header is a "Adjust Font Size" slider. In the center of the page, a sidebar titled "Previous searches" lists two items: "Brooklyn(Business) - 38 results" and "New York(Local) - 155 results".

vii. Flexibility and efficiency of use

For users that cannot use a mouse, or for users who simply want increased efficiency, the system has a number of useful keyboard shortcuts, and all UI elements are intuitively traversed with arrow keys and the tab key. The user can jump to different elements of the system with a quick keystroke, documented in the user manual. Furthermore, for users with limited eye-site the system supports multiple font sizes, and prompts the user for a comfortable size on startup.

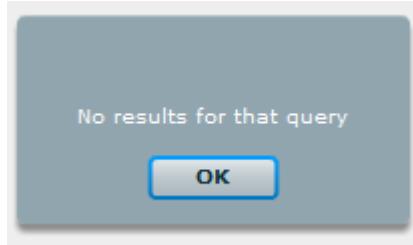


viii. Aesthetic and minimalist design

Everything is to the point. The search results show the title and a little snippet of the news body. Big News does what it is suppose to do and doesn't try anything complicated. It is simple and it is neat and it does what it is suppose to do.

ix. Help users recognize, diagnose, and recover from errors

The possible errors are user typing in the wrong search terms. In which case, they can try again. If the search doesn't return any results, we inform the user that it didn't return any results. If there is a system/network error, we tell the user that and ask to try again.



Furthermore, we keep a history of all user queries. If they want to return to a previous query, it's as simple as navigating to that entry in the history list.

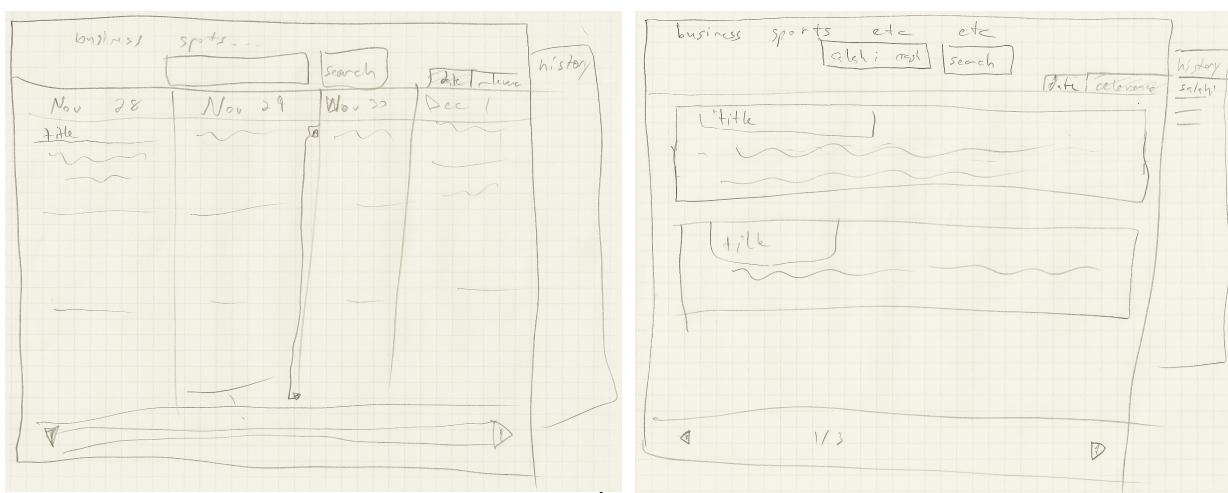
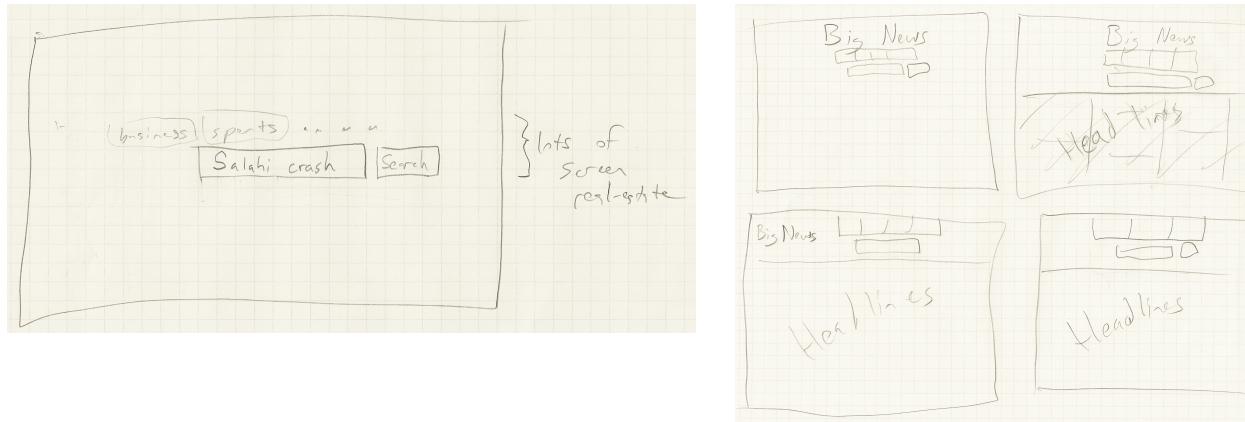


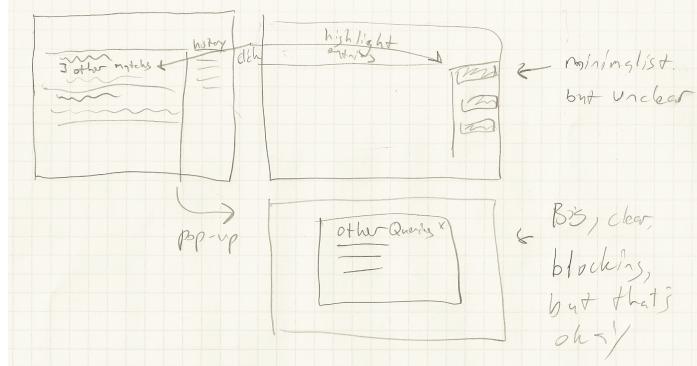
x. Help and documentation

We are providing a User Manual with the system. However, as it has been mentioned various times before in the document, the system is very simple with all functions visible. The user can simply look at it and know what to do.

4. Prototyping and testing process

As it has been mentioned above, right after the user groups and personas were decided, the prototyping process began. Lo-fi prototyping was done. The design is very simple, so there was no need to go through very complicated process.



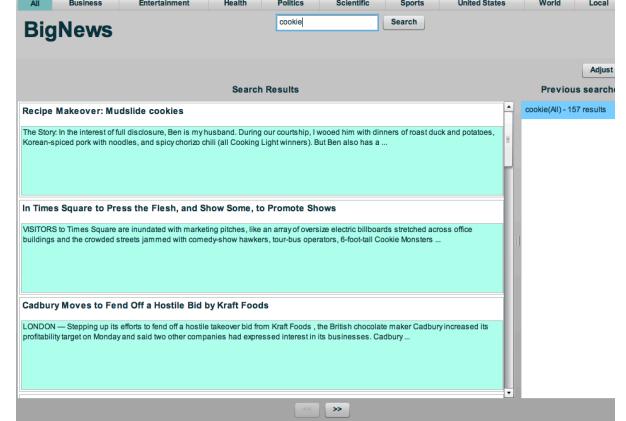
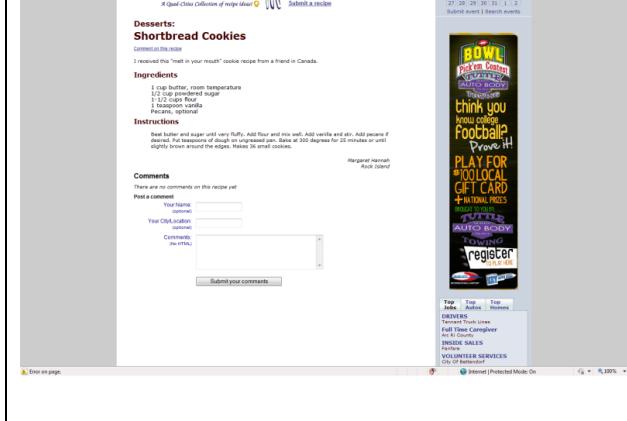
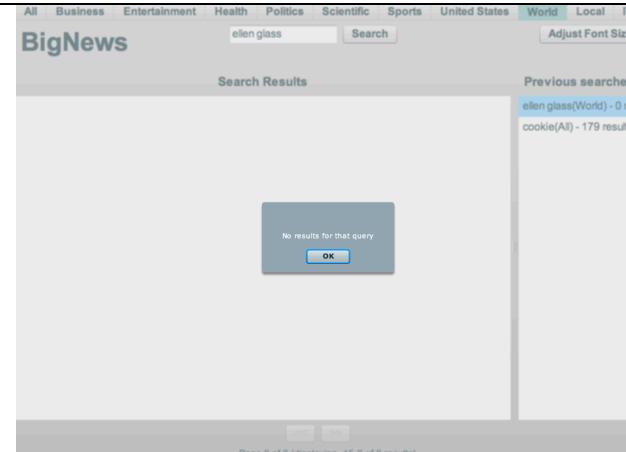
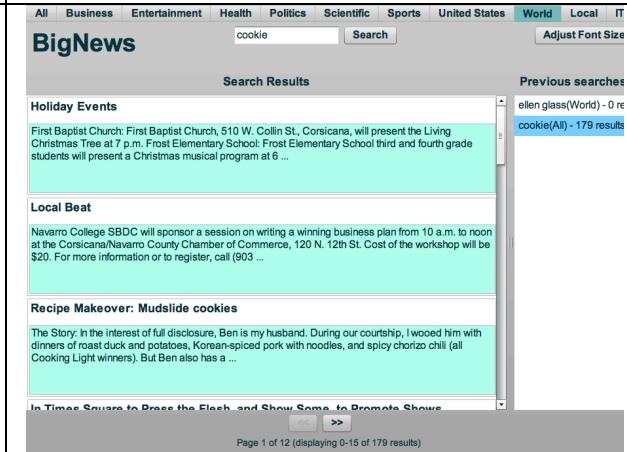


Low-fi prototyping allowed us to quickly layout components, and in the cases where we had multiple ideas for a specific part of the site, allowed us to quickly sketch out how alternatives would look as well.

As far as testing is concerned, we tested the system ourselves. The team members inspected all possible user actions. Errors were found and then corrected. When a member would commit the changes, the entire team would run through a list of all proposed functionality to test the system and see if anything new was broken (or anything broken, fixed). This process prevented a number of bugs from making it into our final system.

The following screenshots correspond to the use case scenario for “Tootsie Glass” described in the above section.

<p>When she logs in, she sets her text size</p>	<p>When she chooses her font, she sees the headlines</p>

	
<p>Tootsie types in “Cookie” and clicks search. She sees this.</p> 	<p>She opens a search result she likes</p> 
<p>She types in “Ellen Glass” but it yields no results</p>	<p>She clicks on the “cookie” entry in previous searches.</p>

5. Software Engineering

Obviously the application was created using Adobe Flex. The Bing API was used to get the news. Flex Builder’s “import web service” functionality parses WSDL web service definition files and generates equivalent Flex classes to interface with the web service in a action-script native way, without having to understand WebServices or WSDL.

<http://api.bing.net/search.wsdl?AppID=YourAppKeyHere&Version=2.2>

To use various animations, we used the AS3Gif by ByteArray (www.bytearray.org/?p=95&cpage=3) combined with an mxml wrapper for It available here (<http://www.dobieag.com/?p=8>)