Indiana University

UReader

IDP Project 2

Ellis, Holt
Hill, John
Lipton, Evan

08

Table of Contents

Executive Summary

Understanding the problem for this particular assignment was vital to the success of the group. We were to design a way for an RSS reader to be integrated into the ever popular web browser, Mozilla Firefox. Additionally, we were to redesign our overall reader to incorporate the same ideas into the new iPhone with the constraints that it carries. The main concern of this problem was to allow for context to be readily available to individuals who wanted more out of their reader, that is, was not satisfied with only being able to view the reader posting, but perhaps those that may want to read previous posts or even visit the source's website for a particular post.

We used two personas that helped us to form rationales by sketching possible design ideas in between meetings. During our meetings, we met, discussed what we had on our sketches, why we thought those particular features may help the cause of the project. After doing this many times, we eventually come to a consensus as to what our prototype might look like. We determined that allowing one to access both previous posts from the same source, and allowing one to access the website, would be the most simple way to allow the user to access the context of the site, our mantra being 'simply context.'

We designed UReader as a feature of Firefox, creating a new interface to the browser. To access the reader at any time, simply push the RSS feeder button within the navigation area of the browser. From here you are able to navigate to posts of your liking, including new posts, all posts, and next and previous post if you were at a website in which you were subscribed to. If you were at a website that provides an RSS feed, but were not subscribed, then a button would appear in the URL address bar that would allow for a simple one-click subscription. Within the feeder itself, we provide numerous settings that allows for the user to control how the reader displays the context of the post. We implemented the features that we came up with for the browser into an application for the iPhone/iPod Touch, while keeping all external links from the application within an inline browser, and keeping many of the multi-touch gestures used for viewing purposes intact within our design.

Personas

As we created our personas, our group wanted to focus on creating an average character that would be found within our user group. As we were discussing what type of person would be using this RSS feed, we realized that technically savvy individuals might have an easier time interacting with any logical design that we may come up with. Therefore, to avoid designing for people who may have an easier time adapting to a poor design, we wanted to focus on individuals who may not have such an easy time adapting. For this reason, we opted to create personas that have less of a technical background. We decided on this so all users of the RSS feed may find it simple, and more importantly enjoyable to use. We then determined that the age range for major use of this

type of technology would be 16-50 years of age. We determined this by noticing that many elders have a difficult time adapting to new technology; therefore it would be more unlikely for an individual over the age of 50 to use our particular design. We included adolescents as low as 16 because our design is meant to help individuals organize their blog posts, and students about halfway through high school tend to develop their organizational skills. As well, since the technology was developed during a child's lifetime, they'd be more prone to using the technology. With that in mind, we'd like to introduce our two personas, Katelyn Anderson and Eduardo Costa.

Katelyn Anderson

Katelyn is a 23-year-old alumnus of the University of Illinois in Urbana-Champaign. Katelyn was a sports marketing major and since graduating, has taken a job on the advertising team of the new television channel, The Big Ten Network. She lives and works out of Chicago, enjoying both her social and professional life. Katelyn has a serious boyfriend in town, and enjoys planning weekends with him in Michigan, where her family is from. She primarily uses Google Reader to keep track of sports news to help her gain an edge in the office as far as new ideas come, but has more recently started added blogs about some of her hobbies, such as scrap booking, crafting, and photography.

Eduardo Costa

Eduardo is a middle aged Portuguese native. After graduating nearly top of his class at the University of Texas, Eduardo moved to California to work in a small technology based company. He has always had a passion for anything technical, and this over the years has helped him create a foundation within his company. He was recently promoted to vice president of Finance. Besides his very successful professional life, he also has an adoring wife, Amber, whom he met at UT, and a young son, Marcos, who had just had his 6th birthday. Eduardo mainly uses Google Reader to keep up with the ever expanding finance market whether he is at home, at work, or traveling. He has recently discovered that it is more than useful, because he can also add the latest news from the Portuguese national soccer team, a team he has followed since his childhood.

Design Rationale

RSS Button

Having the RSS reader button (Figure 1.8) in the main toolbar of the browser is at the core of our design. We did not want to give our personas just an add-in or a URL that they would visit to access our reader. We wanted to design a completely new browser for Katelyn and Eduardo that kept them connected to their sources at all times, without taking away the context of their blogs. With this in mind, we wanted a smart browser that would have a reader that was always "on" in the background.

The RSS button on the main toolbar of the browser would have a small circle above the icon that would display the number of new unread posts. We thought this feature would let Katelyn know when she has new sport story she could read, or a new craft project she should work with. This would be a setting of the browser, because we also thought Eduardo may have too many stories that were new, and may not want a notification of unread posts.

At any time our personas were surfing the web and they wanted to access their RSS posts and feeds, they would simply click the RSS button. The dropdown that would be shown would have four options: *Next Unread Item*, *Previous Item*, *All Unread Items*, and *All Items*. They would act as in the following way:

Next Unread Item	This would take the reader to the newest unread item from their sources in full site mode. (Figure 1.4)
Previous Item	This would take the reader to the older (previous) post from their sources in full site mode (Figure 1.4)
All Unread Items	This would take the reader to the list view of all Unread Items. (Figure 1.2)
All Items	This would take the reader to the list view of All Items. (Figure 1.2)

We wanted to allow for quick and easy access from the browser toolbar to simplify the reading process. This design keeps our personas constantly connected with their feeds no matter what they are doing on the web.

Subscribe Button

Since our browser recognizes when a site has an RSS feed, we wanted to allow for subscribing to new feeds quickly and easily. We thought that if Katelyn was checking out a new craft site recommended to her by her Mom and she enjoyed it, she should have an easy way to add this source to her feed list. We recognized that in many current browsers RSS feed icons show in the address bar of a site that contains a feed. We wanted to keep with this idea and put a "Subscribe" button in the address bar (Figure 1.3). Our browser would know whether or not a site is currently subscribed to, and only be visible on sites that are not already one of the reader's sources. When the subscribed button is

pressed, the browser would then ask the user to either categorize the site or keep it unsorted.

Subscriptions Menu Item

Since our reader is now built into the browser, we wanted to put a Subscriptions menu item in to give additional options to users (Figure 1.9). We thought that this is a place that is very common for people to go when they can't accomplish a task, or are looking for additional options. Also, when thinking about how our personas will use our reader, we realized that there could be times when Katelyn does not want to see all of her items, but only ones related to sports (for example, in her sports folder). By accessing the subscriptions menu item, she can see all of her folders and can click on those folders to only see the related stories. She can also access individual sites this way as well. This menu item also gives an individual the ability to manage their subscriptions and change reader settings.

List View

The list view (Figure 1.2) is accessed within our reader either through the RSS dropdown or the Subscriptions menu item. When designing for our personas, we wanted to keep the meat of why RSS was accepted in the first place. Really Simple Syndication means quick access to stories without all of the graphics of the original site. We did not want to take away this functionality, so we allowed our audience to have the best of both worlds.

In list view, the article length of each post depends on the article length slider (See *Article Length Slider* section below). This allows Eduardo to simply read articles in its simple text form if he chooses. Each post name would be a bolded link with the source and author of the post in smaller text below the link. We thought that this was an important aspect to our design so that when a user is scrolling through multiple posts in list view, they can easily see which source they are from. If Eduardo would like to, he clicks on this link and is taken into full site mode (Figure 1.4), which gives full context to him.

Comments Link

The comments link (provided at the end of each post) is critical in our design. It provides important context for each article (post) to the user. When designing the comments link it became clear to us that Eduardo and Katelyn would have a much better experience if they were able to read the comments within the reader itself. However, it was also clear that our personas would not read the comments to every article they read or skimmed over. With this knowledge we decided to make the comments appear inline (where possible) when the comments link was clicked. At that point the comments would 'slide' into view expanding the container for the post itself. The animation provides transparency within the reader so that users would not be confused. The comments themselves would then appear below the given post with as much information as possible from the source. This allows users to read the

comments within the given context of a post, all without leaving the current page.

Add Comment Link

Adding a comment to a given post is a very important part of our personas lives. It allows our personas to be more involved and connected to the sites, posts, and authors they enjoy. When designing the reader we would have liked for users to be able to post comments to a given article from within the reader interface itself. This would have allowed users to stay within their current state and provided a much more seamless experience. However when discussing the possibilities of doing so we realized that adding a comment inline would not be very practical. Therefore, we decided to have a simple link that reads “Add Your Comment” to each post. When clicked, this would open the site itself where users could enter their comments. Given that our reader is smart enough to determine if you are on a site to which you are subscribed, the process of adding a comment and moving forward to your next article is near seamless. For instance, when Katelyn wants to add a comment to a post about knitting with a question, she simply clicks the “Add Your Comment” link. She is then transported to the site where she can add her comment. Now when Katelyn wants to navigate to her next unread item, she simply clicks the reader icon and chooses “Next Unread Item”. This experience is a near seamless experience for the user, while still being practical.

More Posts From Source

Another crucial way that our design adds context to posts is the “More Posts From Source” button. This button is especially important to users such as Katelyn who will at times need to quickly navigate to an earlier or later post from a particular source. This comes in handy when Katelyn is reading a three part series on knitting sock puppets. Now she is able to simply click on the “More Posts From Source” button and then choose another article from the drop down menu. The menu itself is a list of three to five headlines arranged by date/time corresponding to the post currently being read. It shows the current headline in a grayed out fashion with any two newer posts above it and any two older posts below it. So, if Katelyn is reading the second part of this three part series and needs to get back to the first, she clicks the button and then selects the first part from the drop down. This post is then brought to view in place of the post currently being read. When she is finished reading this post she can simply go back to the button and select part two once again.

Article Length slider

Sometimes Eduardo has so many unread items that he can’t possibly read them all. So, we designed an article length slider (Figure 1.2) that allows users like Eduardo to select the length of the articles being shown. At the left end of the slider the user is shown posts headlines and as much description as can fit on one line. At the right end of the slider a user is shown the post in its entirety as provided by the source. These states are depicted by two icons at the corresponding sides. There are several ‘stops’ along the slider that allow for

different line lengths to be selected. Now when Eduardo needs to catch up on reading he can simply slide the article length slider all the way to the left to quickly skim his list and read the items that are important to him.

Usability Testing

While conducting the usability tests, we kept in mind that knowledge of how RSS worked was vital, however since it was so easy to explain how such a technology worked, it was rather easy to come by users. We went to the Wells Library and recruited two individuals that resembled our personas, at least in the sense of their technical skills. Both users were male, and both gave us very good insight as to how our design both excelled as a simple design, and what few things needed to be changed or added. We created a document that explained what it was that we were doing, along with both a pre-test and a post-test for the user. We informed the user that all parts of the usability test were optional. One of the users opted out of the final post-test, however, with the information we gathered from the user testing and notes that we took, we determined that it was not needed because the users' body language showed what much of the post-test was intending to show. Here is the data that we collected from the users:

Both testers were males between the ages of 18 and 22. Each were also extremely comfortable using Mozilla Firefox. Only one knew what an RSS feed, and we explained it to the other subject. Neither person has ever used an RSS reader.

When watching the users take the test, we observed and recorded both timing and all actions the user performed. Our first user got off to an unexpected start. He originally decided to try to navigate to the reader by using the Subscription menu item. We had not added the feature in that particular menu, so we asked him to try another way. He then picked the button on the navigation bar that we had hoped he would have used from the beginning. The rest of the test ran fluently, however, we did notice that perhaps he did not understand that the post on the reader was not necessarily from the same source. We determined that to be a prototype error, as we simply overlooked the source information, which is now below the title of the post. We also added a way to navigate to the reader within the Subscription part of the file menu to provide users a choice when they want to go to the reader. On average, it took the first user about 7.4 seconds to figure out how to perform a specific action.

Our second user had never previously heard of RSS. We explained the functions of an RSS feeder, relating it to the ever-popular mini-feed on Facebook. Overall, this user had grasped the idea of our design rather easily. He completed all tasks the way that we had expected, except one. The user did not notice the subscription button at the end of the URL box. Instead, the user went to the file menu under Subscription. On average, it took the second user about 6.2 seconds to figure out how to perform a specific action.

Because going to the file menu 'Subscription' was such a common event within our user testing, we have decided to implement all functions within our feeder into that particular menu. Having all options within a specified area will allow for users to use the reader if they wish to perform a specific task and can't find the proper button in the browser. This way the usability of the reader within the browser will always be easy for any user no matter the level of experience.

Redesign Rationale

After our usability testing, we wanted to change what was most confusing about our design. Since both of our subjects did not click on, or even notice the "More from Source" button we decided this needed to be reworked. We finally decided that since it was flush with the bottom corner of the post, it was not visible to the users. It was also very square which we believe did not alert the user of a button. We wanted to keep with the core design of this button, because it allowed the reader to view the names of multiple previous and next posts. We thought this was the best way to keep context at the core of this feature. We decided to move the button away from the corner and more in sight. Since a user reads a page from top left to bottom right, we decided to keep this button in the same area of the screen, just make it more prominent. We also decided to change the text to "More Posts from Source" to alert the user these were from the same blog. We also wanted to change the button to have round corners so it would look more like a button.

The other changes we made were not as drastic. We realized that we had not included the source of the blog, or the author within our design. Since this is an important component that needed to be part of our design, we added this in under the post link. We thought this could have also contributed to the above confusion with the "More from Source" button.

One of our users also wanted to go to the Subscriptions menu item to subscribe to new sources. We thought that this could be a place where many users would like to go to access this feature, so we added this to that menu. We thought that if they were not looking in the address bar this would be the next viable option. This hopefully provides ease of use to our personas and an average reader. Another very small change that we incorporated is we removed the small "Visit" button after the post link. Both users initially clicked on the post head line link and we felt the button was not needed.

iPhone/iPod Touch Rationale

Navigation Bar

The iPhone generally has up to 5 buttons at the bottom of the display that allows the user to navigate freely through any application that they might be in. Our navigation bar stays static, never changing while in our application. We used four buttons for our bar; having a next and a previous post button, a

settings button, and the RSS button that was implemented into our desktop browser. Our RSS button, however, was not entirely the same as the browser version. Instead, we took the navigational buttons (next and previous post) out and put them on the navigation bar. We kept the 'view all posts' and 'view new posts' options in the feed button. We then put a list of individual subscriptions into the feed button. Along with the list of subscriptions, we added a tiny arrow on the right side of each subscription that added a functionality that allowed for the user to look up information about the subscription, including the URL address. In this regard, the RSS button now acts much like the subscriptions menu item of our desktop browser.

List/expanded views

The list view is by far the most fundamental part of the feeder in itself. The list view allows for the user to be quickly oriented with the context of all of the posts. Much like many other applications on the iPhone, we chose to allow for a flicking gesture to scroll up or down if all of the content was not displayed on the screen at one time. If one were to click on a particular post, an expanded view of this post would appear, having the bottom of the post expand downward. This would display more information, including the entire post and a comments link that could expand to show the comments inline. If a user clicks within the post itself, it would collapse back into the original list view. If a user clicks the post name link, they are taken to the post in the inline browser.

Comments

When comments are left and able to be viewed, a link appears at the bottom of the expanded view of the post. When clicked, the post will then expand to show the comments left by users, much like with our desktop client. When the comments link is clicked, the post will expand downward to display each comment individually. The option to leave your own comment will be linked under the comments left by other users. If this link is clicked, the link will be opened in the inline browser to the blog itself.

Inline browsing features

When in expanded view of a particular post, the title becomes a link to the actual website of the post. When clicked, the website appears in place of the reader. The navigation bar at the bottom remains static through this event, as described in the rationale above. This is crucial to the fundamental idea of our design, allowing the user to control the context. This way the user is easily able to steer back to the reader using the navigation bar. Nothing else besides the website and the navigation bar is displayed so that you can easily navigate to other subscriptions or posts within the reader.

Next/Previous from Source

In the browser version of our design, we had a button in the lower right corner of the posts to allow the user to see other posts that the source may have published. We wanted to keep this feature for the iPhone, but the limited amount of display area posed a problem. We did not want to clutter the area of

the reader and have the user be overwhelmed by the amount of options that the reader displayed. Since we decided to keep many of the finger gesture functionalities that the iPhone offers, we implemented a flicking gesture from side to side, moving the current post off of the screen and replacing it with a past or futuer post from the same source as the original post. This allows for users to stay oriented with each subscription while giving them the physical gestures available on the iPhone.

Conclusion

When this project started, our team worked extremely hard to focus on the core of the problem, which is to give context to the user. After creating personas, multiple sketches, redesigns, and usability testing we feel that our design has addressed this problem head on. By making our reader part of the browser it becomes transparent to the user, giving them access to their subscriptions with one click of a toolbar button. With this reader, a user and their stories are now interlocked. Beyond this, we have also implemented an iPhone/iPod Touch application that gives them this same opportunity no matter where they are in the world. Using UReader, a user gains what they have lost by using other RSS readers, simple context.

Appendix

Figure 1.1

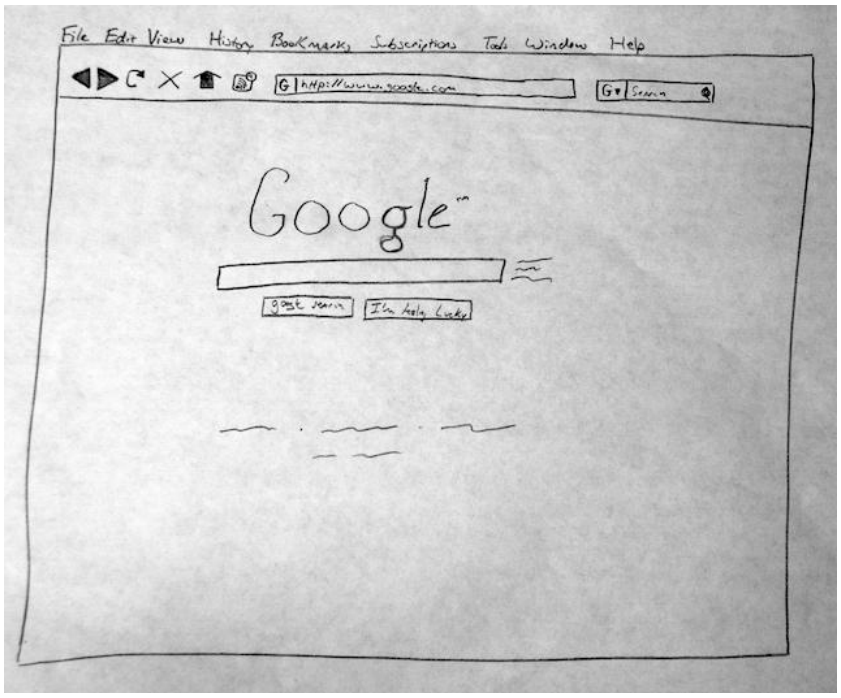

Figure 1.2

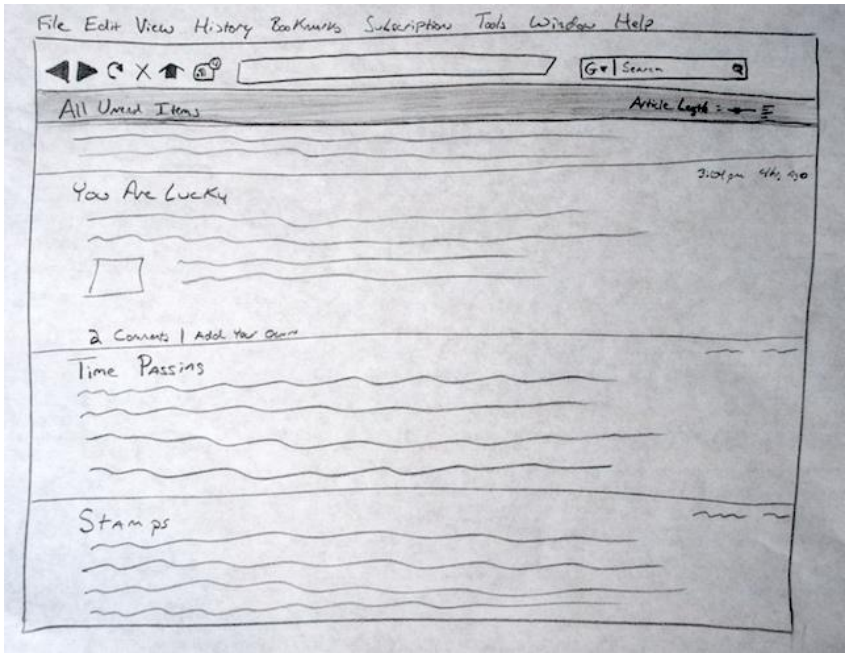

Figure 1.3

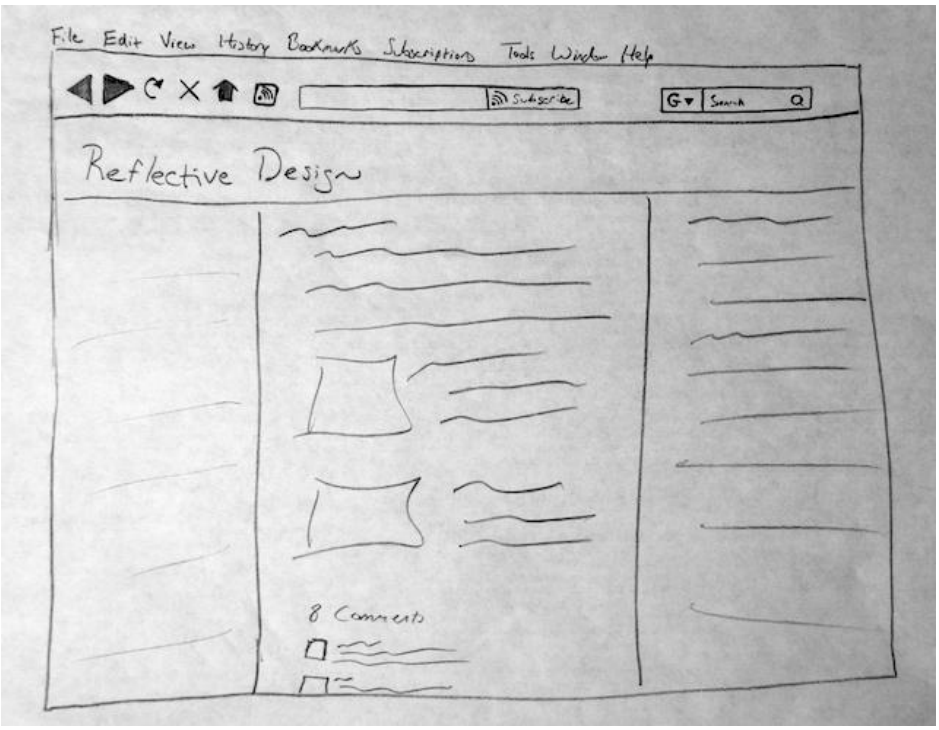

Figure 1.4

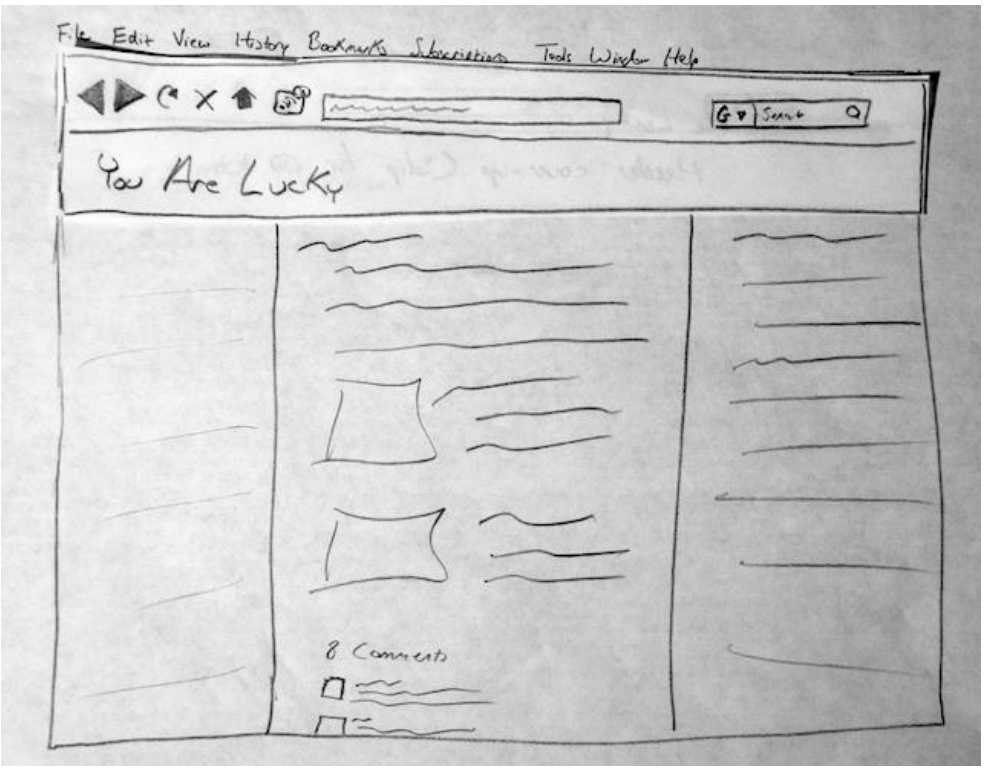

Figure 1.5

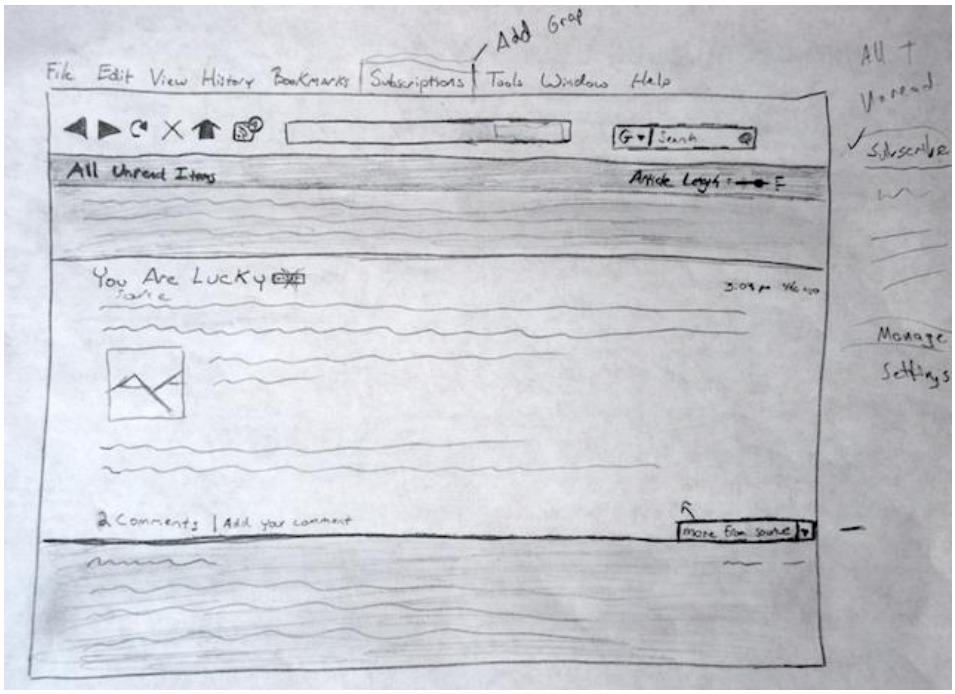

Figure 1.6

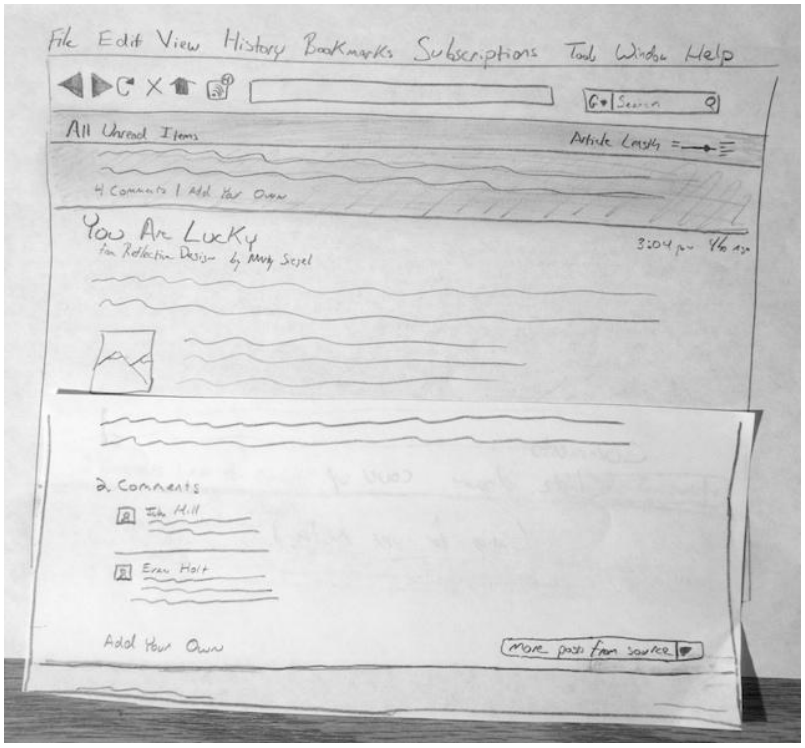

Figure 1.7

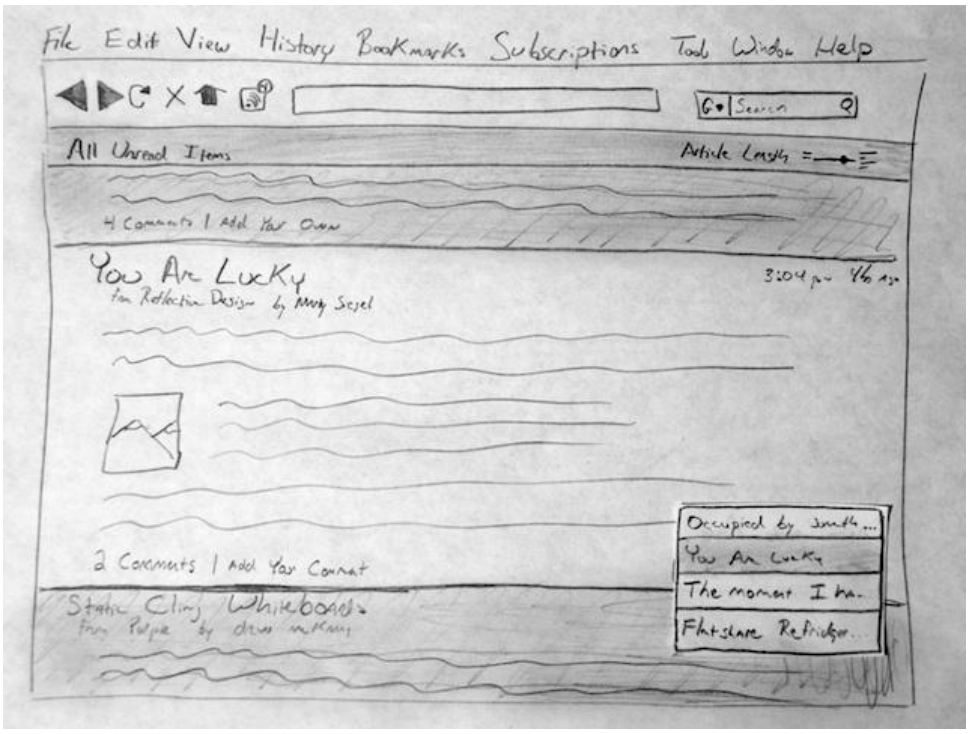

Figure 1.8

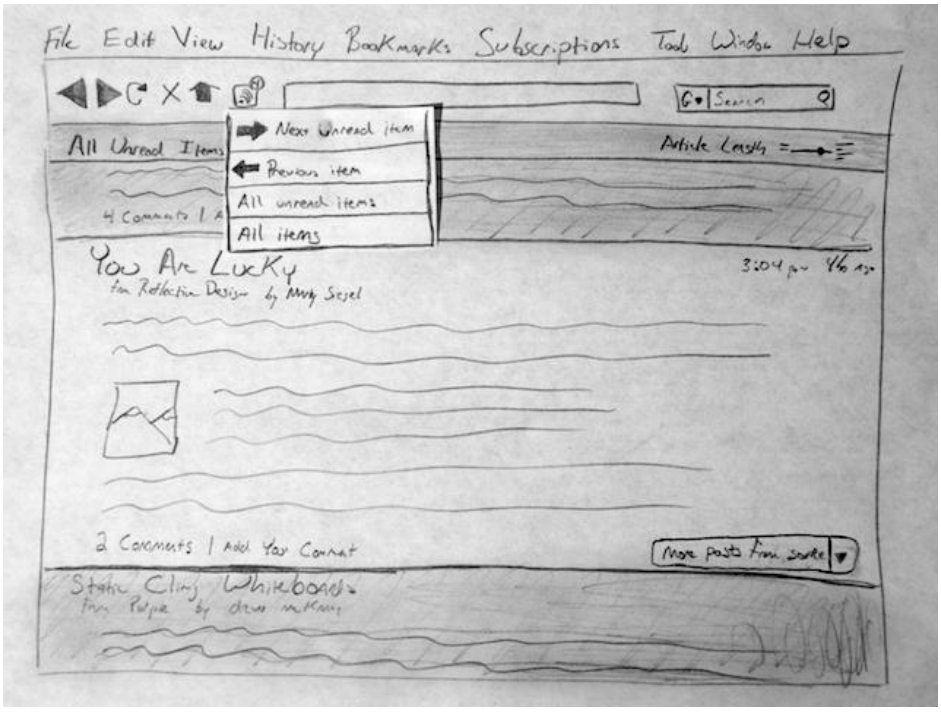

Figure 1.9

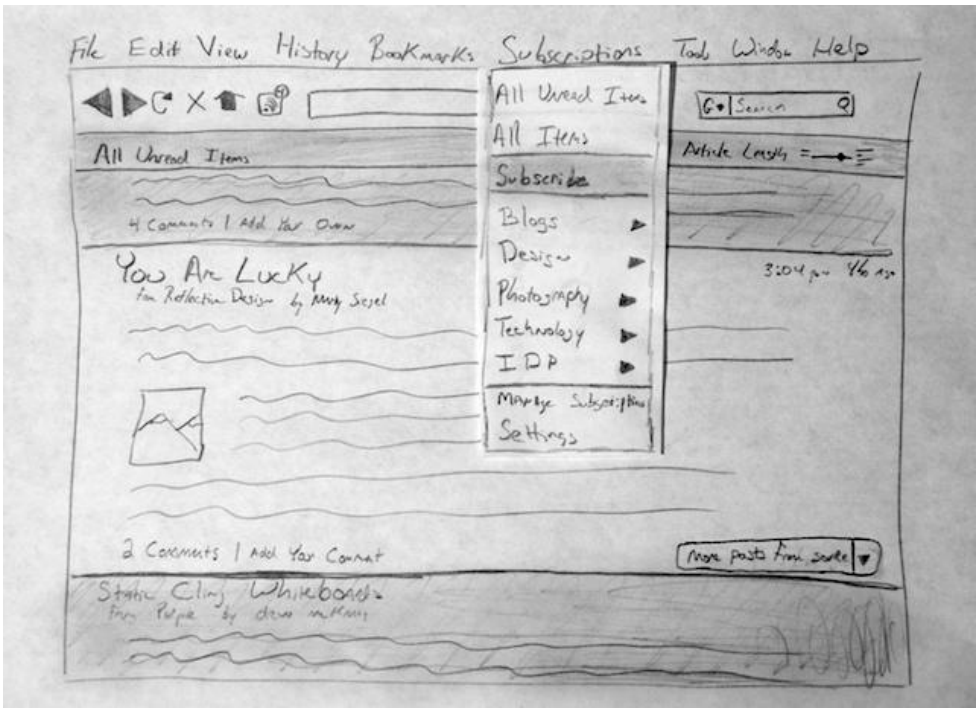

Figure 1.10

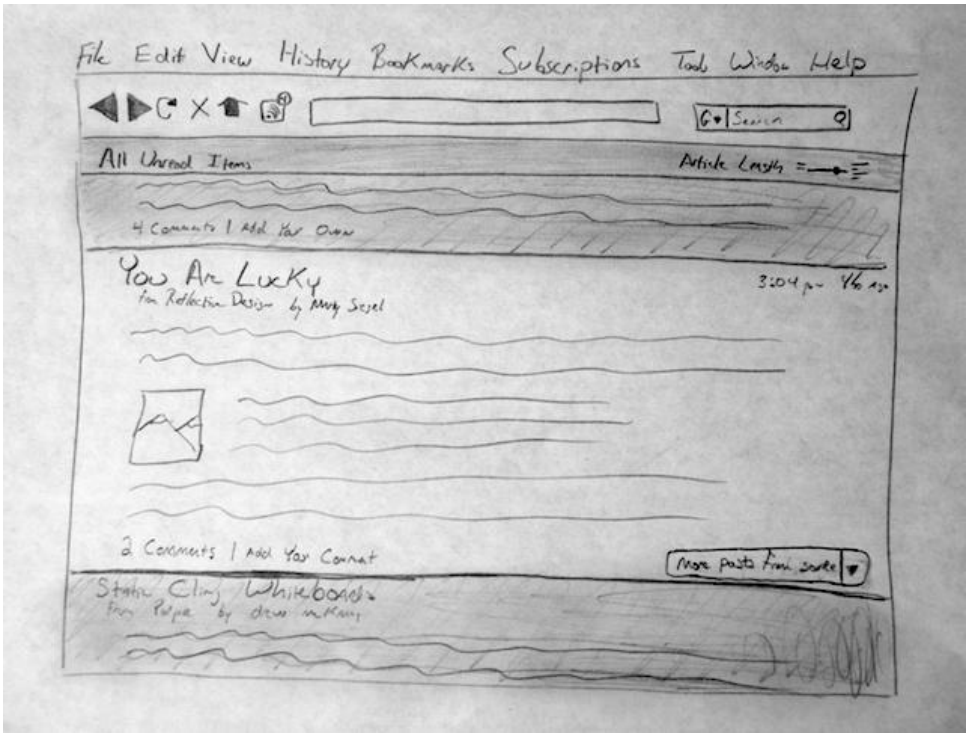

Figure 2.1 – Usability Test

DIRECTIONS

You are about to partake in a usability test. Before the test, you will be asked a series of questions to determine some demographic data to use as research. Following the short pre-test questionnaire you will then be asked to do a series of tasks by the person running the test.

You will be testing a paper prototype of a new Internet browser. The design interface is very similar to the Mozilla Firefox browser that you may be familiar with. If you do not know how to do a certain task, please do not panic and feel free to speak freely about what is stopping you from completing the task. Throughout the entire process we would greatly appreciate if you talk us through what you are doing.

If at any time you have any questions or concerns please let the person administering the test know. At the end of the test you will be asked to take a post-test questionnaire.

Thank you so much for your participation,

Holt Ellis, John Hill, Evan Lipton

Pre-Test Questionnaire

1) What age range do you belong to?

   18-22 22-26 26-30 30-34 34-38 48-42 42+

2) Sex

   Male Female

3) Are you a native English speaker?

   Yes No

4) Are you familiar with Mozilla Firefox?

   Yes No

   a. If yes, how comfortable are you using Firefox? (1 being not at all, 10 being very comfortable)

      1 2 3 4 5 6 7 8 9 10

5) Do you know what an RSS feed is?

   Yes No

6) Have you ever used an RSS reader?

   Yes No

   a. If so, which one?

      _____

   b. What is your level of satisfaction with this reader? (1 being not at all, 10 being very comfortable)

      1 2 3 4 5 6 7 8 9 10

   c. If you are not very satisfied, what could be improved?

Scenarios

1) Find your RSS feed list.
2) Open only your unread posts within your feed list.
3) Open the "You Are Lucky" post.
4) Navigate to the next unread post on your feed list.

Tell them they are now starting a new scenario – put Google page up.

5) Navigate to the web page http://reflectivedesign.wordpress.com that you have never visited before.
    a. Subscribe to this feed
6) Show all items in your feed list.

New Scenario – Put up "You Are Lucky" post

7) You are now reading the "You are Lucky Post" within list mode. You would like to read the previous post from this same blog. What would you do?
8) You are reading the "You are Lucky Post" in list mode. View the comments that have been posted to this post.
    a. Submit your own comment to this post.

Post-Test

1) Was there anything you liked about this reader? What did you like the most? (N/A if did not like)

_____

_____

2) What was one flaw with the reader?

_____

3) What one thing you would change about the reader?

_____

4) Please share any other thoughts about this experience.

_____

_____

Figure 2.2 - Usability Test Results

Usability Test #1

Age - 18-22
Sex - Male
Is a native English speaker.
Is familiar with Mozilla Firefox.. Comfort level 9 (10 being very familiar)
They do know what an RSS feed is.
They have not used an RSS reader before.

Post-Test

5) Was there anything you liked about this reader? What did you like the most? (N/A if did not like)
List mode – view all unread

6) What was one flaw with the reader?
Just make sure the RSS button is obvious.

7) What one thing you would change about the reader?
Make subscribe button different, looks like go button at end of URL bar.

8) Please share any other thoughts about this experience.
NA

Timed Scenario Results (See appendix)

1) 1.4, 5.7
2) 3.0
3) 5.6
4) 5.9
5) 13
6) 6.4 Hesitated, would scroll up
7) 3.2
8) 1.5

Usability Test #2

Age - 18-22
Sex - Male
Is a native English speaker.
Is familiar with Mozilla Firefox.. Comfort level 8 or 9 (10 being very familiar)
They do not know what an RSS feed is, we explained it to him in short detail.
They have not used an RSS reader before.
He choose not to participate in the post test questionnaire.

Timed Scenario Results (See appendix)

1) 1.8, 1.4
2) 1.5
3) 3.4
4) 10.9 – first hit RSS button, then hit subscribe
5) 5.9
6) 12.7
7) 1.6
8) 1.1

Figure 3.I

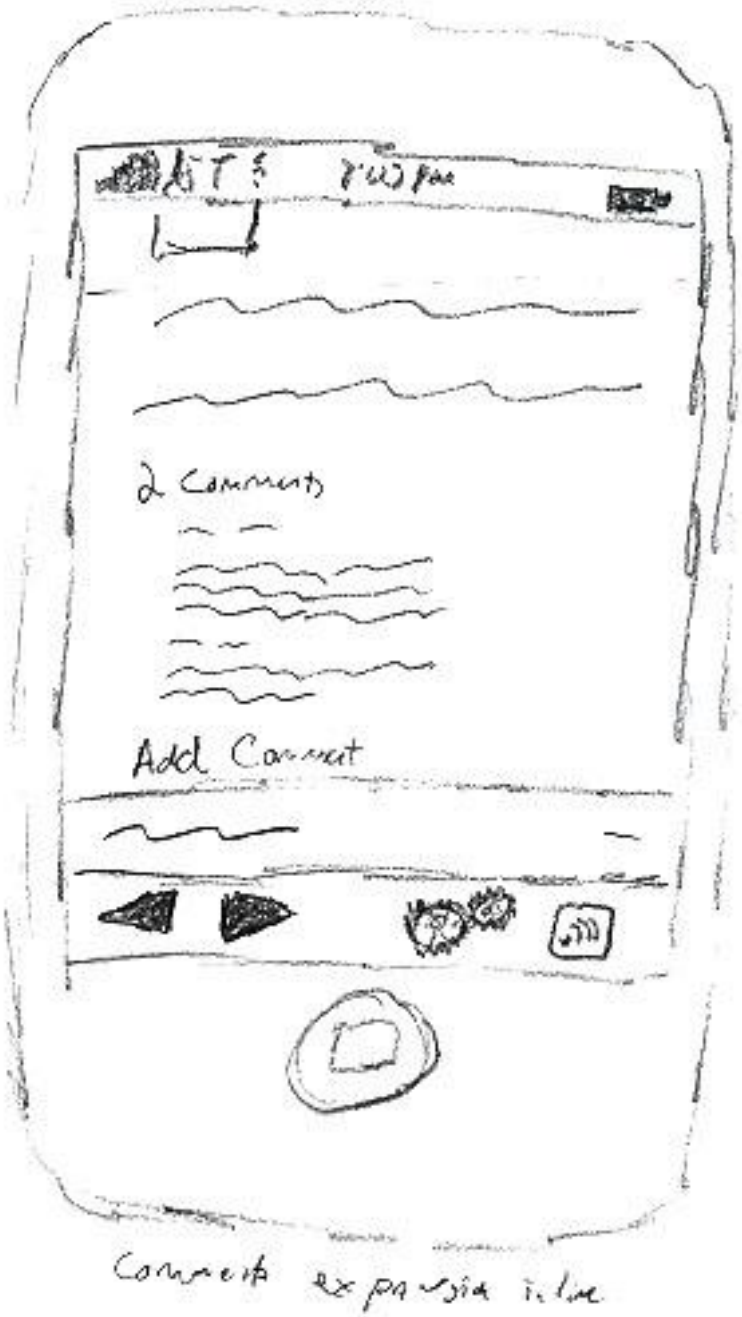

Figure 3.2


select post to collapse
select headline to view in in-line browser
You Are Lucky
Expanded Post View

Figure 3.3

Inline browser view

Figure 3.4

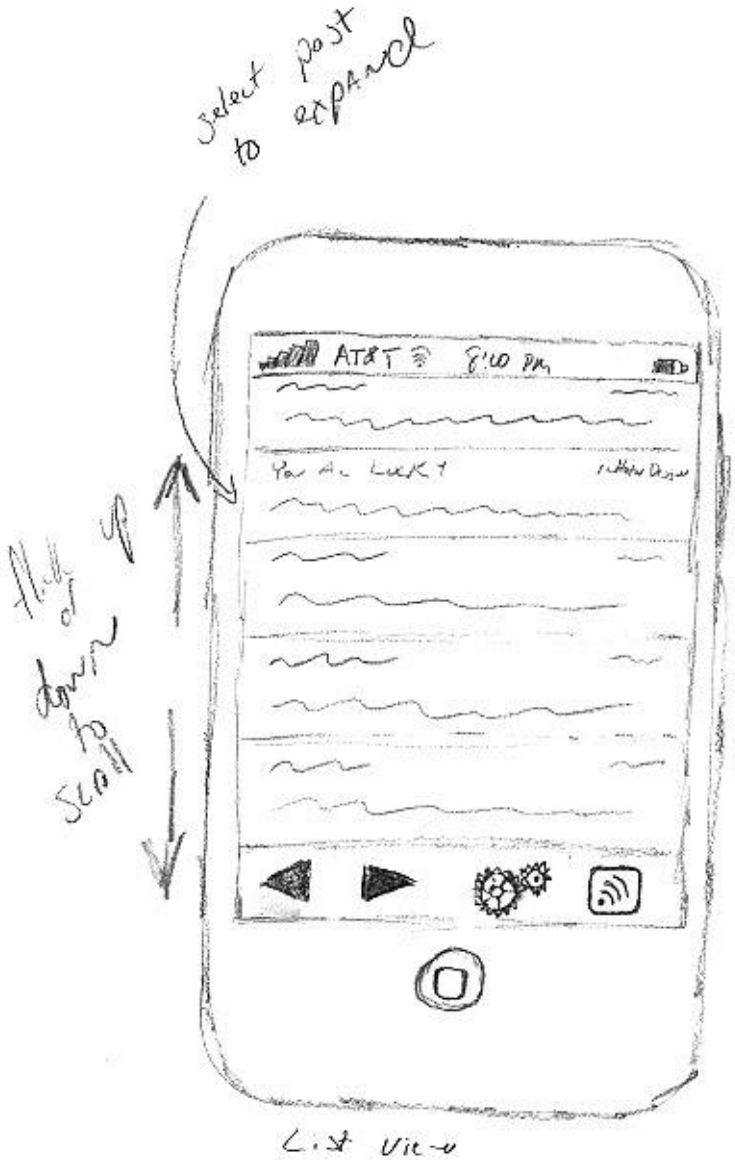
select post to expand
AT&T
8:40 PM
up or down to scroll
List View

Figure 3.5


flick side to side
to see next + previous
posts from source
Lucky
Like C

Figure 3.6

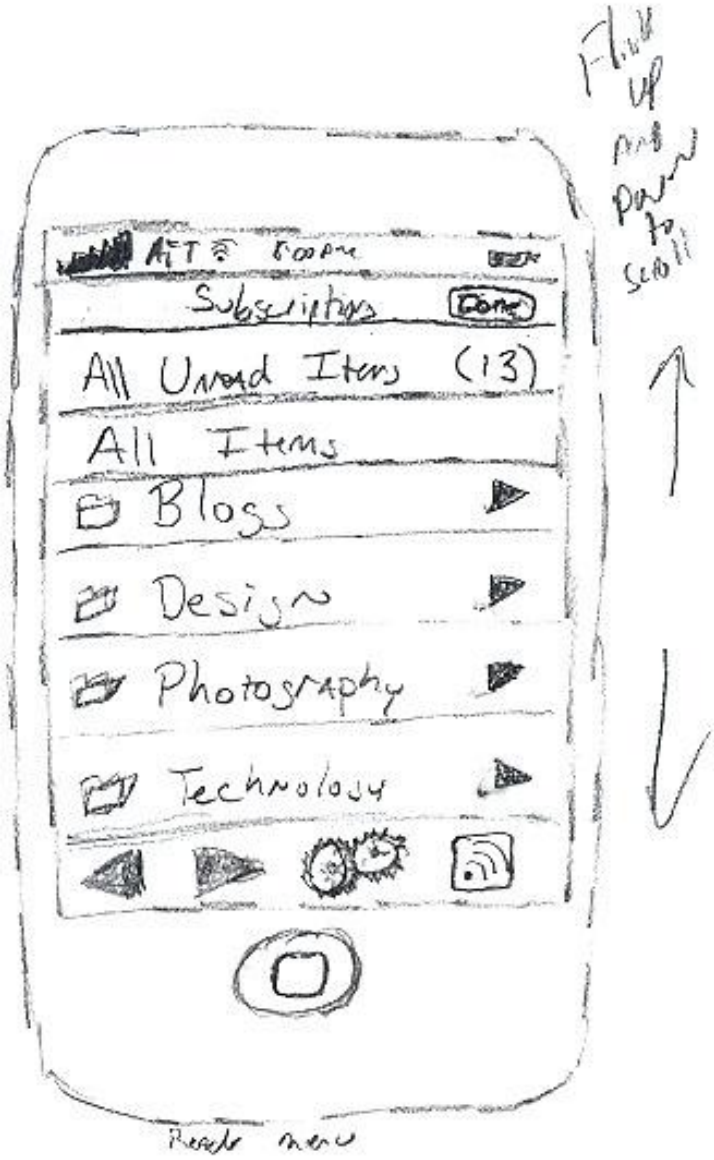
Subscriptions
Done
All Unread Items (13)
All Items
Blogs
Design
Photography
Technology

Figure 3.7