

# Pulse News Mini Heuristic Evaluation

V. Filip Mares

## TABLE OF CONTENTS

---

<b>Table of Contents</b> .....	<b>2</b>
<b>Abstract</b> .....	<b>3</b>
<b>Introduction</b> .....	<b>3</b>
<b>Launching the Application</b> .....	<b>4</b>
<b>Problem 1</b> .....	<b>4</b>
Categorization .....	5
Suggested Solution .....	5
<b>Problem 2</b> .....	<b>5</b>
Categorization .....	6
Suggested Solution .....	6
<b>Browsing Main Screen</b> .....	<b>6</b>
<b>Problem 3</b> .....	<b>6</b>
Categorization .....	7
Suggested Solution .....	7
<b>Problem 4</b> .....	<b>7</b>
Categorization .....	8
Suggested Solution .....	8
<b>Problem 5</b> .....	<b>9</b>
Categorization .....	9
Suggested Solution .....	9
<b>Reading Stories</b> .....	<b>10</b>
<b>Problem 6</b> .....	<b>10</b>
Categorization .....	10
Suggested Solution .....	10
<b>Problem 7</b> .....	<b>11</b>
Categorization .....	11
Suggested Solution .....	11
<b>Managing Sources</b> .....	<b>11</b>
<b>Problem 8</b> .....	<b>12</b>
Categorization .....	12
Suggested Solution .....	13
<b>Summary</b> .....	<b>13</b>
<b>Conclusion</b> .....	<b>13</b>
<b>References</b> .....	<b>14</b>
<b>Appendix a:</b> .....	<b>15</b>

---

## ABSTRACT

---

This paper performs a heuristic evaluation of the iPhone Pulse News Mini application. All tasks within the application are evaluated according to Jakob Nielsen's 1990 heuristics for user interface evaluation. [1] For errors found, solutions are provided. The paper is composed as a requirement for a Software Usability course. Screenshots, material and functionality discussed are in relation to version 1.0.2 of the application. This evaluation is of relevance to iPhone application developers looking to develop based on existing innovative user interface designs of applications available in the iOS AppStore. Furthermore, this evaluation is of interest to usability experts studying the evolution of mobile applications on touch screen smartphone devices.

---

## INTRODUCTION

---

The Pulse Mini News application version 1.0.2 for the iPhone was released on Jul 20, 2010. The application is a companion application to the original Pulse iPad application. Two students at Stanford University who created the iPad version as requirements for a course project developed

the app. The entire development process took 10 weeks to complete. Once released, the application received good coverage in the media for its beautiful design. [2] Pulse is a news application that allows for tracking of news updates from various subscribed news sources and blogs. Previously, RSS feed reader applications for the iPhone followed a more standard approach of listing heading titles as text in a list. Apple's CEO Steve Jobs praised the application for its wonderful design during a keynote for the 2010 World Wide Developer Conference. [3]

In this paper the evaluator will evaluate the application using the 10 heuristics outlined by Jakob Nielsen in 1990. [1] Additionally, the evaluator will use the Apple's iPhone Human Interface Guidelines (HIG). [4] Errors will be identified and then a solution will be provided in order to correct the error. In order to explain certain errors and offer possible solutions, the evaluator researched several online development resources regarding the iPhone software development kit. In order to maintain consistency of naming for views and UI components, the Apple HIG will be used. All tests and screenshots were done on an iPhone 4 with iOS 4.1 installed.

## LAUNCHING THE APPLICATION

---



FIGURE 1 - PULSE APP LAUNCH SCREEN

The first task a user is faced with is launching the application. The evaluator felt this should be added as a major task as heuristic evaluation can also be performed on its user interface.

### PROBLEM 1

---

The evaluator experienced a lengthy loading process for the application. Depending on the length of the waiting time, the user could lose focus and leave the application. The evaluator performed a timed trial of the application Launch process. The test consisted of launching the application from a cold start on an iPhone 4 running iOS version 4.0.2. The completion of the launch was considered to be the point at which the main screen is

viewable to the user.

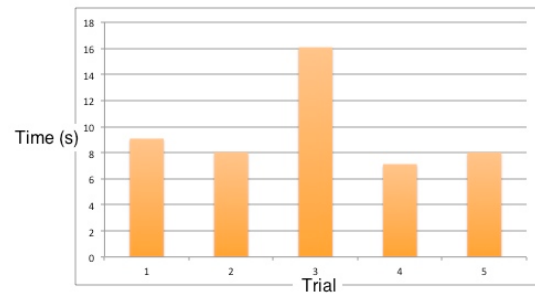


TABLE 1 - TIMED APP LAUNCH RESULTS

Five trials were performed. The results are displayed in Table 1. The third result was rejected due to the large discrepancy in time to load the application. The evaluator believes a daemon in iOS may have caused the application to behave in such a manner. The average time for launching the application was 8.06 seconds. This is below the 10s mark, but above 1s. According to Nielsen's 1990 Heuristics, the user should be presented with simple feedback to the applications status during the loading process. The evaluator only observed the iOS Activity Indicator display in 2 trials. Both times, the indicator appeared in the last second of the loading sequence. There are 2 types of indicators in iOS: status bar for network activity and a larger indicator for application processing. Alternatively, the Alphonso Labs developers could also chose to display the Progress View (Figure 2) if the duration of the task is known. [4 – pp.125] When researching the Apple iPhone Human Interface Guidelines

document, the recommended use of the larger indicator is when the application is taking 1-2 seconds to process the current commands. [4, pp.117] This is violated by the application, which does not display status feedback consistently. The evaluator encounters this problem frequently.



FIGURE 2 - STATUS FEEDBACK INDICATORS

---

### CATEGORIZATION

---

- The problem with the duration of the load time is considered an interaction element problem.
- Likely caused by design
- Only occurs when loading the application.
- Users launch the application by tapping on the icon and the error occurs when executing this action
- Error is caused by poor feedback for the user on lengthy load process.
- Results can be due to the evaluator's 8 RSS feeds in the Main Screen view. Originally, the app installs with 5 feeds. This reduces the number of feeds to refresh when

loading the app.

---

### SUGGESTED SOLUTION

---

The developers need to implement a more consistent feedback mechanism on the loading process in order to inform the user on the activity taking a longer period of time.

Furthermore, when subscribing to additional RSS feeds in the application, the evaluator suggests warning the user that load times might be affected by a large number of RSS feeds. Technically, presenting the user with a shaded screenshot of the last session in the Main Screen view for 1 to 2 seconds can alleviate the problem. This would simulate the act of multitasking and inform the user of the screen that is loading. Following this, the application can continue loading the RSS feeds display a status indicator next to each source's title.

---

### PROBLEM 2

---

In the top right of the loading screen, the developers chose to display text advertising, the iPad version of the application. This text was displayed in red and written in all capital letters. The company name, Alphonso Labs, is also displayed in red on the same screen (Figure 1). This can confuse the user, as one is an advertisement, while the other is company branding. Furthermore, capital letters are harder to read.

---

## CATEGORIZATION

---

- Problem is generated by a design error.
- Developers may have overlooked the loading screen as an important aspect due to quick load times when having few RSS feeds to load.
- Problem is classified as an interaction element problem.
- Presented when the user is executing the launch of the app.
- Error is caused by the failure to highlight the distinction between the company brand and an advertisement.

---

## SUGGESTED SOLUTION

---

The evaluator suggests removing the advertisement from the load screen. Removing it would remove the confusing and the brand would be highlighted. The ad could be included somewhere else in the application, however the evaluator does not recommend it.

Advertisements should be reduced or eliminated in the application. This is due to the fact the app in use was a paid application. This ad would have seemed acceptable in the case where the Pulse News Mini application for the iPhone was released with limited functionality or as a trial version for free. In the case of the trial version, advertising for the full version would have been acceptable.

---

## BROWSING MAIN SCREEN

---

Following the launch sequence the user is presented with the main screen (Figure 3). This displays the subscribed sources in a standard Table View using a simple list. Each row represents a RSS source and stories are represented by an image from the story and its associated title. There are four actionable tasks from the main screen: Browse Sources, Read Story, Edit Source Title, Refresh, Manage Sources and "Love Pulse!".

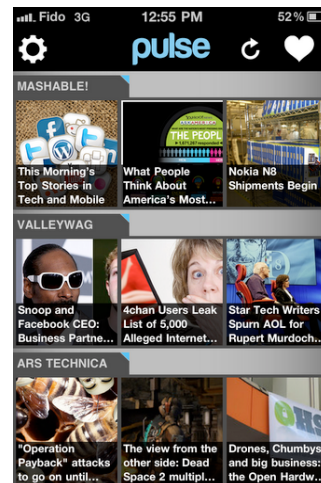


FIGURE 3 - MAIN SCREEN VIEWS

---

## PROBLEM 3

---

When scrolling through sources on the main screen, the user eventually finds themselves at the bottom of the list. A number of native and AppStore iOS applications scroll up to the top of a Table View by simply tapping on the status bar. This then scrolls to the top of the

list. The Pulse application however, does not do this in the Main Screen view. Interestingly, this functionality is supported in the Manage Sources view. This inconsistency may frustrate certain experienced iOS users on a regular basis as they browse through the sources.

---

### CATEGORIZATION

---

- Problem caused by design error.
- The level of the error is classified as an interaction level style problem. It is found both in Main Screen view and Story view.
- This problem occurs when the user is executing the action of scrolling to top by tapping on the status bar.
- The problem is inconsistent by having the functionality in the Manage Sources view and by other apps' standards.

---

### SUGGESTED SOLUTION

---

According to a question thread on Stack Overflow, a popular question and answer website for developers, the error can be caused by using multiple views. [6] It is also possible the main screen view is not designed using the iOS Table View and instead using a custom view.

A possible correction to this error entails

enabling the scrollToTop property on the used view component. This property is available on a Table View for iOS. If utilizing multiple views within the main screen, the evaluator suggests reducing the number of views and rethinking the structure of the design. Furthermore, if the developers have designed the main screen using a custom control they can redesign using one of iOS' existing views that support scrollToTop.

---

### PROBLEM 4

---

When scrolling through the list of sources, the user can tap on the source's title in order to edit it. The evaluator launched this accidentally on a regular basis when scrolling through the sources conducting the evaluation. This feature is mostly needed when shortening a long title of a source. Tapping on the title brings up the keyboard dialog in order to edit the text. This is displayed in Figure 4.

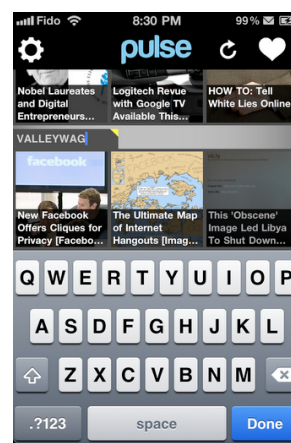


FIGURE 4 - EDITING SOURCE TITLE

---

## CATEGORIZATION

---

- This problem is caused by design.
- It is only found in one place in the app and therefore categorized as an interaction element problem.
- User interprets the shaded background as an exit point. Unfortunately, the Pulse application registers the tap as a selection of the corresponding story. This marks it read with no method to mark it unread.
- A large number of native and AppStore iOS applications can dismiss the keyboard by tapping above it on the background interface.
- This interaction forces the user to recall the result of the action when tapping off the keyboard.
- User otherwise ends up somewhere they didn't want to be.
- User provided only one exit point. Not providing exit for user

---

## SUGGESTED SOLUTION

---

In order to correct the mentioned errors, the developers need to rethink the design of the main screen. This includes reducing the number of buttons on the navigation bar to the standard 2-button design. An edit button can replace the Refresh and "Love Pulse!" buttons. This button would enable and disable an edit mode. As a result, the confusion caused by the current error of

accidentally tapping on the title would be eliminated. The user would simply tap the edit button to enable edit mode. Edit mode would disable side scrolling through the stories and tapping on stories. In order to inform the user of the current status of the application, the edit button could change colour. The user can tap the title of the desired source in order to edit it. The same dialog as in Figure 4 would be displayed with the background interface faded similar to the Google Mobile application for iPhone (Figure 5). This clearly displaying to the user the interface in the background is disabled from accidentally tapping on stories. When done editing, the user has 2 exit paths: tapping the "Done" button or tapping the faded background interface. This would follow the standard that other applications available for iOS have set and follow.

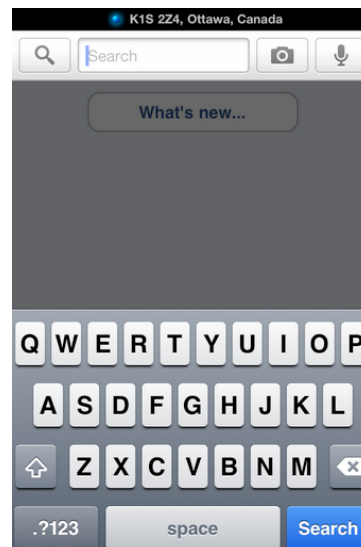


FIGURE 5 - FADED BACKGROUND UI FOR GOOGLE MOBILE APP

## PROBLEM 5

---

The Refresh button is displayed in Figure 3 as the circular arrow on the Navigation Bar. When tapped, the application reloads all sources in a process similar to the initial application launch discussed previously. Apple's Human Interface Guidelines document states that the Navigation Bar usually has 2 buttons used on both sides of the title of the current view. [4, pp.78] This would help reduce clutter in the Navigation Bar.

---

### CATEGORIZATION

---

- This is a design issue for the Main Screen view done by the developers.
- The problem is classified as a conceptual problem.
- This cluttered design can overwhelm the user when specifying the action of refreshing the view.
- The error type is due to the overly complex Main Screen view.

---

### SUGGESTED SOLUTION

---

Sustaining the suggestion previously mentioned of reorganizing the Navigation Bar, the evaluator suggests removing the refresh button and replacing it with another commonly used list refresh interaction. The pull down to refresh gesture was first used by the Tweetie application (Now named

Twitter). This app allows the user to pull down the list of twitter updates and snap it back in order to refresh the twitter feed (Figure 6 and Figure 7). The gesture could also be modified to work horizontally. Each source could be flicked right in order to refresh the feed. The gesture would not replace the universal refresh. In order to account for the original functionality, the evaluator recommends adding the pull down to refresh gesture. This reorganization of the user interface helps reduce the clutter in the Navigation Bar and adds a natural interface for refreshing the sources' feeds. [7]

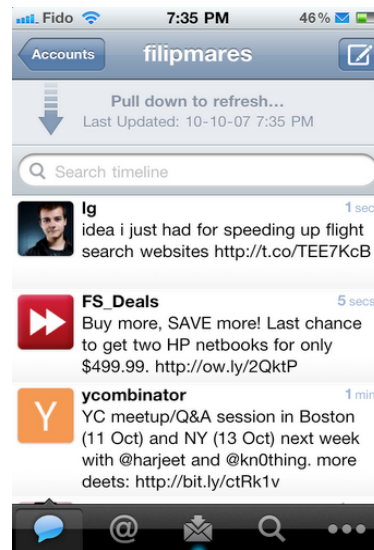


FIGURE 6 - TWITTER APP PULL DOWN TO REFRESH

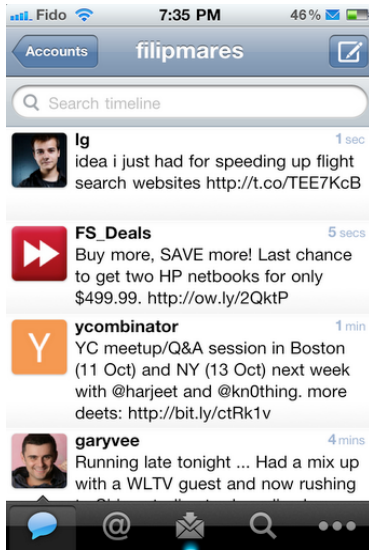


FIGURE 7 - TWITTER APP REFRESH SECTION HIDDEN

## READING STORIES

From the main screen view, a user can select a story by tapping on it. This directs the user to story view (Figure 8). In the story view, the user can read the story and share the story to various social networking and communication sources. The stories displayed at the bottom of the screen are scrollable similar to the main screen view. This is consistent to the interaction in the Main Screen view. Upon scrolling down the story, the stories at the bottom recede which leaves only the title of the source displayed (Figure 9).

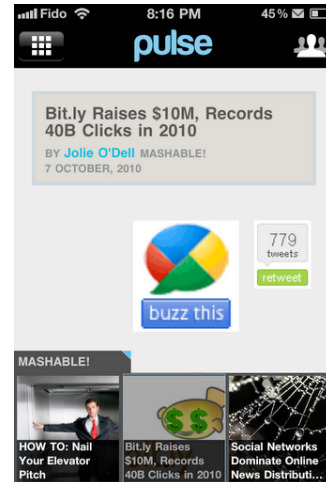


FIGURE 8 - STORY VIEW WITH SCROLLABLE STORIES

## PROBLEM 6

When tapping on the title of the RSS source in Figure 9, the stories are again revealed and can be browsed.

## CATEGORIZATION

- This is a design problem that is only found in one place.
- The user interprets the result to be the same as in Main Screen view and gets it wrong.
- This action is not consistent with the results provided in Main Screen view where the keyboard dialog displayed in order to edit the source's name.

## SUGGESTED SOLUTION

The evaluator suggests fixing the Main Screen view interaction with the source's title. The

current interaction in Story view follows the solution offered by the evaluator for correcting the error in the main screen.

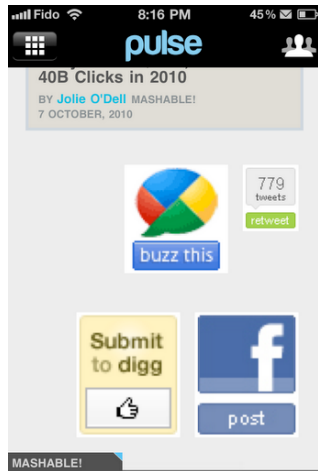


FIGURE 9 - STORY VIEW WITH RECEDED STORIES

### PROBLEM 7

Once a user reaches the bottom of the page on the story they are reading, he/she is not provided with a familiar method of scrolling back up to the start of the story.

### CATEGORIZATION

- This is a design error is the same as discovered in the Main Screen view.
- This is classified as an interaction style problem.
- User doesn't have the expected result when executing the action of tapping on the status bar.

- The problem is inconsistent by having the functionality in the Manage Sources view and by other apps' standards.

### SUGGESTED SOLUTION

The evaluator suggests implementing the scrollToTop function similar to the Main Screen view solution. This would alleviate the problem and follow the solution number of apps are using.

### MANAGING SOURCES

The user can launch the Manage Sources view by tapping on the button with a gear design located on the navigation bar of the main screen (Figure 3). This loads the Manage Sources screen displayed in Figure 12. The user is provided several options in terms of navigation via the Tab Bar at the bottom of the screen. The respective tabs are for Manage Sources, Search, Featured and Google Reader.

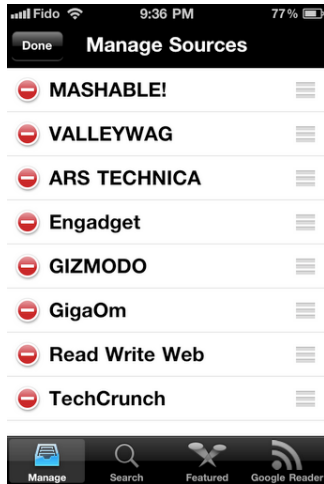


FIGURE 10 - MANAGE SOURCES SCREEN

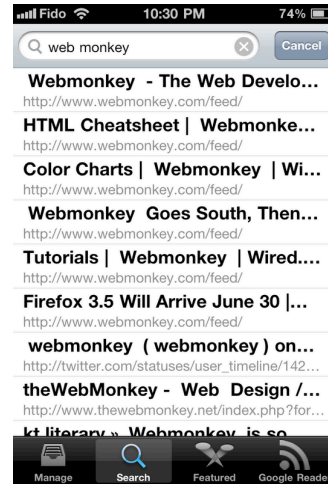


FIGURE 11 - SEARCH SOURCES SCREEN

## PROBLEM 8

When tapping on the Search option in the Tab Bar, the user is presented with the search screen (Figure 13). The user enters search terms at the top of the screen. While searching, the user is provided with a feedback indicator. This indicates the search terms that are being queried. The list is populated with the search results matching the query. The user can scroll through the list when larger than can be displayed on screen. When at the bottom of the screen, the user can tap the status bar in order to scroll to the top. This interaction is also exhibited by the Featured Sources view and Google Reader view (Appendix B).

## CATEGORIZATION

- This is a design error displaying the inconsistencies of the Main Screen and Story view with standard iOS views.
- This is classified as an interaction style problem.
- User is provided functionality of scrollToTop, which is not present in other parts of the application.
- This problem happens when the user taps on the status bar.

---

## SUGGESTED SOLUTION

---

As suggested in previous instances of list scrolling to top interaction level problems, the evaluator recommends implementing the functionality to all views in the application. This would maintain a consistent type of interaction with the application.

---

## SUMMARY

---

Overall, there are a number of usability issues that do not follow Nielsen's heuristics for user interface design. These problems have been outlined in problems 1-8. The user is not provided any Help documentation. This can be beneficial for advanced users that wish to further extend their knowledge of features in their daily usage. It would be great if the developers included a help manual, whether online or embedded, in the application for the various features offered. This could also have been done by a set of video walkthroughs of the features. It is important to note that the evaluator is a regular iOS user and due to the level of experience some problems might have been overlooked. These include design problems of iOS.

---

## CONCLUSION

---

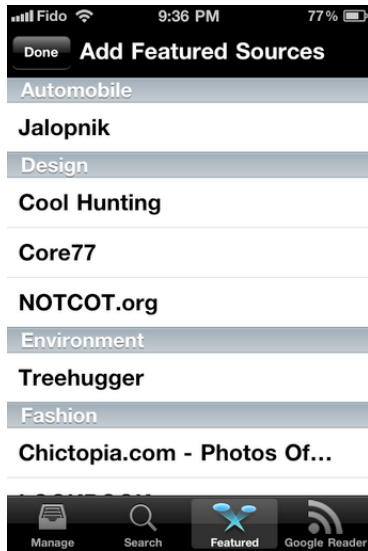
The Pulse News Mini application offers a different interface to other RSS readers available in the Apple AppStore. In doing so, the developers have violated a certain number of user interface design heuristics outlined in this paper. A number of the errors are due mostly to inconsistencies in the interface and users interaction. The developers of Alphonso Labs have attempted to fit as much information as in the iPad version of the application as possible. This is however limited by the reasonably small screen real estate on the iPhone. This at times makes the application feel cluttered aesthetically. One example of this is the main screen navigation bar (Figure 3). The application's different take on RSS feed and story browsing deserves further studies, as the evaluator found the experience more captivating than the more common list style RSS feed reader. Further evaluation of the user interface will be conducted in order to confirm initial error findings in this paper. This will be done via video evaluation with a number of evaluators performing the various tasks outlined in this paper.

## REFERENCES

---

- [1] J. Nielsen. (2005). *Ten Usability Heuristics* [Online]. Available: [http://www.useit.com/papers/heuristic/heuristic\\_list.html](http://www.useit.com/papers/heuristic/heuristic_list.html)
- [2] Alphonso Labs. (2010), *About* [Online]. Available: <http://www.alphonsolabs.com/about>
- [3] Vimeo. (2010). *Steve Jobs highlights Pulse* [Online]. Available: <http://vimeo.com/12782797>
- [4] Apple. (2010, Oct 1). *iPhone Human Interface Guidelines* [Online]. Available: <http://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html>
- [5] Lex Freidman. (2010, Sep. 27). *Pulse News Mini Review* [Online]. Available: [http://www.macworld.com/appguide/app.html?id=581020&lsrc=rss\\_main](http://www.macworld.com/appguide/app.html?id=581020&lsrc=rss_main)
- [6] StackOverflow. (2009, Sep 1). *iPhone OS: Tap status bar to scroll to top doesn't work after remove/add back* [Online]. Available: <http://stackoverflow.com/questions/1361614/iphone-os-tap-status-bar-to-scroll-to-top-doesnt-work-after-remove-add-back>
- [7] Stuart Memo. (2010, Jul. 21). *Pull Down to Refresh* [Online]. Available: <http://www.usabilitypost.com/2010/07/21/pull-down-to-refresh/>

## APPENDIX A:



Add Featured view



Add Google Reader feed view