Need4Feed:
Software Requirement Specification
by BIT

Brian Pohl
Student ID: 9100920124
enzothebaker8@gmail.com

Iker Trun
Student ID: 9101920127
iktrun@gmail.com

Thomas Ingvarsson
Student ID: 9093920122
ingvarsson.thomas@sillys.se

November 26, 2012

Abstract

Nowadays the most common form of communication between people all over the world is taking place on the Internet, or specifically through media such as social networks, forums, and blogs. Blogging has become more and more popular during over the last ten years. The blog is an information sharing tool that allows the user to write about anything that is interesting for him or her and potentially for others with common interests. Through low-maintenance web-based solutions there is also almost no technical knowledge required, compared to more complex web pages, for a user to start and maintain a blog. On the other side taking in information has become just as easy. In today's society everybody has a smartphone with an Internet connection, keeping them online at all time. Users are able to be online on social networks, read forums or follow blogs whenever they like to. The idea of joining the wide assortment of topics a user might find interesting, in a more convenient, more easily consumable form has become attractive for users that would like to have all this information organized in one single location, such as in an application on their smartphones.

Revision History

Version	Date	Description	Author
0.1	09-Oct-2012	First draft.	T. Ingvarsson
0.11	09-Oct-2012	Added Revision History and Table of Contents.	T. Ingvarsson
0.12	09-Oct-2012	Detailed Requirements table has been remade into a multi-page table.	T. Ingvarsson
0.13	14-Oct-2012	Stream-lined the abstract and introduction to not have too much redundancy.	T. Ingvarsson
0.14	15-Oct-2012	Software Interfaces is rewritten as pure requirements that do not imply how it should be solved.	T. Ingvarsson
0.15	15-Oct-2012	Major overhaul of User Interfaces. Simplified and removed unnecessary descriptions. To strict requirements are removed at this stage.	T. Ingvarsson
0.16	15-Oct-2012	Major overhaul of Product Features. Added a use case diagram including a better overview of the features as well as a list of actors to the system.	T. Ingvarsson
0.17	16-Oct-2012	Simplified the System Features section by removing the functional requirements of every subsection, this will be covered in the detailed requirement list anyway. Removed feed list and all with it, is not a feature requirement. Removed subsection Manage Categories as this is covered in Manage Feeds.	T. Ingvarsson
0.18	19-Oct-2012	Added References.	T. Ingvarsson
0.19	23-Oct-2012	Added Software Quality Attributes to Other Non-functional Requirements.	T. Ingvarsson
0.20	23-Oct-2012	Updated use case diagram with more sub-features. Updated System Features to match the use case diagram.	T. Ingvarsson

Version	Date	Description	Author
0.21	24-Oct-2012	Update the Detailed Requirement list to match the rest of the updates.	T. Ingvarsson
0.22	27-Oct-2012	Added new features; Share Post, Import Feed(s), Export Feed(s), Find Similar Feed(s). Changed so it should handle multiple kinds of sources, not just RSS.	T. Ingvarsson
0.30	21-Nov-2012	Added comparison table of the most popular readers existing on the Google Play Store to clarify the differences.	T. Ingvarsson
0.31	26-Nov-2012	Added a system feature; gather and display statistics. Also updated the deadlines mentioned under project constraints.	T. Ingvarsson

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to specify a software for Android users enabling the functionality to gather, organize and read feeds, regardless of if it is a RSS[1], ATOM[2], Podcasts or social medium feed. Furthermore, this requirement specification will act as base for the specification which in turn will be the base of the design document. This will be the first release of the mentioned software which is completely standalone in the sense that it is not part of a larger system.

1.2 Intended Audience and Reading Suggestions

The intended audience for this document are those with an engineering background interested in this project, whether it be the actual teacher assigning this project to the project group, one of the group members or an actual user wanting to know more about the product or continue development. Due to the small size of the actual product I would recommend all readers to read the whole document from start to finish. However to understand the major features, section 2 will do. If one is only interested in the actual requirements they are listed in section 6, Detailed Requirements. A section in which all requirements are listed with their ID, short description, priority and source.

1.3 Project Scope

Nowadays the most common form of communication between people all over the world is taking place on the Internet, or specifically through media such as social networks, forums, and blogs. Blogging has become more and more popular during over the last ten years. The blog is an information sharing tool that allows the user to write about anything that is interesting for him or her and potentially for others with common interests. Through low-maintenance web-based solutions there is also almost no technical knowledge required, compared to more complex web pages, for a user to start and maintain a blog. On the other side taking in information has become just as easy. In today's society everybody has a smartphone with an Internet connection, keeping them online at all time. Users are able to be online on social networks, read forums or follow blogs whenever they like to. The idea of joining the wide assortment of topics a user might find interesting, in a more convenient, more easily consumable form has become attractive for users that would like to have all this information organized in one single location, such as in an application on their smartphones.

The development team for this software project has set out to minimize the clutter that comes along with following many different sources, while simultaneously being able to organize the material in great depth. The goal of the program is to minimize time spent ineffectively trying to manage multiple sources and maximize the content available for the user.

Feature	Feedly	FeedR	Pulse News	gReader	Need4Feed
Supports *RSS*, *ATOM*, and *Podcast*	✓	✓	✓	✓	✓
Supports *Facebook* and *Twitter*			✓		✓
Supports *Youtube* and *Tumblr*	✓		✓		
Supports *Digg*, *Flickr*, *PicPlz*, *Reddit*, and *Vimeo*			✓		
Categorize Feeds	✓	✓	✓	✓	✓
Store Locally	✓	✓	✓	✓	✓
Share to Social Networks	✓	✓	✓	✓	✓
Integrate with Google Reader	✓	✓	✓	✓	
Require Account	✓	✓	✓	✓	
Suggest Feeds	✓		✓		✓
Multi-Language	✓		✓		
Personalize with Themes	✓	✓	✓	✓	
Open-source					✓

Table 1: Supported Features of The Most Popular RSS Readers

Today there exist several similar solutions; web-based such as *Google Reader*[3], desktop clients such as *Reeder*[4] and smartphone applications such as *Feedly*[5]. What this software will offer to its users compared to the already existing solutions is a minimalistic interface with low overhead but still capable of giving the full-fledged feed experience, without relying on a web-based service such as *Google Reader*. An experience including adding your favorite feeds, categorizing them and keeping track of the latest posts. By including the possibility to follow social mediums, such as *Facebook* or *Twitter*, along side your regular feeds the product distinguishes from its competition.

Seen in table 1 are the four most popular *RSS* Readers listed along side with this product. Compared are the main features of each product to visualize where this product succeeds compared to its competition. First thing that can be noticed in the table is that all readers have the same standard support of sources while *Pulse News* plays in its own division with a massive list of supported sources. Here does the product land in the middle, supporting the most popular sources without bloating with less used or popular sources the way one could discuss that *Pulse News* does. Beyond that one can also observe that the product requires no integration with *Google Reader* as mentioned earlier,

actually no account all is required to use the product removing any ties otherwise created using one of the more popular choices. The last but not least difference, one could even consider it being the most important one, is that the product is open-source, something the more popular choices are not.

The user of the suggested solution is anyone with a smartphone wanting to organize and follow their favorite feeds through a software that is convenient and approachable. The user should be able to go to the software whenever they have downtime or the need of the latest information.

1.4   Project Constraints

The most pressing constraint of the project is that it must be completed within a strict time frame where there are deadlines for multiple milestones. These milestones are:

9th October Requirement Analysis: Software Requirements Specification

23rd October Specification: Structured system/Object-oriented analysis

22th November Design/Implementation: Architecture design and Source code/Testing

7th December Demo: Presentation/Installation guide

Another constraint is the limited development team, consisting only of the project group of three people.

1.5   Definitions, acronyms and abbreviations

Below is a list of definitions, acronyms and abbreviations referenced in this document, sorted after appearance.

Android
　　　A mobile operating system developed by Google

RSS
　　　Rich **S**ite **S**ummary

ATOM
　　　The Atom Syndication Format is an XML language used for web feeds

Podcast
　　　A multimedia digital file made available on the Internet for downloading

Google Reader
　　　Google Reader is a Web-based aggregator, capable of reading Atom and RSS feeds online or offline

Reeder
　　　A Google Reader client for Mac

Feedly
   A Google Reader client for Chrome, iOS, Android and Kindle

Facebook
   A social networking website launched in February 2004

Twitter
   An online social networking service and microblogging service

TBD
   To **B**e **D**eclared

Froyo
   Version 2.2 of the Android operating system

Ice Cream Sandwich
   Version 4.1 of the Android operating system

1.6   References

In this section is a list of all documents referenced throughout the rest of this document.

[1] RSS Advisory Board. Rss 2.0 specification (version 2.0.11). `http://www.rssboard.org/rss-specification`, October 2012.

[2] R. Sayre M. Nottingham. The atom syndication format. `http://www.ietf.org/rfc/rfc4287.txt`, October 2012.

[3] Google Inc. Google reader. `http://reader.google.com`, October 2012.

[4] Reeder. Reeder. `http://http://reederapp.com`, October 2012.

[5] feedly. feedly. a news reader for creative minds. `http://www.feedly.com`, October 2012.

[6] Google Inc. Android. `http://www.android.com`, October 2012.

[7] Google Inc. Permissions — android developers. `http://developer.android.com/guide/topics/security/permissions.html`, October 2012.

2   Overall Description

2.1   Product Perspective

The product to be is a new open-source feed reader, with no predecessors, that will compete with existing solutions on the market by offering an low-overhead and minimalistic interface without the need of a web-based back-end. The goal is to give users a simple but powerful feed reading solution.
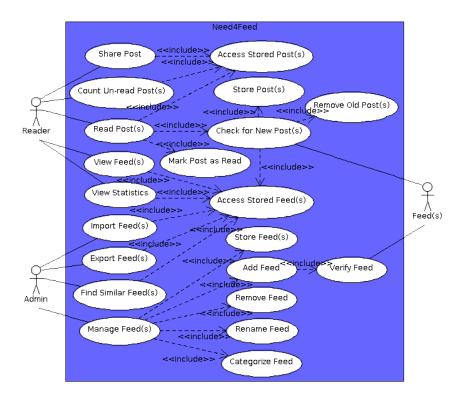
2.2   Product Features

Figure 1: Complete use case of the major features

The product's use case diagram is shown in figure 1, where actors, major features and included sub-features are displayed. The overall feature can be described as a feed library where the user can organize feeds as well as view posts. The actors of the product are Reader and Admin, which are subsets of User, as well as Feed(s). Feed(s) are not a person but still an actor to the product as it provides data.

The major features of the product are:

- Share Post

- Count Un-read Post(s)

- Read Post(s)

- View Feed(s)

- Import Feed(s)

- Export Feed(s)

- Find Similar Feed(s)

- Manage Feed(s)

Underneath the major features are a set of sub-features to describe the flow of the product.

The major user interfaces are:

- View Categories

- View Feeds

- View Posts

- View Post

The product shall include these four user interfaces to enable the user to utilize all the major features of the product.

2.3   Operating Environment

The environment for the product shall be Android[6], which is an operating system for touchscreen mobile devices. The product shall be supported for Android version 2.2, *Froyo*, up to 4.1, *Ice Cream Sandwich*. The product shall comply the hardware requirements that are required by the specified Android versions.

2.4   User Documentation

To complement the product a user manual shall be created. This manual shall contain common user scenarios as well as descriptions of all options available throughout the user interface. An optional addition would be a developers guide to encourage outsiders to get involved.

3 System Features

In this section are the major features mentioned in section 2.2 described, sub-features are handled within each parent feature.

3.1 Share Post

The product shall be able to share a post and its content over a social medium. This feature is an optional feature for the product and has low priority.

3.1.1 Stimulus/Response Sequences

User triggered:

- Share Post - Compile a post containing the post's address and a user comment and send to a social medium.

System triggered:

- Access Stored Post(s) - Access the address of a locally stored post.

3.2 Count Un-read Post(s)

The product shall be able to present the amount of un-read posts stored locally. This feature is an optional feature for the product and has medium priority.

3.2.1 Stimulus/Response Sequences

User triggered:

- Count Un-read Post(s) - The amount of un-read post(s) should be presented to the user in the appropriate view.

System triggered:

- Access Stored Post(s) - Access and check the read status of all locally stored post(s).

3.3 Read Post(s)

The product shall give the user the ability to Read either several posts, by their headers, or a single post with all of its content. To make sure that the user has the latest posts available the product shall check for new posts, and if any store them locally. To preserve memory shall the product only store a certain amount of posts per feed, once this limit has been reached the product shall remove the oldest posts. The posts presented to the user shall be stored locally and when read they should be marked as read. This feature, along with its sub-features, are key features for the product and have high priority.

3.3.1 Stimulus/Response Sequences

User triggered:

- Read posts - List several posts in an appropriate view where the header of several posts can be read at the same time.

- Read post - Display a post, both the header and the content , in an appropriate view.

System triggered:

- Check for New Post(s) - Fetch all the latest posts on start-up or when a new feed has been added.

- Access Stored Feed(s) - To determine where the posts should be fetched from the feed(s) has to be loaded from local storage.

- Store Post(s) - Store all posts along with its attributes: date, parent feed and read status.

- Remove Old Post(s) - Remove old posts either on startup to lower memory usage, combined with removal of feed or removal of read posts.

- Access Stored Post(s) - Load a stored post to avoid unnecessary fetch.

- Mark Post as Read - Mark a post as read when it has been viewed by an user.

3.4 View Feed(s)

The product shall give the user the ability to View feeds. This feature is a key feature for the product and has high priority.

3.4.1 Stimulus/Response Sequences

User triggered:

- View Feed(s) - List all added feeds in an appropriate view where multiple feeds are visible at the same time.

System triggered:

- Access Stored Feed(s) - Access locally stored feeds to be able to list them.

3.5 View Statistics

The product should give the user the ability to View statistics gathered during the usage of the product. The statistics shall include how much time the user spends reading each post, viewing a feed, or category. It should also record the amount of re-reads of posts, amount of un-read posts marked as read, and read posts per day, week, month or forever. The statistics should be presented in such a way that the relevant information is shown. This feature is an optional feature for the product and has low priority.

3.5.1 Stimulus/Response Sequences

User triggered:

- View Statistics - Lists all statistics recorded by the product since install.

System triggered:

- Access Stored Feed(s) - Access locally stored feeds to accesses stored statistics gathered throughout the use.

3.6 Import/Export Feed(s)

The product shall be able to import or export all feeds in a standardized way to, or from, other feed services. This feature is an optional feature for the product and has low priority.

3.6.1 Stimulus/Response Sequences

User triggered:

- Import/Export Feed(s) - Combine all existing feed(s) into a standardized file format to be imported elsewhere. Import is a way to mass-add feed(s).

System triggered:

- Access Stored Feed(s) - Access locally stored feed(s).

3.7 Find Similar Feed(s)

The product shall be able to look for similar feed(s) and present them to the user. This feature is an optional feature for the product and has low priority.

3.7.1 Stimulus/Response Sequences

User triggered:

- Find Similar Feed(s) - Look for similar feeds by the use of online databases.

System triggered:

- Access Stored Feed(s) - Access locally stored feed(s) to use as search term.

3.8 Manage Feed(s)

The product shall give the user the ability to Add, Remove, Rename and Categorize feeds. Adding a feed shall require the product to first verify that the feed exists. The product shall store all feeds locally so that they can be accessed at a later time again. This feature, along with its sub-features, are key features for the product and have high priority.

3.8.1 Stimulus/Response Sequences

User triggered:

- Add, Remove, Rename, Categorize - Options that should be accessible through the user interface in the appropriate view.

System triggered:

- Categorize - All feeds should as default be categorized, either when added without category or at removal of category, to category *Uncategorized*. This will guarantee that all feeds have a category assigned.

- Verify Feed - The product shall verify that the RSS feed exists to complete an add feed operation.

- Store Feed(s) - All feeds should be stored so that they can be accessed at a later time again.

4   System Interface

4.1   User Interfaces

The product shall provide a set of views enabling the user to all system features.

4.1.1   Main View

The product shall be able to list all categories in alphabetic order, clicking on a category shall take the user to the *Feeds View* for that specific category. The view shall include the following options:

- Add category

- Remove category

- Add feed

- Fetch posts

The first entry in the list should be a *View all...* button that should show the latest posts from all feeds in all categories. The last entry in the list should be a *Add Category...* button.

4.1.2   Category View

The product shall be able to list all feeds for a specific category in alphabetic order, clicking on a feed shall take the user to the *Posts View* for that specific feed. The view shall include the following options:

- Add feed

- Remove feed

- Remove active category

- Fetch posts

The first entry in the list should be a *View All...* button that should show the latest posts from all feeds in the specific category. The last entry in the list should be a *Add Feed...* button.

4.1.3   Feed View

The product shall be able to list the latest posts for a specific feed in a latest first order. Clicking on a post shall take the user to the *Post View* for that specific post. The posts shall be displayed with their heading. The view shall include the following options:

- Mark all posts as read/unread

- Remove active feed

- Fetch posts

The last entry in the list should be a *Older Posts...* button that should fetch older posts to show.

4.1.4 Post View

The product shall be able to display a specific post and all of its content. The view should enable sliding; sliding the screen to the right should show a older post, while sliding to the left should show a newer post.

4.2 Software Interfaces

4.2.1 Internet

The product shall require access to Internet so that RSS feeds can be accessed and posts be downloaded. RSS feeds shall be accessed through either HTTP or FTP.

4.2.2 External Memory

The product should be able to store settings, feed lists and posts on an external memory. It should be able to both store and remove files.

4.2.3 Background Task

The product should periodically update; download new posts and delete old ones. If the product's main interface is active it shall make sure that an opened post is not deleted.

4.2.4 Notification Bar

The product should be able to notify the user that there are new posts through the notification bar. It should also show the amount of un-read posts.

5 Other Non-functional Requirements

5.1 Security Requirements

For the product to work properly it will require permission from the user for certain features according to Android[7].

Android user permissions required:

- Full Internet Access - To access and download RSS feeds.

Android user permissions that depends on optional features:

- Modify/Delete SD Card Content - Enables storage of settings, feed list and posts on SD card.

- Modify Global System Settings - Enables the use of notifications.

5.2 Software Quality Attributes

In this subsection are software quality requirements that are non-functional listed.

Availability
The product shall be available at all time, even without Internet access.

Flexibility
Functionality should be possible to add without affecting the current functionality.

Reliability
The product should not lose performance due to too much content in either the amount of feeds, posts or the content within one post.

Scalability
The product shall be able to store and handle feeds and posts in such a way that only the actual storage size in memory restricts. The product should be able to handle throughput from unlimited amounts of feeds without preventing the user accessing already downloaded posts.

Usability
The user shall without reading a manual be able to add, remove, categorize feeds. Beyond that the user shall also be able to, again without reading a manual, view all feeds, view feeds of a certain category, view posts of a certain feed and view individual posts.

6 Detailed Requirements

In table 2 all the requirements are listed with an ID, short description and priority. The different ID categories are:

P Post

F Feed

U User Interface

I Interface

S Security

Q Quality

Table 2: List of Detailed Requirements

ID	Description	Priority
P-1	Check for new Post(s)	High Priority
P-2	Store Post(s)	High Priority
P-3	Remove Old Post(s)	High Priority
P-4	Access Stored Post(s)	High Priority
P-5	Mark Post as Read	Medium Priority
P-6	Count Un-read Post(s)	Medium Priority
P-7	Share Post	Low Priority
F-1	Add Feed	High Priority
F-2	Remove Feed	High Priority
F-3	Rename Feed	High Priority
F-4	Categorize Feed	High Priority
F-5	Verify Feed	High Priority
F-6	Store Feed(s)	High Priority
F-7	Access Stored Feed(s)	High Priority
F-8	Export Feed(s)	Low Priority
F-9	Import Feed(s)	Low Priority
F-10	Find Similar Feed(s)	Low Priority
U-1	Present the Amount of Un-read Post(s)	Medium Priority
U-2	Present a List of Post(s)	High Priority
U-3	Present a Single Post with Header and Content	High Priority
U-4	Present a List of Feed(s)	High Priority
U-5	Present a List of Categories	Medium Priority
	Continued on next page	

Table 2 – continued from previous page

ID	Description	Priority
I-1	Internet Access to Fetch Post(s)	High Priority
I-2	External Memory Access to Store/Access Feeds and Posts	High Priority
I-3	Periodical Background Tasks to Fetch Latest Posts.	Medium Priority
I-4	Notification Bar Access to Notify on New Posts and Amount of Un-read Posts	Low Priority
S-1	Android User Permission: Full Internet Access	High Priority
S-2	Android User Permission: Modify/Delete SD Card Content	High Priority
S-3	Android User Permission: Modify Global System Settings	Low Priority
Q-1	The product shall be available without Internet Access	High Priority
Q-2	The product should be possible to be extended with more functionality without breaking the existing.	Medium Priority
Q-3	The amount of feeds, posts or post content to display shall not affect the performance to drop	High Priority
Q-4	The amount of feeds/posts to manage shall only be restricted by storage space	Medium Priority
Q-5	Accessing feeds to fetch posts shall not affect the user from accessing locally stored posts	High Priority
Q-6	The user shall be able to manage feeds without using a manual.	High Priority
Q-7	The user shall be able to navigate through categories, feeds and posts without using a manual.	High Priority