News Feeder

Requirements Specification

Group LM01
Michael Boge, Aron Hardy-Bardsley, Ian Mckay, Shiwei Zhang

Table of Contents

Introduction 2
Purpose 2
Scope 2
Glossary 3
System Overview 4
System Structure 5
Users 6
Constraints 6
Requirements 7
Priorities 7
Requirements Summary 7
Functional Requirements 9
Front-end 9
Back-end 13
Non-Functional Requirements 15

Introduction

Purpose

The purpose of this document is to inform the project team and project stakeholders about the News Feeder service and all of its subsystems, as well as outlining its core and extension requirements.

Scope

This document describes the News Feeder service (referred to as News Feeder throughout this document). It provides a high level description of the service as a whole, including its structure, sub-systems and potential users. This document also outlines the requirements/objectives to provide the features and functionality of the News Feeder service and assigns priorities to each of these.

This document's scope is limited to describing objectives at a high-level and therefore will not go into specific detail about each requirement's implementation in the system.

Glossary

Item/Article – An individual piece of content. This includes articles from RSS feeds or items from other types of content sources (such as an email, forum thread, weather forecast, etc).

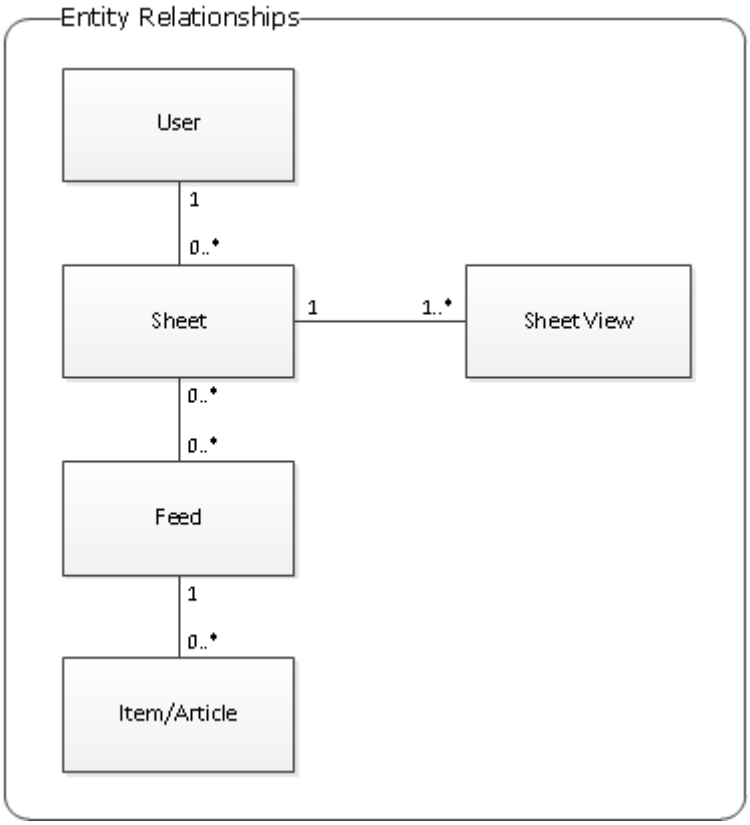
Feed – A collection of items/articles all related to the same content source (i.e. the same site or RSS feed).

Sheet – A grouping method for feeds. It displays feeds on the website as views the user has created.

Sheet View – The layout used to display feeds and content associated with the sheet. An individual sheet may have many of these views. When referring to a sheet and the way it displays its content, we are implicitly referring to its default view.

Front-end – The web interface of the system, where users gain access to and interact with the system.

Back-end – An application which runs a continuous queue and crawls/parses content sources for articles to add to the database.



System Overview

News Feeder is a news feed aggregator service, which compiles content from multiple online content sources. Its main purpose is to provide a single location and format to view news content instead of users having to access multiple websites for their specific news.

News Feeder allows users to specify which type of articles they wish to view, based on a number of criteria, including; location, category, author, publishing site and specific feed URLs.

The system crawls feeds which match the specified criteria and stores the relevant articles/items and their content. The articles/items can come from a number of different content sources (such as RSS, forums and social networks) and their content can be of any media type, including; text, images, video and geo-locational information.

The user can then view the stored articles and the article content on their sheets. The layout of the sheet will be in a standard news sheet format by default, but the system also provides a simple drag and drop interface for creating custom user layouts for sheets. Sheets will allow users to set filtering options to ensure certain types of content are not displayed to inappropriate audiences (such as pornographic material to minors).

Users can choose to subscribe to the News Feeder service for a monthly fee. This provides them with a number of benefits over standard users, such as access to the downloader application, customisable sheet layouts, increased feed/sheet limits and increased priority for content retrieval.

The system provides access for administrators to manage the News Feeder service. Administrators can manage users, feeds and items, as well as managing and monitoring back-end system processes (such as starting and stopping the crawler).

Access to this functionality is available via a web-interface. Users may also access News Feeder content offline via the use of a downloader application.

System Structure

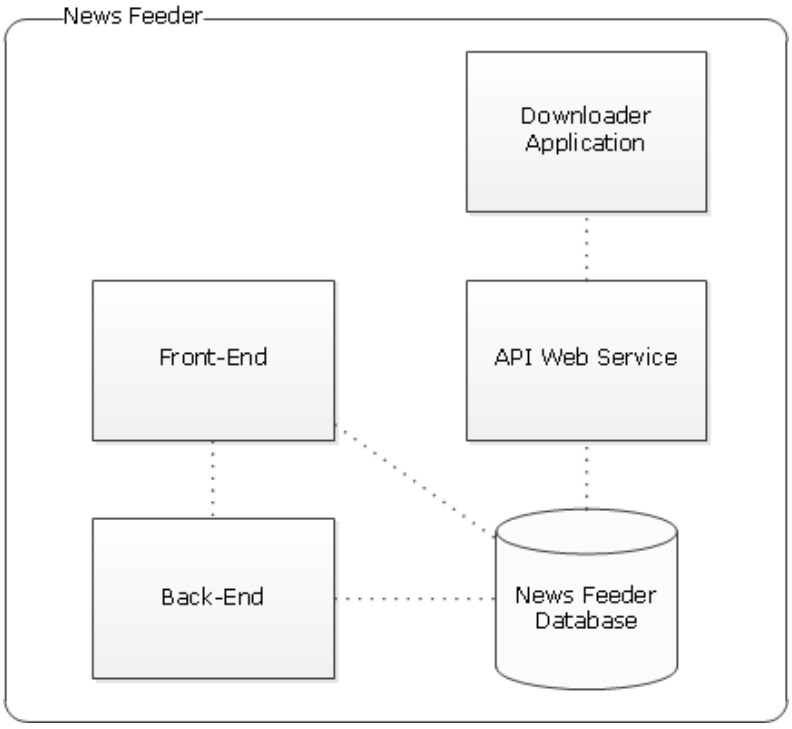
The service's structure can be broken down into 3 main sub-systems:

The Database – This is where user, sheet, feed and item information is stored. It is used by both the front-end and back-end components and acts as a repository between the two.

The Front-end – This is the interface of the system. It consists of a number of web pages accessible by users and administrators. It provides the means by which the users and administrators interact with the system (including the back-end). This interface provides methods for users to manage and view feeds, as well as managing the layout used to display these feeds. The interface also provides methods for administrators to manage users, feeds and items, as well as allowing them to manage and monitor the back-end processes.

The Back-end – This is an application that is continuously running in the background. It provides the main processing of the system. Its main role is to crawl and parse feeds and content. To perform this role this component contains a continuous queue holding references to all feeds and ordered according to crawl priority. The parsers mainly operate on RSS and HTML items, but other content sources (such as weather feeds, forums and social networks) can also be parsed for storable content.

Additional non-essential components of the service are the API web service and the downloader application. The downloader application (and other possible applications) access the system by using the API web service as a gateway.



Users

There are 5 main users of the News Feeder service:

Standard User – The main end-users of the system, including anyone in the general public. They have registered with the service and can specify feeds they wish to view. They can set a number of user preferences and settings regarding what content they wish to display and how they wish to display it (such as content filters and sheet layouts). They are expected to have basic computer knowledge (sufficient for using a web browser).

Subscribed User – Standard users who have subscribed to the system for a fee. As such they gain additional benefits related to their use of the system, such as access to custom sheet layouts and access to the downloader application.

Administrator – Managers of the service. They manage a number of system features and components (such as user accounts and items) and are able to manage and monitor system processes (such as starting and stopping the crawlers). They are considered to have more in depth computer knowledge than a standard user and understand the internal workings of the system.

Guest – Anonymous users that can register to the service in order to become standard users. Guests can access the main page, FAQ and are also able to view links sent by registered users of the system (such as a link to an article).

Web Master – Owner of an external site that is being crawled. They access the system via a portal referenced in the crawler HTTP headers in order to specify how frequently they prefer their site to be crawled (or if they want it to be crawled at all).

Constraints

The News Feeder project and its requirements are limited by the following constraint:

Network – The performance of the service will be dependent on the network connection used to access it. The requirements defined in this document assume a standard ADSL network connection.

Requirements

Priorities

The priorities used in the requirements have the following meanings:

Base - Requirements that must be met before any high level requirements can be implemented. These will definitely be delivered.

High - Requirements which are necessary to fulfil the core functionality of the system. These will definitely be delivered.

Medium - Requirements that provide additional functionality for the service. We will attempt to deliver all of these requirements.

Low - Requirements that are the stretch goals for the project. These will only be implemented if time presents itself.

Requirements Summary

ID	Requirement	Priority
F1.1	Front-end Database Interaction	Base
F2.1	Login	Base
F2.2	Manage Users	Base
F3.1	Add Feeds	Base
F10.1	Back-end Database Interaction	Base

ID	Requirement	Priority
F2.3	Account Registration	High
F3.2	Search For Feeds To Add	High
F3.3	View Sheets	High
F3.4	Additional Media - Images	High
F4.1	Manage Sheets	High
F4.2	Manage Sheet Feeds	High
F5.1	Back-end Monitoring	High
F11.1	Crawl RSS Feeds	High
F11.2	Crawl Priority	High
F12.1	Parse RSS Feeds	High
F12.2	Advanced Parsing Techniques	High
F13.1	Logging	High
N1	Accessibility	High
N4	Performance (Front-end)	High
N5	Performance (Back-end)	High
N7	Stability (Back-end)	High
N8	Maintainability	High
N9	Sustainability	High

ID	Requirement	Priority
F2.4	User Profile/Settings Configuration	Medium
F2.5	User Subscriptions	Medium
F2.6	Subscription Management	Medium
F3.5	Printing Support	Medium
F3.6	Additional Media - Video	Medium
F3.7	Additional Media - Maps	Medium
F4.3	System Defined Sheet Layouts	Medium
F4.4	Custom Sheet Layouts	Medium
F5.2	Manage Back-end	Medium
F5.3	User Statistics	Medium
F5.4	Abuse Reporting	Medium
F6.1	Content Filtering	Medium
F6.2	Content Blacklisting	Medium
F6.3	Bad Word Censoring	Medium
F7.1	Web Security	Medium
F8.1	Downloader Application	Medium
F9.1	Recommended Feeds	Medium
F9.2	User Sheet Update Notifications	Medium
F9.3	Mobile Support	Medium
F12.3	Retrieve Additional Media	Medium
F12.5	Tag/Keywords Generation (Categories)	Medium
F12.7	Multiple Content Sources	Medium
F14.1	Email Notifications	Medium
F15.2	API Web Service	Medium
N2	Ease of Use (User)	Medium
N3	Look and Feel (Aesthetics)	Medium
N6	Portability (downloader application)	Medium
N10	Help	Medium
N11	Scalability	Medium

ID	Requirement	Priority
F8.2	Mobile Application(s)	Low
F9.4	Social Network Authentication	Low
F12.4	Geo-location Retrieval	Low
F12.6	RSS and HTML Parsing Configurations	Low
F13.2	Auditing	Low
F14.2	SMS Messaging	Low
F15.1	Image Storage (Thumbnails)	Low
F15.3	Learning Algorithms	Low

Functional Requirements

Front-end

F1 – Database Interaction

ID: F1.1	**Requirement:** Front-end Database Interaction	**Priority:** Base
Description: The front-end is able to retrieve and store information about feeds, users and sheets in a database.		

F2 - User Management/Login System

ID: F2.1	**Requirement:** Login	**Priority:** Base
Description: The system will provide users and administrators with the ability to login to the system via the website by providing a username and password. After logging in, users are to be able to access account specific information (such as their feeds and sheets) and functionality.		

ID: F2.2	**Requirement:** Manage Users	**Priority:** Base
Description: The system will allow administrators to add/remove/edit users from the system for administrative purposes (such as removing users who have been inactive for too long). The editing ability also includes resetting user passwords/password attempts.		

ID: F2.3	**Requirement:** Account Registration	**Priority:** High
Description: The system will enable guests to register an account by providing an email address, username, appropriate password and other relevant user information. All account information will be validated on registration, such as checking for unique usernames.		

ID: F2.4	**Requirement:** User Profile/Settings Configuration	**Priority:** Medium
Description: Using the web interface, users will be able to modify their profile information (email address, name, avatar image, etc) and personal settings, such as how much priority they put on a particular feed, how often they wish to receive notifications.		

ID: F2.5	**Requirement:** User Subscriptions	**Priority:** Medium
Description: The system will allow users to subscribe/unsubscribe to the news feeder service to gain additional benefits such as access to the downloader application, customisable sheet layouts, increased Feed/Sheet limits and increased priority for content retrieval. Subscriptions will be recurring on a monthly basis for a fee.		

ID: F2.6	**Requirement:** Subscription Management	**Priority:** Medium
Description: The system will allow administrators to create/edit/delete/review user subscriptions via the web interface.		

F3- Feed Management

ID: F3.1	**Requirement:** Add Feeds	**Priority:** Base
Description: The system will provide users with the ability to add feeds to their sheets. The feeds to add can be newly created, pre-existing user feeds or default feeds.		

ID: F3.2	**Requirement:** Search For Feeds To Add	**Priority:** High
Description: The system will provide the users with the ability to search for feeds either currently in the database or from the internet (using a search API) matching specified search criteria (such as keywords, author and location). The resulting feed(s) can then be added to the user's sheet(s).		

ID: F3.3	**Requirement:** View Sheets	**Priority:** High
Description: The website will display sheet content in a suitable web format for the user to read.		

ID: F3.4	**Requirement:** Additional Media - Images	**Priority:** High
Description: The front-end is able to display image content that was included in the original item in standard HTML format.		

ID: F3.5	**Requirement:** Printing Support	**Priority:** Medium
Description: The system will provide users with the ability to print off their sheets and individual feed items in a readable format similar to that of the site.		

ID: F3.6	**Requirement:** Additional Media - Video	**Priority:** Medium
Description: The front-end is able to display video content that was included in the original item in standard HTML format.		

ID: F3.7	**Requirement:** Additional Media - Maps	**Priority:** Medium
Description: The front-end is able to display geo-locational content that was included in the original item. The content will be displayed using the relevant map API.		

F4 - Sheet Management

ID: F4.1	**Requirement:** Manage Sheets	**Priority:** High
Description: The system provides users with the ability to create sheets which contain a number of different feeds. Users will be able to set the sheet layout from a predetermined list (1 column, 2 column, and 3 column layout). Users will be able to delete their sheets.		

ID: F4.2	**Requirement:** Manage Sheet Feeds	**Priority:** High
Description: The system will allow users to add and remove feeds from their sheets. The will also be able to edit a feed's information in relation to how it is displayed on the sheet, such as what content (title, description, body, images) of the feed will display or how many items to show from that feed.		

ID: F4.3	Requirement: System Defined Sheet Layouts	Priority: Medium
Description: The system will provide a set of pre-defined layouts which the user will have access to when creating sheets and sheet views.		

ID: F4.4	Requirement: Custom Sheet Layouts	Priority: Medium
Description: The system will enable users to create their own custom sheet layouts via a drag and drop interface. Users will then be able to drag and drop feeds of their choice into the layout.		

F5 - Administration

ID: F5.1	Requirement: Back-end Monitoring	Priority: High
Description: The front-end web interface allows administrators to retrieve statistics on the back-end (such as crawler, thread and database statistics). Administrators will also be able to monitor the current back-end load.		

ID: F5.2	Requirement: Manage Back-end	Priority: Medium
Description: From the front-end web interface, administrators will be able to start and stop crawling as needed. Administrators will also be able to change the back-end configuration information such as the number of threads available to each part of the back-end system.		

ID: F5.3	Requirement: User Statistics	Priority: Medium
Description: The system will display basic user statistics (such as how many users, how many user feeds, etc) on the home page of the website.		

ID: F5.4	Requirement: Abuse Reporting	Priority: Medium
Description: The system will allow webmasters of content sources to adjust the frequency that the system crawls their sites (including the ability to disable all crawling), to ensure it does not exhaust their resources. This is done via a portal referenced in the crawler HTTP headers.		

F6 - Content Filtering

ID: F6.1	Requirement: Content Filtering	Priority: Medium
Description: The system will allow users to specify categories/keywords which they do not want displayed in their sheets. This can be used to filter out inappropriate material (such as pornography).		

ID: F6.2	Requirement: Content Blacklisting	Priority: Medium
Description: The system will enable administrators will be able to add specific sites/content sources to a blacklist that the system will ignore.		

ID: F6.3	Requirement: Bad Word Censoring	Priority: Medium
Description: Offensive words (listed in the database) will be censored in content displayed to the user. This will be a user preference.		

F7 - Security

ID: F7.1	**Requirement:** Web Security	**Priority:** Medium
Description: The front-end will use SSL for secure communications when communicating important information with the user (such as logins). This will be a user preference.		

F8 - Additional Applications

ID: F8.1	**Requirement:** Downloader Application	**Priority:** Medium
Description: The service will provide subscribed users with access to a downloader application. This application will provide a method for viewing sheets offline, either by providing PDFs or an offline version of the content.		

ID: F8.2	**Requirement:** Mobile Application(s)	**Priority:** Low
Description: Specific applications will be developed for mobile devices to provide access to and use the system.		

F9 - Miscellaneous

ID: F9.1	**Requirement:** Recommended Feeds	**Priority:** Medium
Description: The system will be able to suggest relevant feeds the user may be interested in. This will be based on the feeds the current user is and has been subscribed to, and any relations between these feeds (such as keywords, author, location, source, etc).		

ID: F9.2	**Requirement:** User Sheet Update Notifications	**Priority:** Medium
Description: The website will display the number of new items available in a content area on a sheet for each user. The number of sheets which have new content available will be displayed in the global menu area.		

ID: F9.3	**Requirement:** Mobile Support	**Priority:** Medium
Description: The web site will support mobile access. All pages and content will be able to be viewed on mobile devices (without issues and in the correct format, including sheet layout).		

ID: F9.4	**Requirement:** Social Network Authentication	**Priority:** Low
Description: The front-end will integrate with social network authentication providers (such as Facebook, Twitter) to provide access via users' respective logons.		

Back-end

F10 – Database Interaction

ID: F10.1	**Requirement:** Back-end Database Interaction	**Priority:** Base
Description: The back-end is able to retrieve and store information about feeds and crawled/parsed content in a database.		

F11 - Crawling

ID: F11.1	**Requirement:** Crawl RSS Feeds	**Priority:** High
Description: Given an RSS feed, the back-end will crawl it and retrieve items for parsing into content. It will continue to crawl the feed for new content on a regular basis.		

ID: F11.2	**Requirement:** Crawl Priority	**Priority:** High
Description: Based on a number of feed statistics the crawler will dynamically arrange the feeds so that feeds of higher priority are crawled more regularly than others. The priority will be based on properties such as; the average amount of time between content updates, the number of subscribers and feed type (new, user defined, system defined, etc).		

F12 - Parsing

ID: F12.1	**Requirement:** Parse RSS Feeds	**Priority:** High
Description: Given RSS XML input the back-end will extract information (such as feed title, author and content).		

ID: F12.2	**Requirement:** Advanced Parsing Techniques	**Priority:** High
Description: If an article's full content is not provided in the RSS feed, the back-end is able to apply techniques which will retrieve the full article content.		

ID: F12.3	**Requirement:** Retrieve Additional Media	**Priority:** Medium
Description: The back-end is able to retrieve non-text elements (such as images and videos) that are related to the article by parsing the actual HTML page it is displayed on.		

ID: F12.4	**Requirement:** Geo-location Retrieval	**Priority:** Low
Description: Given an article the parser will attempt to determine locations relevant to the item (this will work by using any available Maps API references in the RSS, or by looking for place names in the body).		

ID: F12.5	**Requirement:** Tag/Keywords Generation (Categories)	**Priority:** Medium
Description: The parser will be able to retrieve/generate tags/keywords for an item, to assist in the searching of items. These will be generated using the attributes, such as the description, if not directly available in the RSS feed.		

ID: F12.6	Requirement: RSS and HTML Parsing Configurations	Priority: Low
Description: The system will allow administrators to configure the back-end to parse items differently based on heuristics within the RSS/HTML (e.g. meta-tags identifying creation with WordPress) using configuration files.		

ID: F12.7	Requirement: Multiple Content Sources	Priority: Medium
Description: The back-end will be able to crawl and parse content from sources other than RSS (such as weather, email, social networks, forums, etc).		

F13 - Logging

ID: F13.1	Requirement: Logging	Priority: High
Description: The back-end will provide error logging and optional verbose logging. These will be output to log files which are accessible on the back-end server.		

ID: F13.2	Requirement: Auditing	Priority: Low
Description: The system will provide logs of users' interactions with the system in the form of audit trace logs.		

F14 - Notifications

ID: F14.1	Requirement: Email Notifications	Priority: Medium
Description: Email notifications may be sent when new content is found for a user's sheet. Alternatively, it may be configured to send 'batch emails' which consist of all updates over a period of time.		

ID: F14.2	Requirement: SMS Messaging	Priority: Low
Description: The system will provide update notifications to users when new content becomes available via SMS messaging.		

F15 - Miscellaneous

ID: F15.1	Requirement: Image Storage (Thumbnails)	Priority: Low
Description: Images will be resized and stored in the database to reduce bandwidth usage in the user's browser, prevent linking to missing images and also prevent possible HTTPS issues.		

ID: F15.2	Requirement: API Web Service	Priority: Medium
Description: The system will provide an interface for the downloader application and potentially, other 3rd party applications. Its purpose is to act as a gateway for external systems to access content.		

ID: F15.3	Requirement: Learning Algorithms	Priority: Low
Description: The system will be able to gradually learn the best method for parsing particular content sources. This will be based on what the system has learnt and parsing configurations from administrators.		

Non-Functional Requirements

ID: N1	**Requirement:** Accessibility	**Priority:** High
Description: The system will be accessible via any modern browser (Chrome, Firefox, Safari, IE7+) on any modern Operating System (Windows, Linux, Mac).		

ID: N2	**Requirement:** Ease of Use (User)	**Priority:** Medium
Description: Users will be able to learn to use the features of the system within a short time period (e.g.10 mins), without any help/training external to the site.		

ID: N3	**Requirement:** Look and Feel (Aesthetics)	**Priority:** Medium
Description: The front-end interface should be intuitive, consistent and familiar to users, such that they can easily navigate and use the site.		

ID: N4	**Requirement:** Performance (Front-end)*	**Priority:** High
Description: The front-end sub-system will adhere to the following performance metrics: • Page Load – All pages should load (excluding media, such as images, video, maps) in under 2 seconds. • Searching for News Feeds – First results should be returned within 5 seconds. • Crawling Feeds – First results should be available within 1 minute.		

ID: N5	**Requirement:** Performance (Back-end)*	**Priority:** High
Description: The back-end sub-system will adhere to the following performance metrics: • Responding to Front-end – Calls will on average take 1 second to respond. • Crawling and Parsing a Feed Item – Crawling for a feed item (including media, patch and match) will be completed within 5 seconds. • Queuing – One queue item will not be in the queue longer than its expected update time without being crawled.		

ID: N6	**Requirement:** Portability (downloader application)	**Priority:** Medium
Description: The downloader application will be available on all modern platforms (Windows, Linux, Mac OSX)		

ID: N7	**Requirement:** Stability (Back-end)	**Priority:** High
Description: The back-end is able to handle invalid inputs and malicious SQL without error. Any errors which occur will be handled cleanly, as will service stops and restarts.		

ID: N8	**Requirement:** Maintainability	**Priority:** High
Description: The source code of all subsystems will conform to code styling standards and implement good design principles, which will allow for future upgrades and maintenance without requiring a system overhaul.		

***** *Please note that these performance requirements exclude possible network limitations*

ID: N9	**Requirement:** Sustainability	**Priority:** High
Description: The system will have mechanisms in place to maintain itself without the need for external interaction (e.g. the system will clear the logs automatically after a set period/size limit).		

ID: N10	**Requirement:** Help	**Priority:** Medium
Description: The site provides help and FAQ for the user, both directly (i.e. showing a help page and tutorial) and indirectly (i.e. descriptions like tooltips).		

ID: N11	**Requirement:** Scalability	**Priority:** Medium
Description: There can be an unlimited number of users, sheets, feeds and items. The system will be able to cope with increased numbers without requiring changes to the internal subsystem (i.e. changes that require re-compiling/deployment).		