Utilization of Tags in a Knowledge Bank

Simon Johansson

ic08sj9@student.lth.se

Department of Design Sciences at the Faculty of Engineering
Lund University

Supervisors:
Susanne Frennert
Ulf Svensson

August, 2014

Abstract

In this age of information, a person is accustomed to on demand and within seconds retrieve all thinkable information between heaven and earth with the help of a technical device. It's therefore important that information is stored and presented in such way that a person can embrace it with ease. The IT company Nilex, where this research took place, provides such a storage in the form of a knowledge bank. The premise for this thesis is to investigate how information from a knowledge bank can be displayed and found with the principles of interaction design. A centralized focus of the research was if tagging functionality could be a positive effect for this purpose.

This thesis describes the problem identification phase followed by the solution design phase which also includes testing of these proposals. The conclusions drawn from the discussion of each method were essential to move the project forward. The testing was performed in two phases, first without tagging functionality and after with, to try to answer if the hypotheses surrounding tags could be answered. No tests or implementations to a real software were done during the research but conclusion from the tests on the design proposals indicates that tags may have a positive effect which makes it applicated to a real environment.

Table of content

Keywords

Article   -   Document inside the knowledge bank with useful information.

Agent   -   Person handling incidents reported to the helpdesk.

Incident   -   A problem reported in the helpdesk.

ITIL   -   "Information Technology Infrastructure Library". A set of practices for upholding an IT system.

KB   -   Knowledge bank, place where information is stored and presented.

KM   -   Knowledge management, organizational practises for handling knowledge.

NSP   -   Nilex Service Platform,Nilex' ongoing developing software where the new knowledge bank will be.

Tag   -   Descriptive text of an object.

Ticket   -   See Incident.

Widget   -   A part of the graphical user interface that displays information.

1 Introduction

Nilex has for several years delivered successful solutions in the IT service management segment [1]. Being responsive to customers' needs and challenges in the progressive IT field have made it possible to deliver more than 1000 solutions from their Delphi-developed system to the public and private sectors. At the moment, they are about to release a new and future proof system based on ASP.NET MVC framework which delivers a modern web experience with techniques such as HTML5, Javascript and C#. An important part of the system is the knowledge management, in particular the knowledge bank, which is seen as a competitive extra edge. It is therefore of Nilex biggest interest to have a knowledge bank that the customer understands and appreciates for the value it can bring to whole system.

1.1 Purpose

The main purpose for this thesis is to investigate the knowledge bank concept with focus on Nilex's current system and from that specify and implement improvements. Some vital points are as follow:

- Investigate what symbolize a knowledge bank with good interaction design practices.

- Make the search function deliver relevant information, i.e articles about problems the user wanted information about.

- Create underlying structures (tags) for a better knowledge bank experience.

1.1.1 Hypothesis

1. A knowledge bank with a tagging feature is preferable against a knowledge bank without a tagging feature.
2. A tagging feature can improve the usage of a knowledge bank.
3. A tagging feature can help a service desk agent understand when it's time to create knowledge bank article.

These hypotheses will be tested by letting users test two designed interactive knowledge bank systems, one with a tagging feature and one without. The users will thereafter rank which of the two choices they preferred to test the hypothesis. To be noted is that the conclusions will be drawn from qualitative and not quantitative metrics which mean that it is opinions and not undeniable facts.
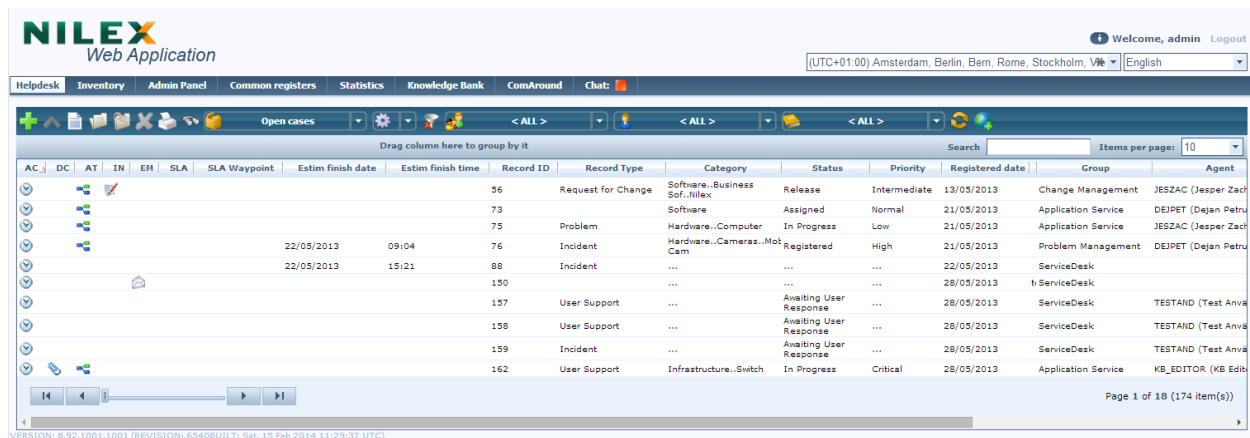
1.2 Background

This chapter explains the software Nilex is developing and why a knowledge bank is essential for their work. The final part mentions the limitation of this research and why those bounds were chosen.

1.2.1 Nilex software

Nilex successful service management system comprises of several parts which makes it a unique holistic approach. Depending on the customers' needs and size are specific tailored solutions proposed to match their requirements. The main functionality is however revolved around ticket management. Tickets are collected information regarding things like:

- Incidents or problems, both for within the IT-environment and other business related aspects.

- Requests, for example purchases and missing functionality.

Figure 1.1 Ticket overview in Nilex

These tickets are created and handled by agents from inside what is called the helpdesk. What action an agent is allowed to perform on a ticket depends on a permission system. For example may not all agents have the permission to close a ticket when they think it's resolved but instead it needs to be done in a higher instance. The view an agent is presented with inside the helpdesk is a list of available tickets (see figure 1.1). Except the ticket type, other classifications such as "priority", "status", "registration date", "estimated time for solving the incident" and "linked agent" are displayed and can be sorted upon.
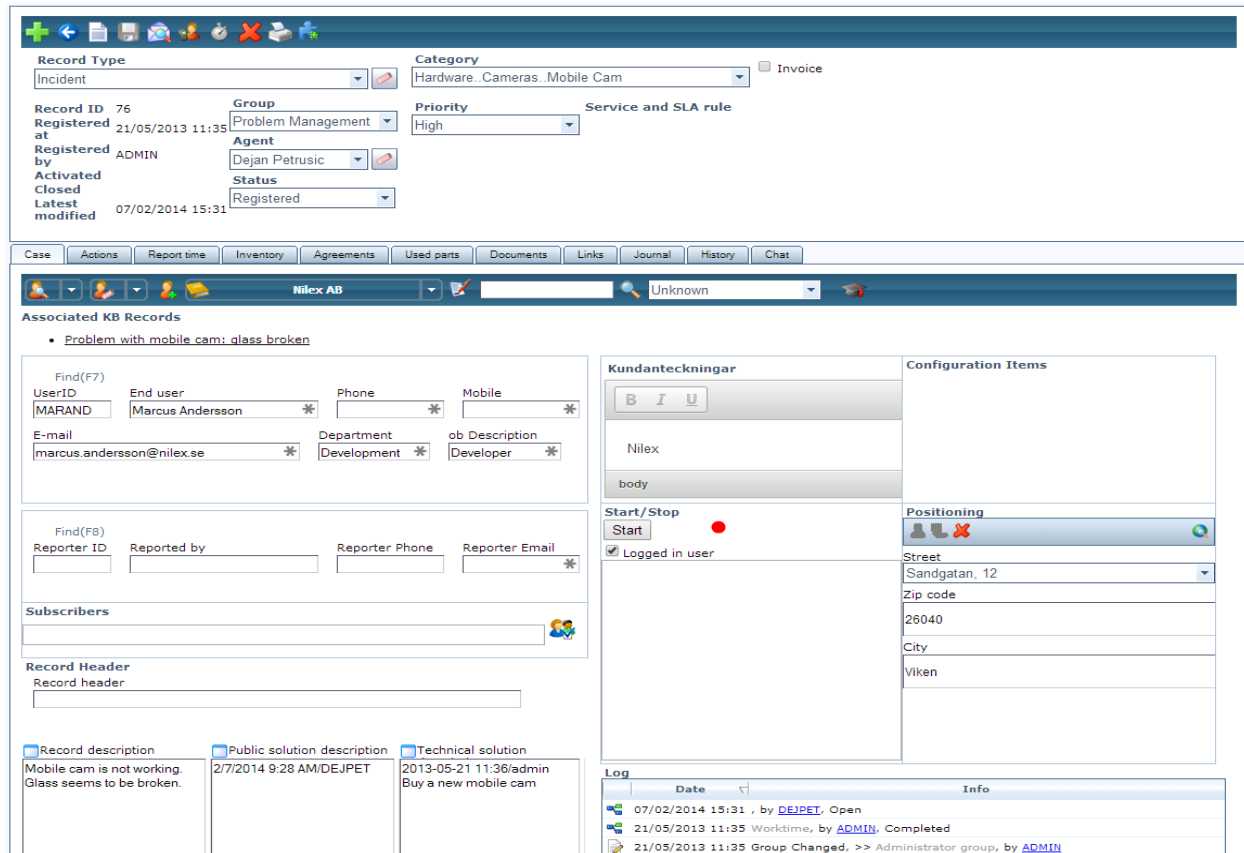
Figure 1.2 Overview inside a ticket

Selecting a ticket will bring up a view with many configurable options (see figure 1.2). The most fundamental aspect is to see who created the ticket and the reason behind it. Depending on what modules are connected and used can the ticket, for example, be specified by a Service-level agreement. In practice is this a time metric for how long a person has worked on an issue and when it's estimated to be finished. Nilex also offers the possibility of linking inventory objects the company is managing, such as mobile phones, computers etc., to a ticket for a better understanding of what physical components is of interest for the issue.

1.2.2 Knowledge Bank

The warrant for a knowledge bank in the system is to ease the burden on the helpdesk. If the users can solve their own problems without the need of contacting a person for handling the issue, or even technicians further down the line, both time and money be saved. The current system's implementation of the knowledge bank may have the fundamental parts for a knowledge bank but there is room for improvements. One of those possible improvements is the tagging feature. Now, with the implementation of their new modern web solution, Nilex is interested in that the customers really understands the value of a knowledge bank. Understanding what customers appreciates, together with how other successful knowledge

banks are designed will this thesis try to shed some light over good knowledge bank practices. This research is also interesting for any platform gathering and sharing data, something which is essential in this age of information.

1.2.3 Limitation

The focus for this thesis has been on the conceptual design and the proposal of requirements for knowledge bank functionality with special focus on tagging. The knowledge bank is only one part of the software Nilex provides and therefore has the investigation in other parts been left out. To be noted is that without the overall acceptance that it is a well provided system, parts such as the knowledge bank can be of minor use.

There have been real world implementations but that process wasn't finalized as the development is ongoing. To make a complete implementation from the ideas proposed in this thesis was not feasible for the scope and time. The goal was to create an underlying structure, like tags, to the system as a start for further development.

Technical limitations have been considered with the design but a degree of freethinking was still there. That meant in practices that if some design moves couldn't be translated to the real world application, a compromise reached to achieve the thought behind the design.

The final testing was done on a rather rough visual version of the design. A few testers may have misunderstood some components because of this reason but nothing essential. The main focus was to test possible functionality and secondarily its looks. The reasons for the rough visuals was time constraints and aesthetic shortcomings by the researcher.

2. Theory

This theory chapter tries to explain the roles behind different parts of the software Nilex provides. Knowledge bank is in reality a pretty small part for solving bigger problems which cannot be forgotten. The first sections describes these bigger problems and is gradually aiming towards definition for the role of knowledge bank and the possible role and description for the tagging functionality.

2.1 IT-services

To understand the role of a knowledge bank in an IT-system there needs to be some understanding of the impending bigger picture. As the name suggests in IT, information technology, *information* is a keyword. The quality of IT-services is measured on how an organisation is producing, distributing and analysing information. Structural investments must be made to accomplish this even though many organisations disregards it or only has an extremely shallow implementation to acknowledge it.

Organisations are dependent on that the symbiosis between the IT and business-side is working to achieve good IT-services. The main problem is that the two fields can in unsuccessful environments talk two different languages with no translation, something which "IT-service management" is meant to handle. In "An Introductory Overview of ITIL" is a service described as "*... a means of delivering value to customers by facilitating outcomes customers want to achieve without the ownership of specific costs and risks*" [2]. This description means that when a customer specify an outcome by an IT-service such as "sales people getting weekly updates of sales" it is processed as "email-service that collects list of weekly sales and sends an email to every sales person's email-inbox". It is the "IT-service management" objective to achieve the customer specified outcome by knowing cost and risks for a service as well as the value it brings. The "IT-service management" is therefore described in "An Introductory Overview of ITIL" as "*... a set of specialized organizational capabilities for providing value to customers in the form of services*". One common guide of practices of making these organizational capabilities valuable for all parts involved is the ITIL.

2.2 ITIL

The "Information Technology Infrastructure Library", ITIL, was an initiative by the UK government in the early 80s [3]. The purpose was to have standard practices for both private and government agencies to avoid the ongoing growth of unique IT management practices. It was also in the mind of supporting the transition from centralized IT structures to solutions where location and distributed computer power was no longer issues. The approach ITIL takes is described as a holistic view in managing services with end-to-end focus in comparison to old IT mindsets. That means that every service is considered in a bigger picture to achieve the expectation of the business customer and not just looking at different parts separately. The following section describes the simplified steps in the service lifecycle to accomplish that.
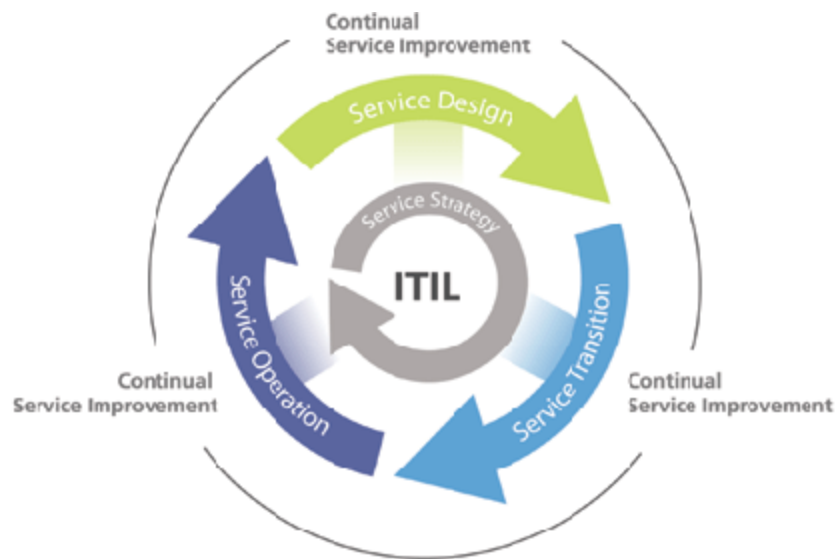
Figure 2.1 ITIL process [4]

Service Strategy

The first step in the lifecycle is the "Service Strategy" which all the other steps are built upon. To thoroughly understand all the different aspects involved from who the customer is, what services are required to fulfill their needs, what IT resources are required to develop these services as well as the specified demands to execute it are defined here. The most important factor to consider when dealing with the "Service Strategy" is that the cost doesn't exceeds the final value for the customer.

Service Design

This step's purpose is to make sure that the services are designed with efficiency in mind. To reach what the customer expects and at the same time be cost-efficient is vital for successful

service management. Both tools for monitoring services and tools that provides metrics of the efficiency must be considered.

Service Transition

During the "Service Transition" occur the implementation, testing and production so that the customer receives its specified value. Several processes are compromised in this phase but the most important one for this paper is the knowledge management which will be described in more detail.

Service Operation

The role of the life cycle phase "Service Operation" is to ensure that the produced service continuously works appropriately. Some of the tasks that are involved are the handling of disruptions and how to get the service back to its normal state as well as finding reason for the disruption.

Continual Service Improvement

Enclosing the whole service life cycle is the "Continual Service Improvement". Its purpose is to investigate that the service level is appropriate for each stage or otherwise deliver improvements to make it so.

2.3 Knowledge Management

The importance of knowledge has always been a main pillar for the foundation of the human society. From the early days of agriculture where the offspring from the first farmers were taught how to grow crop to sustain a livelihood to Johannes Gutenberg's invention of the printing press to spread the written word, are both enormous milestones for the progress of the society. The exact definition of knowledge has been debated since ancient times by philosophers and isn't something this paper will dwell further on. There are however definitions of variations of knowledge in regards to organizational "Knowledge Management" that is of interest. These are tacit and explicit knowledge with the extension in some research with implicit knowledge.

The work of Nonaka is one of the most cited for defining the first two mentioned types [5]. The tacit knowledge is described to be deeply linked to experience and participation in contextual events. It is furthermore divided into both technical and cognitive aspects. The technical side is actual expertise and skills to contribute to a specified context while the cognitive side is a person's opinion or belief for it. An example of the tacit knowledge is to know how to interact

with a customer in the context of verbally and body language expressions to suit that particular customer. Explicit knowledge on the other hand is knowledge that can be codified and expressed through dialog and written text.

The view of only two knowledge types do Koenig think is too simplified to be applicated to the real world [6]. Instead is implicit knowledge proposed to be included as description for intangible knowledge that can be processed to be explicit knowledge.

One of the earliest attempts for a definition for administering such mentioned knowledge, i.e "Knowledge management", was made by Davenport (1994); "Knowledge management is the process of capturing, distributing, and effectively using knowledge" .
As the years passed and with that the growing usage of "Knowledge management", a broader definition, which undoubtedly is one of the most cited today, was proposed by the Gartner group;

"Knowledge management is a discipline that promotes an integrated approach to identifying, capturing, evaluating, retrieving, and sharing all of an enterprise's information assets. These assets may include databases, documents, policies, procedures, and previously un-captured expertise and experience in individual workers".

An important topic to furthermore explain the combined entities and structures that embodies knowledge management is "Lesson Learned databases" [6]. The simple premise is that knowledge obtained by a person "doing" something must be saved as a future explicit reference as it isn't a natural step of events. In the early stages of Knowledge management was the terminology "Best Practice" used for describing such occurrences. This was later disregarded for the expression "Lesson learned" as the former expression is subjective and dependant on context for what really is the best practise. Koenig explains further that military was one of the first instances that embraced the concept of "Lesson learned" as the debriefing after missions. The person performing the mission often doesn't have the time to convert the performance to explicit knowledge is the task instead passed onto a person that works with KM. The same commitment can be transformed to other organisational projects where team members in most cases after finishing a project just abandon it with no follow-up. Understandably, that is one of most important objectives for a person working with "Knowledge Management" to capture and store such information. There must also be some sort of directive and review process for accepting new content to be included in the database as well as functionality to determine and set an expiry date.

Another important task for knowledge management is to identify location for expertise. Previously the term "Yellow page" system was used to describe it but nowadays the more correct and precise terms "expertise locator" or "expertise location" are the standard. The information provided for building such system is from employee resumes and their self-awareness of knowledge that lies within them by answering surveys. Even smart algorithms that searches through email communication by employees to discover and identify

expertise is sometimes used. When the expertise is listed, some sort of prioritisation of the available experts must be considered for not overwhelming a particular expert and that really complex problems are sent to the most suited one for solving it.

The last topic for providing a well thought out Knowledge management system is by "Communities of practice". The primarily role is to supply an online community where knowledge and experience can be shared between persons. The community is often closed connected to the organisation's intranet for discussions and help regarding problems that occurs. It's important that there are persons linked to specific roles for managing the community [7]. Their objective is to make sure that correct and valuable information is distributed to the community.

2.4 Knowledge bank

One can argue that Knowledge bank is foremost a type of "Communities of practice", but several of the mentioned subjects in KM can be considered a part of it. It's difficult to determine the general definition of what a knowledge bank is since many institutions and organisations uses the terminology differently. In the sense of this paper does it mean a place where knowledge can be shared both or either internally and publicly, depending on the IT-system's application area [8].

In "A simple guide to creating a knowledge base (and keep it running)" the author discusses good practices for a knowledge bank [9]. She explains that the primary fundamental is to have an understanding of the target audience. The knowledge bank can be a bunch of sorted articles or a wiki but as long as the people who will use it don't understand the structure or language it is meaningless. Therefore it is essential to investigate the user base by spending time and effort with the users. Common methodologies as interviews, surveys and observations are proposed as support.

2.4.1 Roles

As part of the sustainability of the knowledge bank, user roles with different rights must be considered. This is a list with roles that in some implementations can be merged to a role with wider privileges.

-Role that adds content.

-Role that reads content.

-Role that manage content.

-Role that review content.

The action to accept submissions from more users than only the ones with administrative rights is a good way of building a broad and information rich knowledge bank, but the content must most likely be reviewed. The reviewing process can include several people with different tasks;

-Is the information understandable, as in correct format and linguistic?

-Is the information technical correct?

-Is the information of value to justify it to be shared?

-Is the information specified to be shared with accurate counterparties without compromising possible confidentiality?

-Is the information published in the correct context to be able to be found?

The importance of the last point is not to belittle. If the information of interest isn't easy accessible or is even difficult to find means that the usability is questionable. A fundamental first step can be categorizing the information together with creation of sub categories. With a tree structure can the user easily navigate up and down through the nodes to find the area of interest for the wanted knowledge. Visual cues are very important in this regard to know exactly if there are parent and child nodes worth traversing to.

2.4.2 Tags

Information needs to be structured in some way to be found and helpful. The paper "Usage patterns of collaborative tagging systems." explains that the most common taxonomy of information is hierarchy and exclusive based [10]. A typical example is a normal folder structure for a computer file system. If you have three documents regarding "apple pie", "hamburger" and "pizza" and want to find them easily you have to place them in strategically chosen folders. "Apple pie" can be placed under the "Dessert" folder and "hamburger" and "pizza" are placed under "Main courses". Both these folders are placed within the "Food" folder to complete the taxonomy. This described system is what's called hierarchy and exclusive based which in comparison with tags that are non-hierarchy and inclusive.

With tags can objects be described with several parameters instead of just their name and the name of the parent folder. A "main courses" folder that is filled with content makes it hard to find a particular "pizza" document if that is of interest. It would favour the search if that "pizza" document also was tagged with both "main course" and "fast food" to decimate possible results. This implies that the constructed tags are correct and understandable to be helpful.

The nature of a tag creates three different description problems. These are synonyms, polysemy and basic level variation. Synonyms are described as the most troublesome problem. Tags that are described with words that nearly mean the same thing can fall under the radar for some users if the "right" synonym wasn't searched for. In a collaborative environment must this either be solved by setting standards or accepting a wide range of tags on an object which makes the whole system more complex.

Polysemy indicates words which nearly have the same area of use from the same definition. An example is the term "computer problems" which can be problem with the physical computer object or within software running on the computer. Searches can therefore give unwanted results even if it was a correct choice of search term.

Basic level variation means that from one user to another can the descriptiveness of an object vary. The definition of basic level is what is set as the standard from a human interaction point of view. An example is a dog of the breed "Golden Retriever" is by the definition of basic level most commonly indicated by "dog" instead of "Golden Retriever" or "animal". But a user with expertise in a field may be more specific so in this case can a dog expert use the term "Golden Retriever" more likely. Hence, the problem can rise of basic level variation.

When a research investigated a collaborative tagging environment from the social website of Delicious could they quantify seven different types of tags that were user created and managed [10]. These seven were as follows:

1. Noun of what (or who) the object is about.
2. Identification on what it is.
3. The creator of the object.
4. Categorising by counter. These tags doesn't work on themselves but was often used as numbers like "25" and "100" indicating a collection of some sort.
5. Adjectives for its characteristics.
6. Self-reference, like "my" or "mine".
7. Task organizing, if the object could help perform an task it could be tagged by for example the terms "toread" or "jobsearch".

These collaborative tags, or social tags, may not suit everyone's interest as many tags are for a personal level. This is also seen in a research about tags for the music site "last.fm" [11]. Tags like "best song" or "saturday music" is mainly of interest for a particular user and not for the whole user base. Other problems with social tags are that it's hard to define a common ground for example what is "pop" or "rock". Music also changes as times goes by which mean that a particular genre may sound completely different with a decade apart. Misspelled tags are also described as problem social tags have to deal with.

Delicious' and Last.fm's way of handling tags implicative introduces a freedom as there are no set connections between the tags. Another big internet website is Wikipedia [12], a

collaborative encyclopaedia, has another approach of tag handling with a bit more structure [13]. It is based from a thesaurus perspective for controlling different relationships. Thesaurus is a word when introduced to the English language in the year 1736 had the broader meaning of "a treasury or storehouse of knowledge, as a dictionary, encyclopaedia and the like". With the release of Peter Mark Roget's book "Thesaurus of English Words and Phrases" in 1852 it introduced the linguistic thesari. It helped the reader find synonyms or more detailed description of different kind of words. As information retrieval and indexing became common concepts the word thesaurus evolved with it. Today there is an international standard for thesaurus, ISO 2788, which defines it as "the vocabulary of a controlled indexing language, formally organized so that the a priori relationships between concepts are made explicit". There are three different kind of relationships:

- Equivalence
- Hierarchy
- Association

Wikipedia uses these three relationship terms to define their tag handling. Equivalence for redirecting terms to a correct corresponding article, hierarchy for categorization and sub-categorization between articles, and association for linking of related articles within the same category.

The three mentioned structures can be represented by figure 2.2 on how they differentiate from each other.

Classification(filesystem)   Unconnected tags (Delicious)   Thesaurus (Wikipedia)
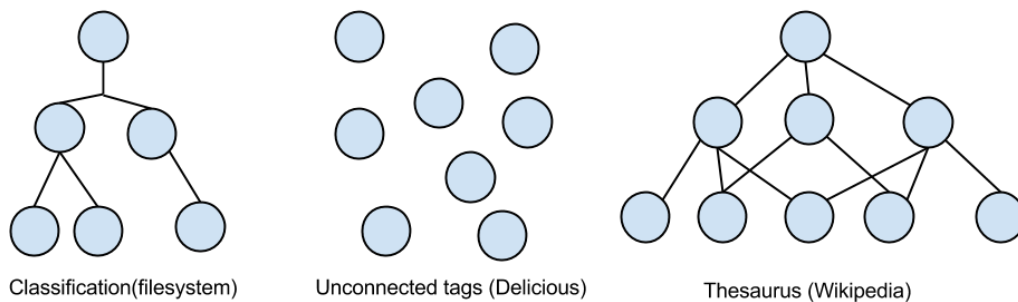
Figure 2.2 Variants of tag systems

2.5 Previous work

Some previous work has been done in regards of the tagging concept but not in the context of a knowledge bank. Despite this did the results from these papers still provide important conclusions that were used for this research.

A broad research about tags was made in "Usage patterns of collaborative tagging systems. Journal of information science 32.2 (2006): 198-208." where the concept is investigated and thereafter is the implications it brings to a social site looked at [10].

If tagging is a successful functionality for a music website by tagging song, artists etc., is investigated in the paper "Social tagging and music information retrieval." Journal of New Music Research 37.2 (2008): 101-114" [11].

3. Methodology

This chapter explains how the theory is applicated to the chosen methods for the "action research" method process. First there is a theory part over the chosen methods and what they excel at to perform this research. That is followed up by the execution of methods to achieve the purpose of this research. All methods are in chronological order as they occurred even if some happened simultaneously. Many of them also have a result part which is presented in their corresponding subheading.

3.1 Interaction design

A technical system can be extremely complex with endless of different functionalities to solve an equal amount of problems but not necessarily with the user of the system in focus. At glance can it appear as such solid systems are worth investing both time and money in but actually may end up hurting the business. There are countless of examples where a company or an organisation wants to expand by reconstructing a vital system and the production instead gets worse because the users don't understands it. Interaction design, IxD, are practises for making interactive systems on the terms of the user and not on the terms of the system [14]. The Interaction design association is a group dedicated to sharing and discussing interaction design and they define the term as follows:

 "Interaction Design (IxD) defines the structure and behavior of interactive systems. Interaction Designers strive to create meaningful relationships between people and the products and services that they use, from computers to mobile devices to appliances and beyond" [15]

3.1.1 Principles and techniques

A proven design principle that is applicable for the computer interface is "Fitts's law". The principle says that "The time to acquire a target is a function of the distance to and size of the target" [14]. This means that important and frequently used objects should be bigger and closer to where the user navigates. The principle may seem straight forward but regardless is it still overlooked. One example from the real world is that the law suggested that the Windows start menu in fact was built backwards for its usage, with the least used options closest to the main button, which also was shown to be the case.

Affordance and natural mapping are two other important topics in the schemes of human-machine interaction [16]. It's about knowing what to do by just looking at an object or interface. An example from the real world is that a person understands that to be able to open

a door with a door knob it needs to be turned. If it instead is a door handle, the correct approach is to pull the handle downwards. To make sure that the mental picture of how to interact in a situation also gives the desired outcome without it causing confusions is what symbolizes good affordance. In the sense of a computer interface does it mean that buttons work as they are supposed to do and symbols for icons is a natural mapping representation of their actual meaning.

Goal-driven design is one of the essential concepts that interaction design can be driven by. The ideas has been evolved from the concept of user centered design but now includes all the possible stakeholder [17]. That means using elicitation methods for both the end users and the business side are important. From the end users perspective is to understand the reason why they are using the the system and from that design a particular well thought out solution for that special reason the main goal to solve. The constraints that have to be dealt with can be found by performing user researches that matches the type of the stakeholder [18].

3.2 Performed methods

The following section describes the theory behind the methods used for understanding the user base and their problems. The methods were chosen on the basis of being feasible in the time scope and their familiarity for the researcher as well as recommendations from theory studies.

3.2.1 Meetings

There are several reasons for conducting meetings in a professional environment. These reasons could be information sharing, decision making and other idea and opinion discussions. It is also a place for team members to gather for debriefing and other activities connected to the group maintenance. There are different kind of meeting types to choose between depending on what is most suitable for the activity [19,20,21].

Formal meeting is a type where often a structured agenda is set and participants for the meeting are defined by roles. The leading role is a chairperson whose main purpose is to make sure that the agenda is followed and control the order of speakers. The administrative role of documenting opinions and conclusions is done by a secretary. This role also has the responsibility of booking a meeting room and make sure that the participants receives the documents needed for engaging in the meeting. A good general participant to the meeting must be up to date with the discussion points and see beyond interpersonal conflicts and opinions to make the meeting constructive. Understandably, these kinds of meetings are well prepared in advance so they can be performed in a smooth manner without hiccups.

In comparison with formal meetings are informal meetings not as strict in any regards. For example isn't there any particular precondition or preparations that must be met. It's a more

relaxed environment which in many cases means that it can occur on a location that suits the participants better instead of the ordinary boarding room. The relax environment may also improve the ability of sharing unfinished ideas and opinion as the pressure of result isn't as high. The group of people meeting is also often smaller which makes it easier to speak more freely.

3.2.2 Interview

Interviewing is a good method for receiving understandings for the current work situation. That gives guidance to defining a future system where the current problems are gone. Though interviewing will not reveal the complete problem picture it's often used as one of the main methods. The selection of person to be interviewed is an important choice [22]. Often is a middle manager selected as a representative to speak for a group connected to the problem area. The faulty idea is that this person knows and understands the problem for all the employees and is therefore most suitable. This has unfortunately been proven to sometimes be incorrect. It's therefore important to reach out to more users by either more interviews or other methods to get a broader picture.

When performing the interview is it described as good practice to follow the interviewee instead of taking a strict authoritarian approach. The consensus is therefore to prepare some questions but depending on answers and mood of the interviewee to not be afraid of going beyond of what was prepared. Although it's important to not get too side tracked as well to be able to ask all the prepared questions.

3.2.3 Questionnaire

The previously mentioned techniques are methods of receiving information from fairly few people. A questionnaire is instead a valid possibility to retrieve information from many users under fairly easy conditions. Its purpose is to see if purposed assumptions can be proven statistically and gather other ideas or opinions.

One solution of the questionnaires format is to split the questions into two parts, closed and open questions [22]. With closed question can issues connected to a current system be asked by selecting an answer from multiple choices. These answers are easy to represent by statistics which isn't possible in the same way with an open question. With open questions can the participants with written answers describe what can be improved or changed. These answers need to be interpreted by the person responsible for the questionnaire which isn't always an easy task. It is therefore of highest importance that both the open and closed questions are tested before sent out.

3.2.4 Brainstorming

Brainstorming is a way of letting a person or a group of people think freely about ideas. Most often is some sort of documentation used, like a whiteboard, for writing down ideas. Important to note is that no idea should be treated with disregard in this step. Whatever the idea might be, it should also be put down on paper. Sometime after, but not directly, should the ideas be prioritised by what is feasible and what is not. The extra thinking time means that ideas that were considered unrealizable at first, may now be seen in a new light.

3.2.5 Testing

Usability testing is necessary to provide a good evaluation of a design [23]. Different kind of testing methods can be used depending on the current phase of the development. In the beginning, the first ideas are tested to be reworked for the future design. As the project progresses more extensive testing is performed to see if the first issues are solved by observing and taking notes on how a user group solves a task. This can give both quantitative data, for example in the form of time of finishing a task, and qualitative data, where the users opinions and thoughts about the design is noted.

Paired comparison testing

A typical answer from a user after an unstructured user interface test is that they did or didn't like it. Such answer doesn't give much information to work with and may only answer if they managed to navigate at all. A successful design isn't a one-solution riddle but instead must the designer consider many aspects to make a valid choice. By comparing design proposals, competitors and previous design choices for a system against each other can the tester with greater confidence answer what they prefer and why [24]. It's good practice to prepare scenarios of interaction for testers to compare similar situations.

During the test can the *Think aloud* technique be used to retrieve information from the participants [25]. By encouraging the participants to speak out every thought including something they don't understand or have an opinion about is a robust and cheap way of finding issues. Robust in the sense that you don't have to strictly follow the methodology to receive good results and cheap in the way that you don't need any special equipment or location to perform it. At the same time are there some drawbacks to be noted with the technique. For example can it feel unnatural and obstructive for the testers to vocally explain their thought process and opinions if they aren't used to it. Another problem is filtered statements which means that the tester might withhold opinions to appear clever or only say what's "correct". It's also necessary that the testers have a stake in the project as their words

will in the end shape up the final design of the product for its end users. If the testers think and act differently than the typical end users can the test instead just be a waste of time.
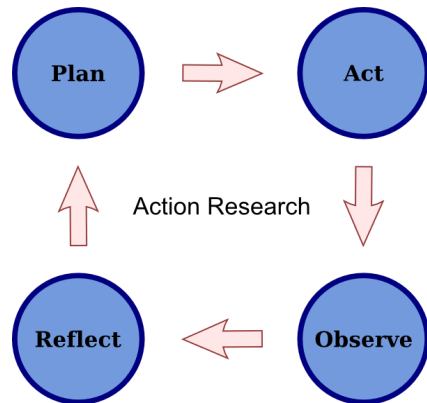
3.2.6 Action research

Figure 3.1 The process of action research

The chosen approach for the research in this paper is "action research" [26]. Action research is simplified described as "learning by doing". The primary step is to identify the problem, come up with a solution for it, see if the expectations were reached or otherwise start over. In comparison with other research techniques is the focus here more pinpointed towards an acceptable result. In a normal research is it often important that the researcher stays objective to create a trustful work. As the investigated concept, the knowledge bank, is in direct contact and applicable to the real world, it seemed as logical choice.

3.3 Problem identification methods

3.3.1 Meetings

The first step was the problem identification phase where meetings were one of the methods practised. During the whole research have many informal meetings occurred which has provided useful information for the further process. Every design choice and proposition with solutions has been debated with the supervisor and other managers at Nilex for an agreement that the possible result is of value. The supervisor, as the head of development at Nilex, has years of experience and is almost in daily contact with customers which gives an understanding for what works and what doesn't. The first meetings were the main discussion about his perspective of the knowledge bank and how he perceived the customers feelings towards it. Lunch meetings also occurred with sales managers that are even more in touch with current and potential customers. These meetings weren't documented in any formal way except sometimes as some scribbled down keywords on a paper to be remembered later. The discussions behind these keywords were investigated in later methods with personal contact with customers.

The meetings were very important as a stage to understand the perspective from the managers point of view about the knowledge bank but it wasn't sufficient as a base for the whole study.

3.3.2 Self-studies

The task of self-studies was done doing literature studies together with investigating Nilex current knowledge bank and their competitors [27,28,29,30]. Primary source for finding information to the literature study was the "Google schoolar" site [31], a website designed for finding research papers from all around the Internet, and LUBsearch [32], Lund Universities service for finding ebooks, papers and other information documents. Search phrases were keywords thought to be related to the research. This were words like:

- "Knowledge bank"
- "Knowledge base"
- "Knowledge management"
- "IT services"
- "Tagging"
- "Information tagging"
- "ITIL"
- "Interaction design"

Links to interesting information was saved for further studies. The ordinary google search engine was also used for findings articles that wasn't published in a scientific research. In those cases were extra care put towards making sure that the cited sources and authors for the article was reliable. Books and eBooks were recommendations by the supervisors together with previous knowledge of what could benefit the research were essential for the literature study. Ebooks was borrowed or bought from the Internet while the books were found in the researcher's bookshelf. To be noted is that many times, as new terms and concepts were unfolded, was this step returned to.

Access was also given to a sandbox environment of the current Nilex software for investigation about the current situation. This made it possible to edit and change options related to the knowledge bank and other modules. The description of the system from this investigation is described in the examination section below. These studies were essential for further methods and results.

Even if no real implementation was made for this research was it beneficial to understand the programming model and languages that are used for developing to make the design somewhat plausible. Primary, as ASP.NET MVC is a Microsoft framework was Microsoft's developer portal "Microsoft Virtual Academy" proposed for guidance. There were two online

courses taken called "Developing in HTML5 with JavaScript and CSS3 Jump Start" and "Developing ASP.NET MVC 4 Web Applications Jump Start". The premise of the first course was to give an understanding of modern web technologies. HTML5 for example introduces many new elements like <nav>, which surrounds navigation links, for its markup. Cascading style sheets, CSS, is a language that describes the layout and looks for a markup language such as HTML5. Javascript is among other things a language to define the behaviour on a website. The second course explained how to use these three web technologies within a Microsoft framework that often has C# as the language for the server and database logic. The learning material was foremost videos held by Microsoft employees with expertise in their respective fields. After finishing each session was a test was performed on the site to make sure the information was understood.

Additional web courses were taken to further improve the understanding of the source code. This time from a website called "Pluralsight" that provides courses in thousands of subjects intended for developers and IT-enthusiasts. Here was knowledge received about Nhibernate, a functionality of translating programming objects to database rows and vice versa, and KendoUI, the HTML5 framework NSP uses for its implementation.

Examination of the current Nilex Knowledge bank

The following section is an investigation on what the current knowledge bank consist of, both in terms of functionality and layout. This was a part of the early self-study to receive an understanding of the current knowledge bank. First in this section are the steps of finding and displaying articles from the knowledge bank followed by the steps of creating an article. The section ends with a description of the connection between an incident and the knowledge bank.
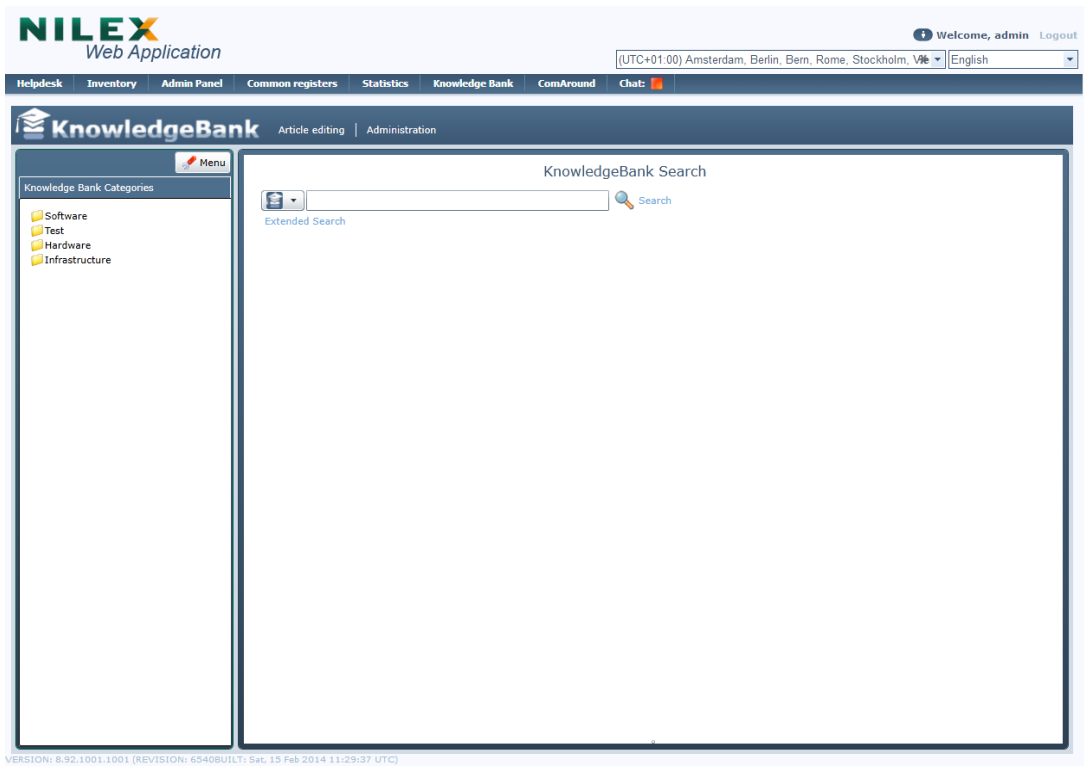
General overview

Figure 3.2 Home page of the knowledge bank

Nilex current knowledge bank is built like a typical forum where articles are placed within categories and sub categories. Visually it shares similarities with the folder structure for the file manager in the Windows operating system (see figure 3.2).
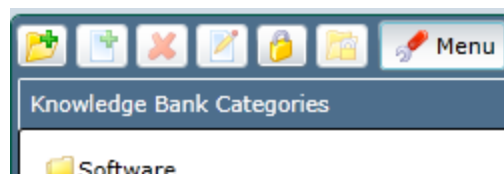

Figure 3.3 Menu in the knowledge bank

Above the folders is there a menu button which opens up more buttons for users with administrative rights. They can from there add or remove category folders as well as the content stored inside them (see figure 3.3).

Figure 3.4 Search source

The search bar has a central role in regards of both placement and size for high discoverability. By pressing the button with a drop down icon, to the left of the search bar, can the user choose from what source the search should be performed at (see figure 3.4).
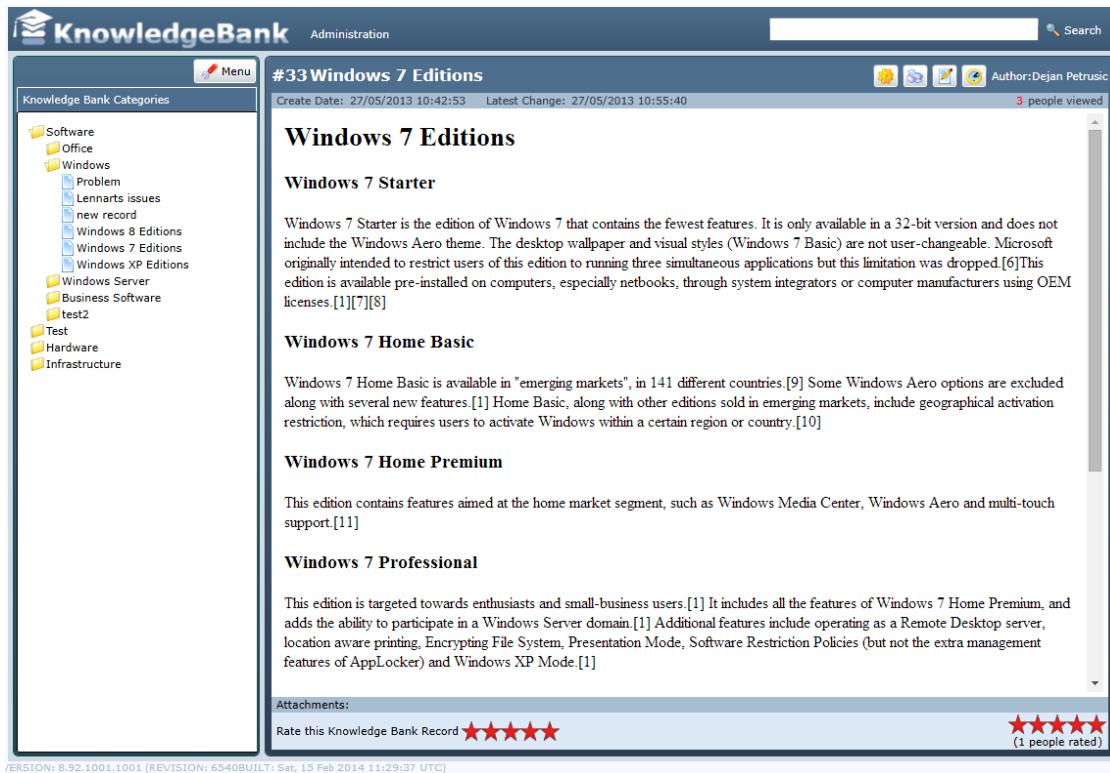
Reading articles

Figure 3.5 Inside an article

When reading an article is the main panel's content, where the search bar was situated, replaced with article information. The layout structure stays the same but the search bar is moved to the title header to always be visible and useable. An article has a title and formatted text to facilitate an easy reading experience. Besides the article are there information and statistics about when the article was created or updated and how many people have viewed it. Below the article is it information regarding attachment and two rating scales with stars. The leftmost one is for the current user to give a score by picking between one to five stars and the right one combines the scores for all the users (see figure 3.5).

23

Figure 3.6 Options inside the article

There are four buttons alongside the text describing the author of the article in the topright corner (see figure 3.6). The first one (1) is a way to notify the author that the article needs to be updated. After pressing it a popup appears where the user has a chance of describing what is wrong. The notification then appears under the "Article editing" tab in a dropdown list for the author to handle. The following button (2) is for a printer friendly version of the article in a popup window. The third button (3) is enabled for users with administrative rights to edit a published article. The button will open the same view as for creating a knowledge bank article. The last button (4) will show a grid layout with the history of the article. The user can from that grid view old versions of the article to see what has changed.

Finding an article

Figure 3.7 Search result in the knowledge bank

After performing a search the result will be displayed as a list of articles (see figure 3.7). Each list item includes of a title together with a short summary of the first words inside the article.

The rating together with how many votes are also displayed as an indication to how relevant the article is to the search. With a normal search will only articles with words that are an exact match with the search term included. Those words are also highlighted inside the summary text. To be more specific from where the result should be taken from can the user select a category in the side panel before performing the search.
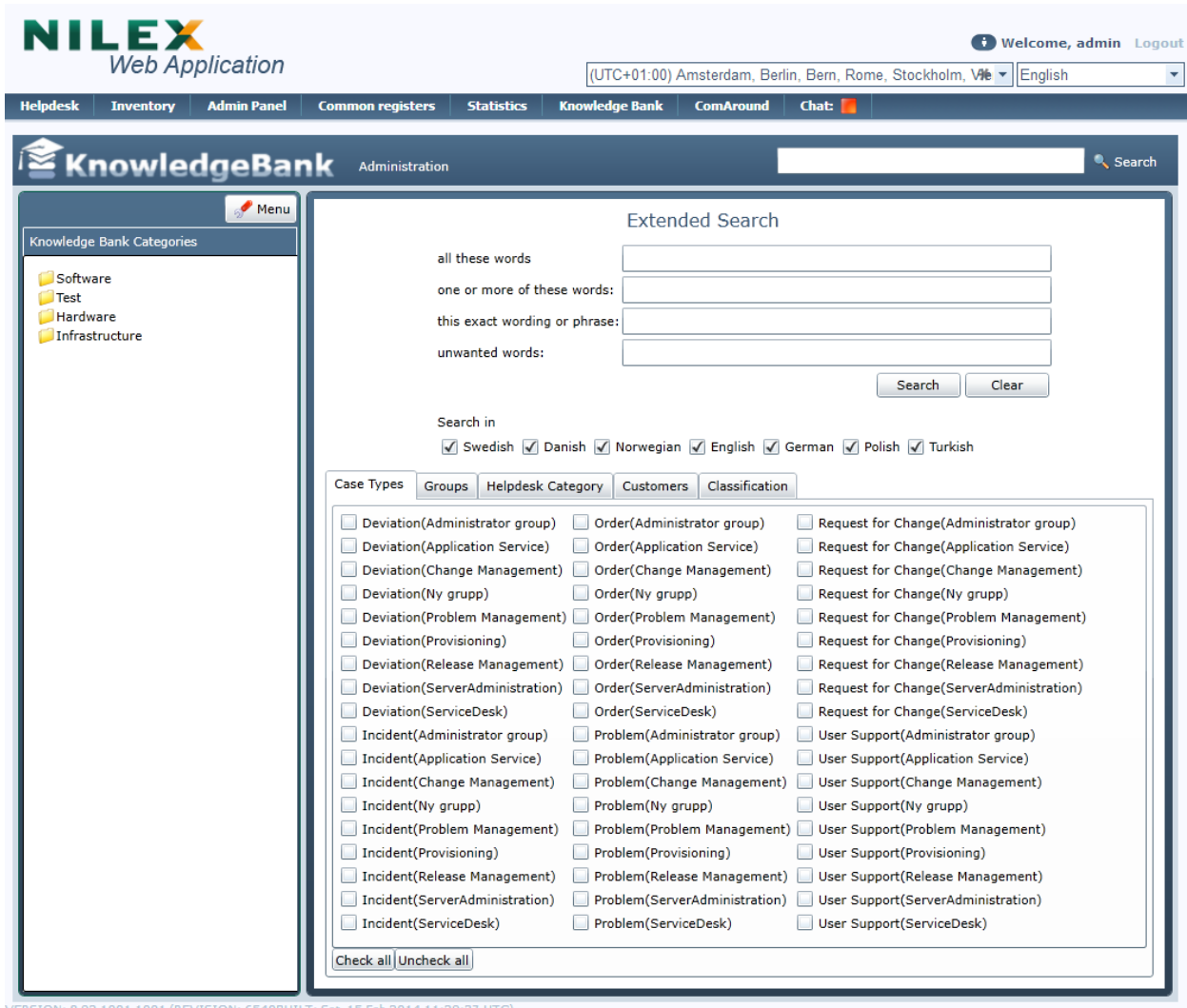
Figure 3.8 Search options in the knowledge bank

There is a possibility to be very specific in a search by using the "extended search" menu (see figure 3.8). Parameters such as "exact phrase", "must include several words", "disregard articles containing a particular word" and choosing what language the article is written in are fields for a more narrow selection. The user can also sort the articles depending on if they know what kind of case type was linked to it on creation. In the same tab menu is there even more choices for example selecting user groups and helpdesk categories, with last one being the same as selecting it in the left panel.

Article creation
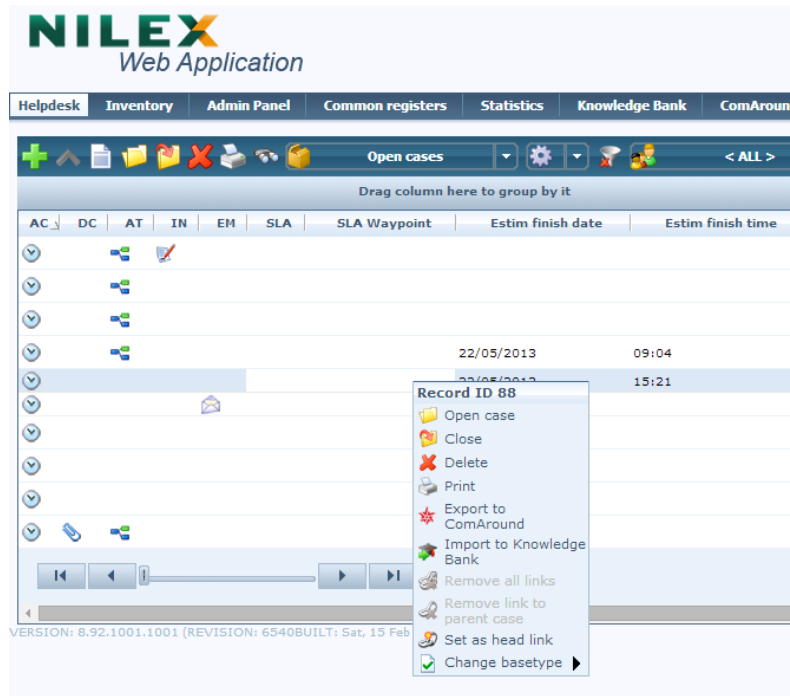
Figure 3.9 Create article from ticket overview

There are two different approaches of initiating the task of creating an article. The first way is by right clicking a ticket inside the helpdesk view and choose import to knowledge bank (see figure 3.9).
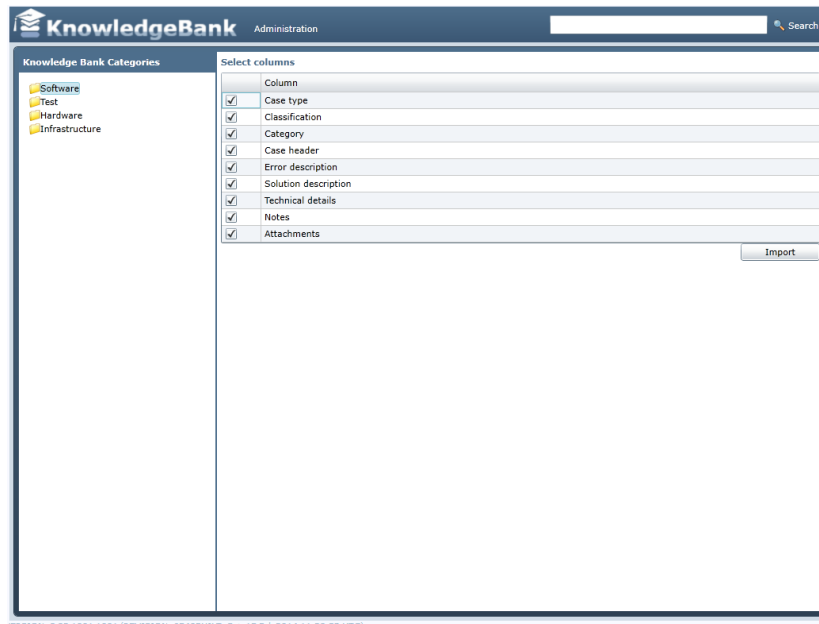
Figure 3.10 Include data from the ticket

The agent has the option of deciding what data to be imported by selecting corresponding checkboxes as well as picking a knowledge bank category by selecting it in the left side panel (see figure 3.10). That information will carry over to the "create article view" as pre filled fields and linked data. The other way is by creating an article from scratch, which is done through the knowledge bank section in the menu button. In either way will the user eventually end up at the "create article view".
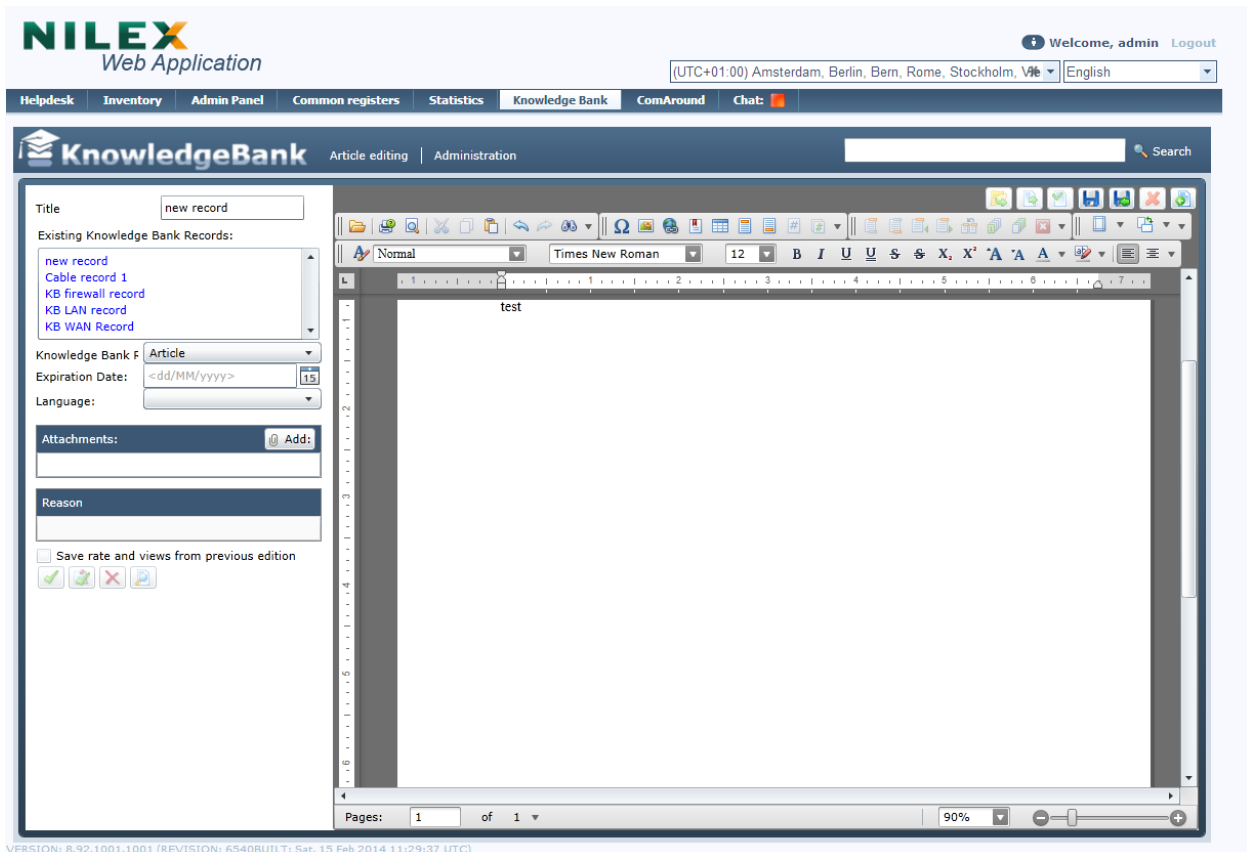
Figure 3.11 Creating a new article view

The resemblance of this view is very much in common with all the ordinary word processors (see figure 3.11). There are all kinds of tools at the disposal for the editor to create an article with a pleasant reading experience as possible .
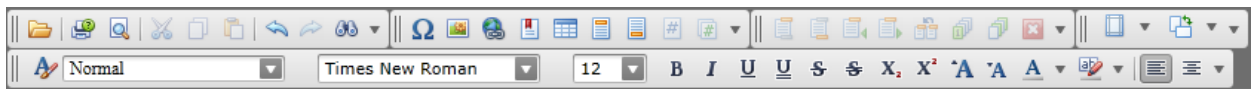
Figure 3.12 Toolbar for creating a new article view

Formatting text by changing font, size, color and other common manipulation functionality. Some tools are greyed out and none-selectable but activates once another tool has been used. An example of this is the button for inserting a table. After picking the values for the

27

numbers of rows and columns can the user thereafter add or delete as the table toolbar group is now active. Each toolbar group can be moved, added and removed all according to the article creator preferences. To be noted though is that all toolbar groups are visible as standard.

Figure 3.13 Left option panel for creating a new article

In the left panel the creator specifies the title visible when searching and browsing for articles (see figure 3.13). While typing out the title are existing articles presented in a list just below as an indicator on what has been made in that regard earlier. Further down is a dropdown list for selecting what kind of knowledge bank record it is, in this case article, and another one for selecting the language. By selecting an expiration date will the article's status change to obsolete when a certain date has passed. The "Add attachment" field is for the option to upload files stored locally on the computer as addition to the written information in the article. Reason is a field to specify why an article was edited in the first place.
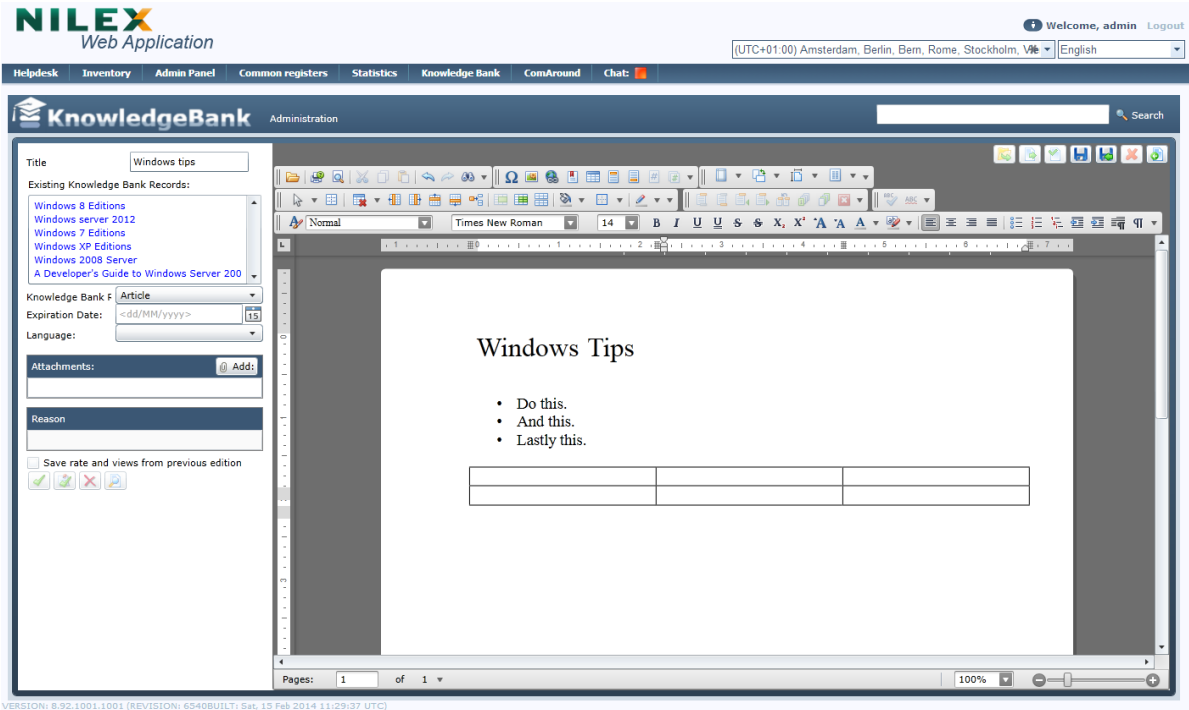
Figure 3.14 An article after some editing.

After the article creator is pleased with the outcome of the article it is time to go through the steps of publishing.

Figure 3.15 Save the article

An article must always as a first step be saved as a draft. That can either be done by the "save" button (floppy disk symbol) or "save and close" button (floppy disk with green arrow symbol) (see figure 3.15).

Figure 3.16 Publish the article

If the normal save is performed will the view will stay the same but the ready for publish button (document with a green checkmark symbol) will now be enabled (see figure 3.16). Otherwise with the "save and close" button will the user return to the previous page before starting editing.

Figure 3.17 Notification about a draft

Thereafter must the user hover the article editing tab to see that there is an notification telling that there is an article in the draft-collection, ready to be handled (see figure 3.17). Once inside the editing view again will the user see that the "ready for publish"-button is active.

Figure 3.18 Notification about a pending validation

After pressing the "ready for publish" button will the notification status change to "Pending validation" and new options will be available in the left panel (see figure 3.18).
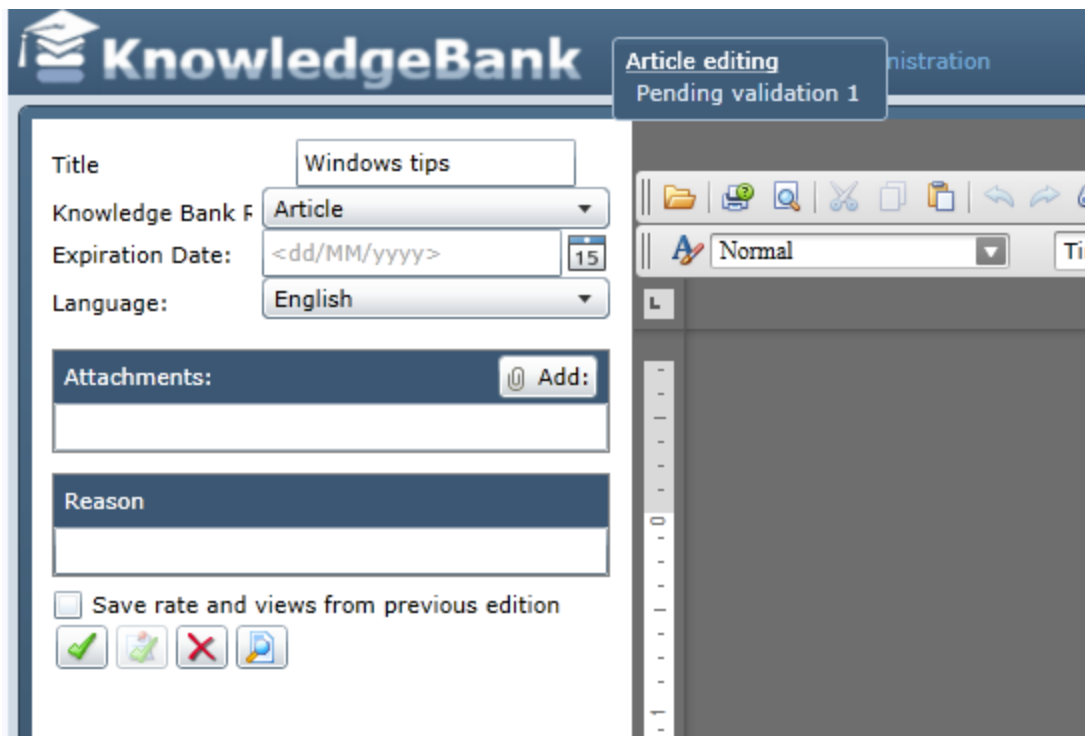
Figure 3.19 Left panel with publishing options

In this example is the user logged in as an administrator and can therefore press the acceptance button to publish the article as an available record in the knowledge bank. Otherwise, depending on rights, may the article undergo a technical review by someone experienced in that segment.

Figure 3.20 Notification about a technical review

By pressing the "technical review" button (paper with a magnifying glass symbol) will a user from the group of technical reviewers receive a notification indicating that there is an article ready to be reviewed (see figure 3.20). After the review, where the reviewer can either press "Tech Accept" for the article or tell a reason why it needs to be updated and decline it, will the article creator receive the reviewed article. If it was accepted, then it's time to publish it (button with a green check mark) or otherwise update what is incorrect and send for review once again.

Ticket perspective

The connection between the helpdesk and the knowledge bank is an important feature for the Nilex service management. It serves as a purpose of both enrich a case with linkage to a knowledge bank article and the functionality of importing a ticket as a base for creating an article. At the other end can ticket types be used as parameter of finding knowledge bank articles if it has been associated with a such entity.
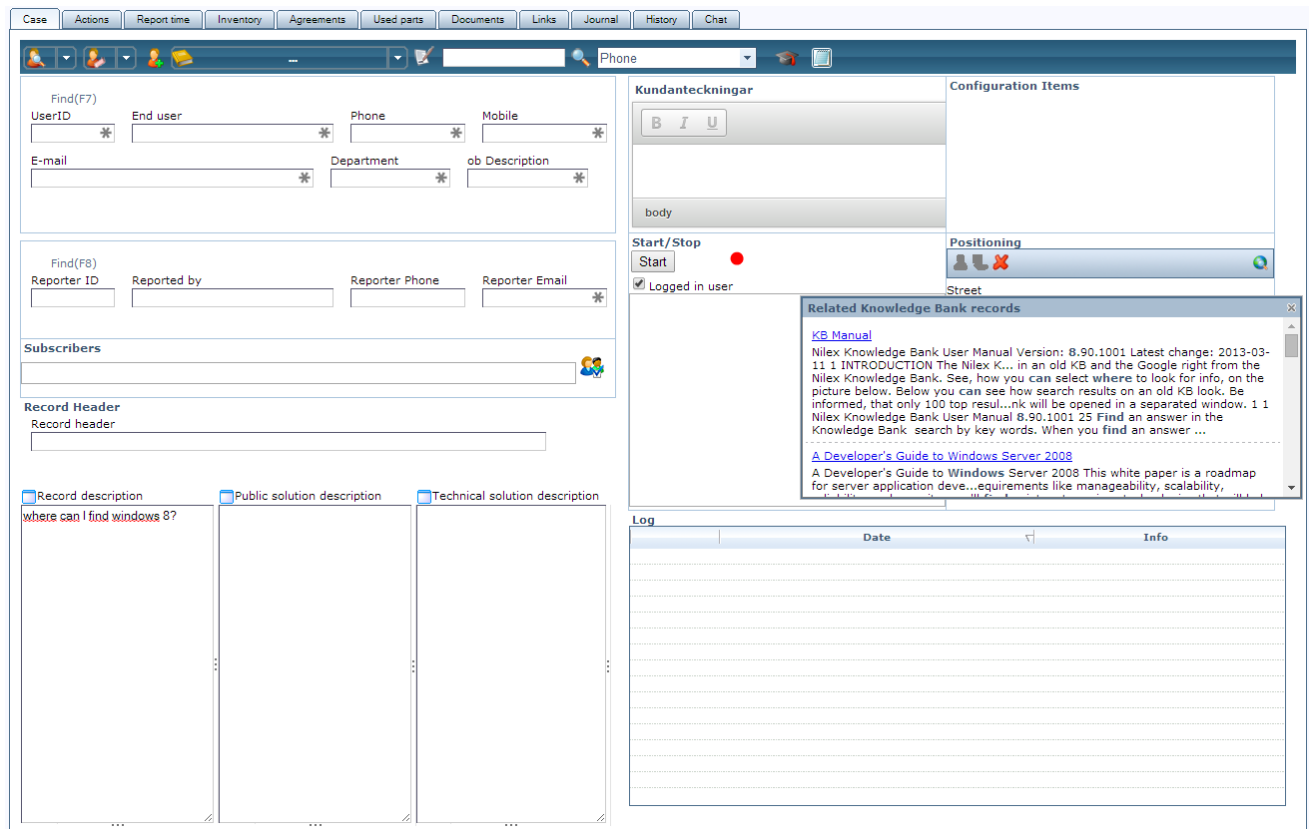
Figure 3.21 Knowledge bank search from within a ticket

When an agent is editing a ticket by typing information in the one of the three description fields will the system performs a knowledge bank search for those words in a popup (see figure 3.21). Clicking one of these links will take the agent to that particular knowledge bank article.

Figure 3.22 Linking a knowledge bank article to a ticket

Linking an written article to an ticket is done by clicking the the icon with a graduation hat inside the incident (see figure 3.22). That will open up the knowledge bank in a new window to perform the selection.
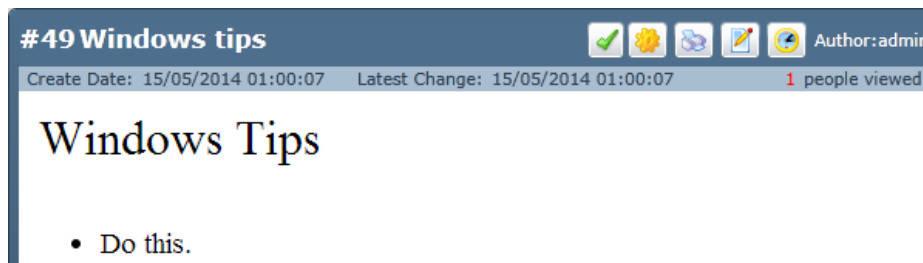
Figure 3.23 Option to link the article

From that new window is the task to navigate and click an article and inside there press the "check mark" button for associating it with that case (see figure 3.23). All associated articles are displayed in a list inside the case.

3.3.2.1 Discussion

The primary problem is that it's pretty hard to know if an article is of interest when the result is listed. The user sees the title but that is often not enough to evaluate if you should click it. The rating is an indicator that you can use but the sorting algorithms seems wrong as sometimes lower ranked articles did appear higher up in the list. The user can use the extended search but there are so many options that it can appear overwhelming. The connection to the helpdesk is very important but the functionality and oddly button placement makes it seem that it was somewhat rushed to be there. It also seems unnecessary that you have to save a draft of the article before you can send it for technical review or publishing.

3.3.3 Interview

Performing an interview is vital to understand a problem beyond what the earlier mentioned methods can provide. This was the first interaction with a customer so the intention was to see if aspects learned could be confirmed. Contact information was given from my supervisor to a customer that is deeply engaged in the software Nilex provides. She is the manager of the Nilex software at her company with hundreds of employees. Understandably she is very keen on that the software meets the expectations to provide the value agreed upon when invested in Nilex. The agreement also includes listening to possible improvements and implementing those. The interview was done with a semi-structure approach where some general questions were prepared prior to the meeting and follow up questions were asked depending on the answers received. Besides the questions was an early draft of the survey shown to provide feedback if the questions were understandable and reasonable. It was performed at her workplace in a meeting room.

Summary of the interview

● How are you and your company using the knowledge bank today?

The immediate answer to that question was not at all so I was curious why that was the case. One mentioned reason was that all their problems today were unique so the knowledge bank would serve no real purpose. She also explained that the organisation in reality lacks the structure for upholding the knowledge bank, there needs to be documents and other information to help and guide the users on how to use a knowledge bank. As the company exits in several countries this means that the documents also needs to be translated and approved by each division which takes time and planning. But she could see the benefit of a well managed knowledge bank so the strive was towards it.

● Is it a reasonable workload to create knowledge bank articles?

She explained that it could be made easier but had no direct solution she could give. But one problem she was experiencing during creation of articles was that it was difficult for the technical reviewer to acknowledge that an article was ready for review. The reviewer has to first login and thereafter hover over the notification bar to be notified. A simple solution as a mail notification when an article is ready for the review process would suffice.

● Is it easy to find correct articles when performing a search?

She said that some words generate many articles and some doesn't generate any at all. An important feature is the system of rights. If that is set up properly, with different roles being able to see and read the intended content for that group, much is already won in the quest of making articles easy to find.

- What do you think of enhancing the KB-experince with tags?

It can work but it must be easy to manage and not a painful extra step.

During the presentation of the draft of the survey did she once again mention that it's meaningless to notify an author that won't see the notification. She also questioned how many customers in reality are using the knowledge bank.

Another comment was that she wanted to see the case number as an agent while browsing the knowledge bank.

3.3.3.1 Discussion

Many issues she mentions are connected to the management which isn't something this research will investigate further but it's important to note. Though can a KB with good interaction design ease the burden of setting up the management if no extensive training or learning is required for controlling it efficiently. A better notification functionality for administrative users that isn't logged in is important to keep the KB updated. The tagging functionality seems like she is cautiously positive towards but it must be made correctly. The next step is therefore to investigate how and where the tags should be implemented. Except to make some small clarification on a few questions did the questionnaire seem reasonable and ready to be sent to other customers.

3.3.4 Questionnaire

Nilex customers are spread all over Europe so a viable solution to receive feedback about their general opinion was done through online surveys. The questions were tested in the interview phase to make sure that were no misunderstandings. The goal was to get general knowledge on how they are using the knowledge bank and what improvements they would like to see. Those that answered are mainly agents in the service desk which means they are the ones operating the knowledge bank. See appendix A for the overview of the questionnaire.

Result of questionnaire

The combined answers from the 19 participants gave the following graphs:

What is the size of your organisation?
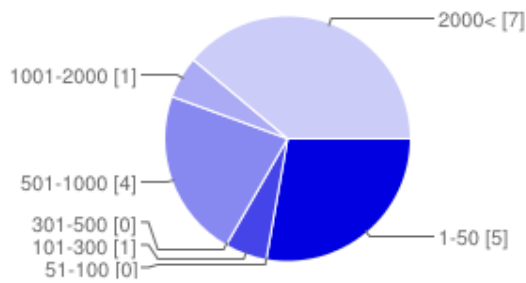
1-50	5	28 %
51-100	0	0 %
101-300	1	6 %
301-500	0	0 %
501-1000	4	22 %
1001-2000	1	6 %
2000<	7	39 %

How often do you use a knowledge bank?
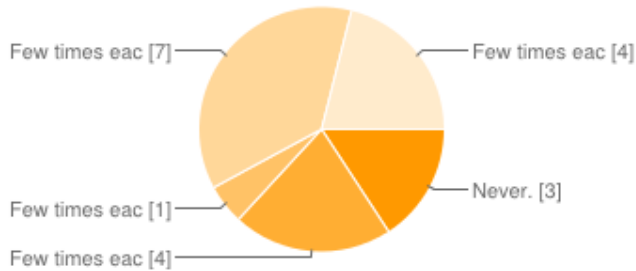
Never.	3	16 %
Few times each year.	4	21 %
Few times each month.	1	5 %
Few times each week.	7	37 %
Few times each day.	4	21 %

How often do you rate an article in a knowledge bank?
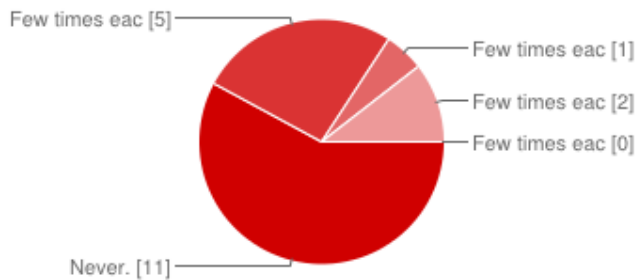
Never.	11	58 %
Few times each year.	5	26 %
Few times each month.	1	5 %
Few times each week.	2	11 %
Few times each day.	0	0 %

Figure 3.24 Answers to the questionnaire

Do you primarily create knowledge bank articles from user-submitted incidents or from scratch?

I don't create articles.	9	50 %
User-submitted incident.	6	33 %
From scratch.	3	17 %

If you have a common problem or question, do you foremost turn to a written source or a knowledgeable person for answer?

Written source.	11	58 %
Knowledgeable person.	8	42 %

If you have a more technically complicated problem or question, do you foremost turn to a written source or a knowledgeable person for answer?

Written source.	8	42 %
Knowledgeable person.	11	58 %
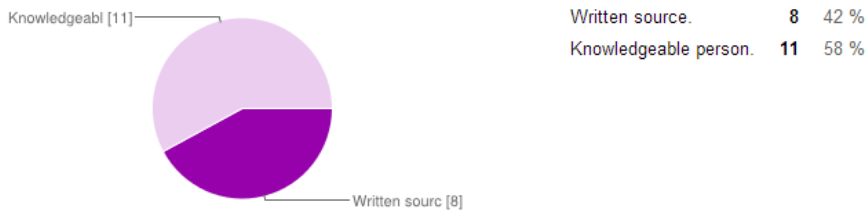
Figure 3.25 Answers to the questionnaire

The last question was "What do you think would make your company and/or your customers to use a knowledge bank more frequently?" and gave the following summary of answers.

- "Good search functionality."
- "Examples on how to use it."
- "Content tags."
- "Short loading times."
- "Easy to use i.e no signup process or accepting terms for reaching the knowledge bank."
- "User submitted comments (If they are moderated / rated after quality)."
- "Need to have more content to fill it with."
- "Superfast and accurate search options with ultra quick preview/ intelligent storting[!sic] etc... Display as you type with automated filtering suggestions as you type etc. etc. quite simply magically give me what I want directly or tell me you don't have it.. Don't make me look through loads of info in a time consuming manner."
- "The more information the knowledge bank includes the more people will use it."

- "User friendly screen, easy to search and find answers. Possible suggestions could make it more interesting."
- "Information and a structured base for the knowledge bank. Build it up with a structure that makes it easy to find answer when searching."
- "A change in how users and company management in my company look upon the possibilities. A change in average user age, ie lower user average user age - younger users are more prone to use self-help. A change in how my companys management staff look upon self-help."
- "Routines for documentation would be good."
- "Have it accurate every time you use knowledge, this means as an Agent and as an normal end-user you should be able to request and submit modifications. Be a part of it! Evolution is more valuable than perfection; in other words: perfect knowledge is out-dated knowledge. Ensure that the content fits the company's need. If someone searches your knowledges base 2 to max. of 3 times without getting proper Content, they will never use your km base again."

3.3.4.1 Discussion

The general conclusion from the answers is that the KB is used to some degree but it can be improved upon. Just like the answers from the interview is the management and documentation surrounding the usage of a KB one important aspect. Even if the questionnaire didn't explicitly mentioned tags did in fact some users mention it as a functionality that can improve the experience. This can be seen as an indication that it's reasonable to investigate the value of tags for a KB. It's absolutely necessary that there aren't too many steps of searching and navigating to receive the desired information. This is hopefully a problem tags can remove or at least somewhat diminish.

3.3.5 Brainstorming

With all the information regarding the old version it was now time to transform the knowledge to ideas. As the main purpose of a knowledge bank is to find relevant information was the brainstorming technique used as a next step for coming up with ideas. The concept was to think about what the web and different type of computer software are using for solving similar issues.

One idea was the usage of a virtual agent to help the users find KB articles. A virtual agent is an interactive chat bot that is represented by a visual humanoid figure. Its purpose is to make the interaction more natural and mimic a real world conversation. This idea was after investigation crossed out as that too many parameters could go wrong with a virtual agent. The time scope for creating a virtual agent that would be feasible to test and not end up in the uncanny valley [33], a middle ground place of visual representation that makes the interaction

feel uneasy for the user, where the biggest issues. Instead was idea of tags elaborated further on which in the end was essential for the whole research.

Result of Brainstorming

Pre-studies had shown that "search and find" is a vital concept for a knowledge bank and therefore must there be parameters to differentiate the KB articles. A search bar is an essential object in a knowledge bank. The search bar must have proper space and location to avoid the paradoxical situation of searching for a search bar and accommodate the Fitts's law.

When a user has typed a search phrase and submitted can the sorting algorithm for results be chosen in several ways. The most "unintelligent" way is just to build a un-sorted list of articles which somewhere contains the exact same search phrase. This solution is only acceptable in the most trivial and very small knowledge banks as the user must follow-up with a time consuming manual search for the information they seek. A better solution is having functionality of rating an article and then sort the result by that ranking. Ranking can be done in several different ways with different parameters to consider:

-Is the article well written?

-Is the article answering the question the user had in mind when typed the search phrase?

-Is the overall assessment of the article higher than in comparison with other articles?

These three items works independently of each other so the user must be fully aware of on what basis the ranking is supposed to work on. For example a common rating system of valuing things from zero to five can be totally insipid if the users answers different questions. The question must therefore be clear and thought out so the ranking means something.

Even the search result after combining ranking and search phrase can be misinterpreted. If an article just contains the search phrase but the overall content is about something else and at the same time has a high ranking, the article will be among the first results even though it's irrelevant. That problem can be solved by "tagging" an article. A tag is a linked word to the article describing the content. An article can have many tags to be more precise of the content. So by searching for a tag combined with a correct implemented ranking system, the user will achieve a better result and overall experience of the searching system.

3.4 Design process
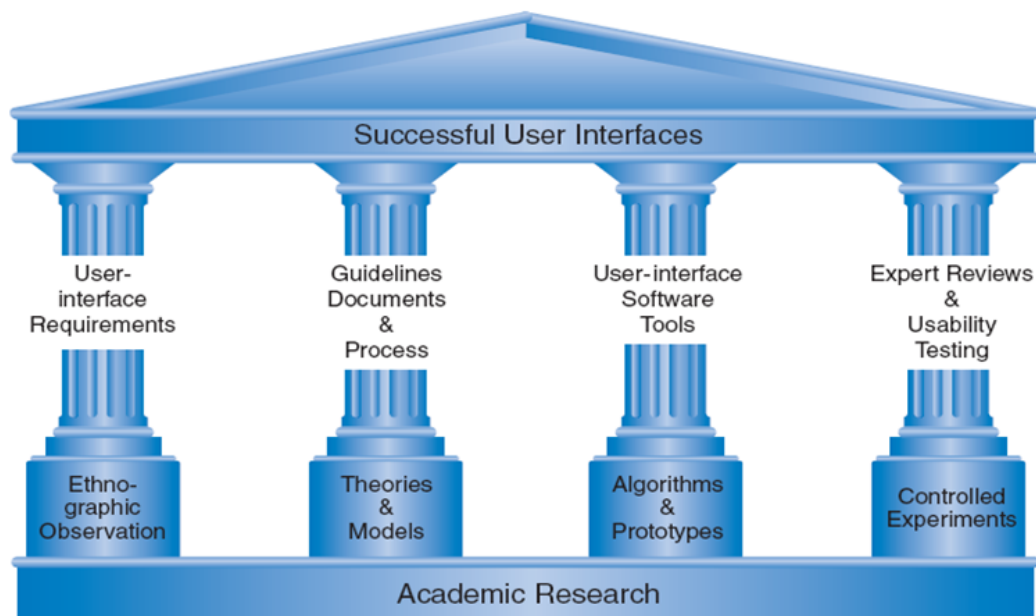
Figure 3.26 The four pillars of design [34]

A way of turning ideas to great innovations can be done by the four pillars of design described in the "Designing the user interface" (see figure 3.26). It's not proven to work without any complication but it has improved the work process and speed of those that have tried to follow it. After the initial work of uncovering the problems of why the knowledge bank isn't used to its full potential, it was now time to propose solutions. As the four pillars has been a factor in successful environments earlier, the idea was to try follow to its principles.

3.4.1 Requirements

As the work included both functional and visual improvements for the knowledge bank were both these factors in mind during the creation of requirements. All the data collected from each of the problem identification phases of meetings, self-studies, interview and surveys were parts of the resulting requirement list. Functionality users from the old knowledge bank have gotten used to must be included in the new design to sustain familiarity but at the same time improve the user experience of it. Some of the requirements were also there to fulfill demands of ITIL software. No attention was given to performance issue as it wasn't plausible or interesting for this work. The requirement list ended up with more requirements than what was included in the testing prototype as the work progressed towards investigating specific the "tagging" functionality. See appendix B for the full list of requirements.

3.4.2 Prototypes with guidelines

The next step in the design process was creating prototypes for the requirements. The approach was to combine the two center pillars, guidelines documents and user-interface software tools, to make it easier to understand the reasoning behind the decisions and what they try to solve. Prototypes can be made in several formats depending on how far the progress has gone and what the purpose is. In this case was the focus set on creating low-fidelity prototypes as this was still an early stage in the project. Low-fidelity, lo-fi, prototyping is a great way of creating a draft proposal of the design without the need of a finished system for the users to still get a comprehensive clarity [35]. In practice is all you'll need a pen and a paper but a specialised program for creating the design or mockup called "Balsamiq Mockups" [36] was used.

The mockups were created as a type of storyboards, a famous concept from the movie industry, as a way of visually explain what a layout or interaction means or leads to by combining images and text [37]. The storyboards were made as a web browser interface where the proposed design was defined within a web page. With the NSP project being a web solution, this approach seemed to benefit the parties that would examine the proposal in the future stages. The reason being that the design would hopefully come closer to what the final product may look and behave like. Outside the border of the web browser was information boxes put down for each of the storyboard pages. In these boxes was information regarding design choices and interaction behaviour described with arrows pointing towards the subjects described. The purpose was that the information should be interpreted as a sort of guideline of the system with a direct connection to the content being described.

The key point of the design was to include the tagging functionality and the parts that will collaborate with it. As this knowledge bank is more a closed software with higher demands on the administrative role of controlling the interaction, in contrast to a social website, the functionality of tags must also be administered. The idea was therefore that user may not have the possibility to freely tag articles to their liking but instead is that handled by an administrative role. With this concept could possibly many of the problems associated with tags vanish.

Categories already exists in the overall NSP system and that can hopefully be taken further advantage of. The idea was therefore to design a tagging system in the same way as Wikipedia, namely the Thesaurus system. With the combination of tags and categories can for example an knowledge bank article be linked to another article from a subject of related articles which is one of the benefits of thesaurus.

The storyboards were discussed in consultation with the supervisor at Nilex to make sure the design was going the right path. Subjects that were needed to be changed or clarified lead to updates to the prototypes.

First iteration of design
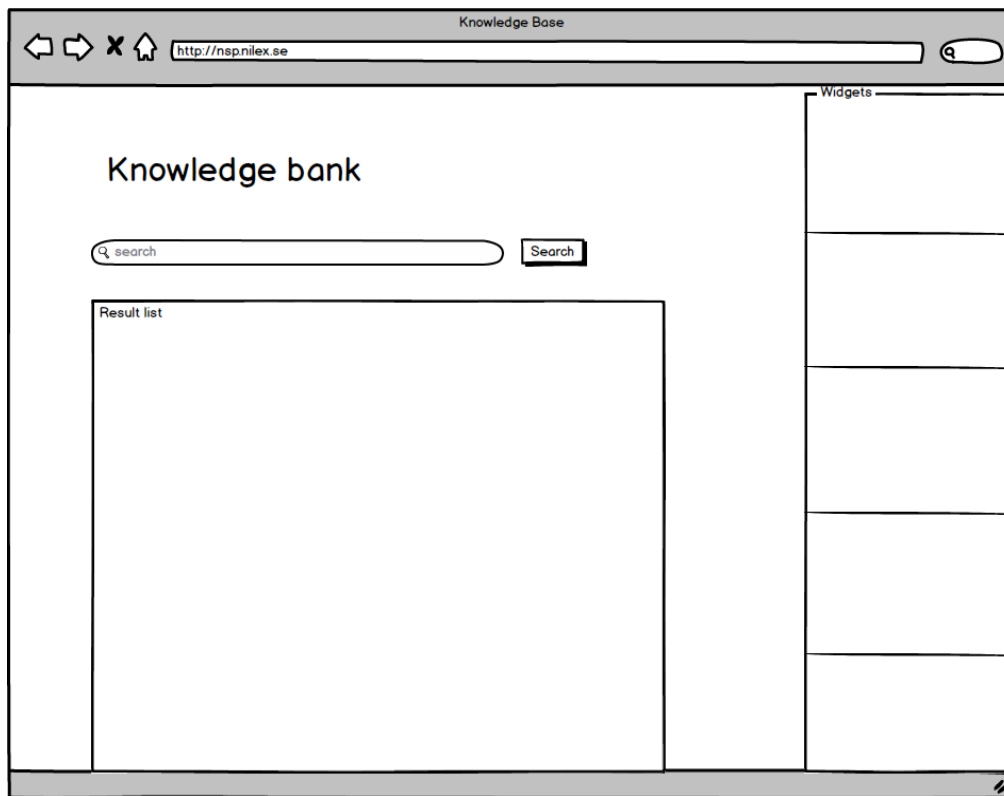
Figure 3.27 Initial mockup

The first step was to design what areas and spaces that should be available and with that in mind set up the general structure (see figure 3.27). The prominent title of "Knowledge bank" has a central role in the left corner with generous margins to avoid uncertainties what page it is. Just below the title there is a search field with a search button to perform the search.

The widget row to the right was an early idea for improving the usage and willingness of usage of the knowledge bank. A widget is a convenient information source for a quick glance to see what can be of interest for the user.

Biggest space was given to the frame just underneath the search field, where most of the interesting information will be displayed. The information inside this frame will be updated depending on what links and buttons that has been clicked. The premise is that the surrounding objects of this frame will stay the same and only the information inside the frame will change. This means that user never will be insecure in the navigation as the primary design stays the same with the search field and widget row intact.
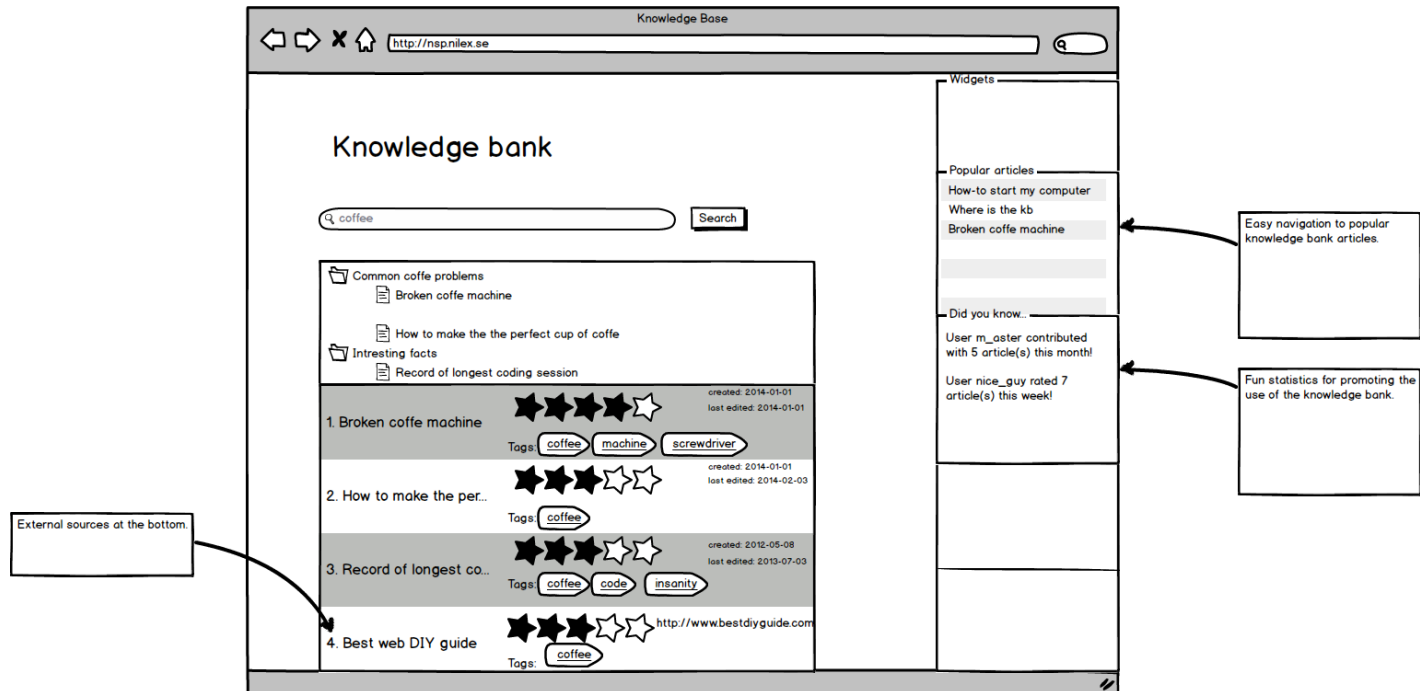
Figure 3.28 Mockup with a search result

The following mockup was created as an indication over a possible result list design (see figure 3.27). In this case has a user searched for the word "coffee". As stated earlier, the main design stays the same but the bottom frame has changed.

In this example is four different articles listed and what categories they belong in. Categories are symbolized by a folder icon while an article is a document icon. These two icons are natural mappings of their representation, both in the context of a computer file system and for the real world.

The sorting of the articles in the list is a result of rating combined with tags, which was the conclusion from the brainstorming. Rating is represented by a "zero to five stars" scale that is a sum of the user votes. The tags associated to an article also have a central role for each displayed list item. They both works as a description over what the article is about and clickable for searching that particular tag. The visual representation was to mimic a packet or can label. The purpose of the last item was to test the possibility of including external sources from outside the source of the internal knowledge bank articles. Two widgets were also proposed. One listing "popular articles" in a recent scope of time and the other showing statics over user behaviour.
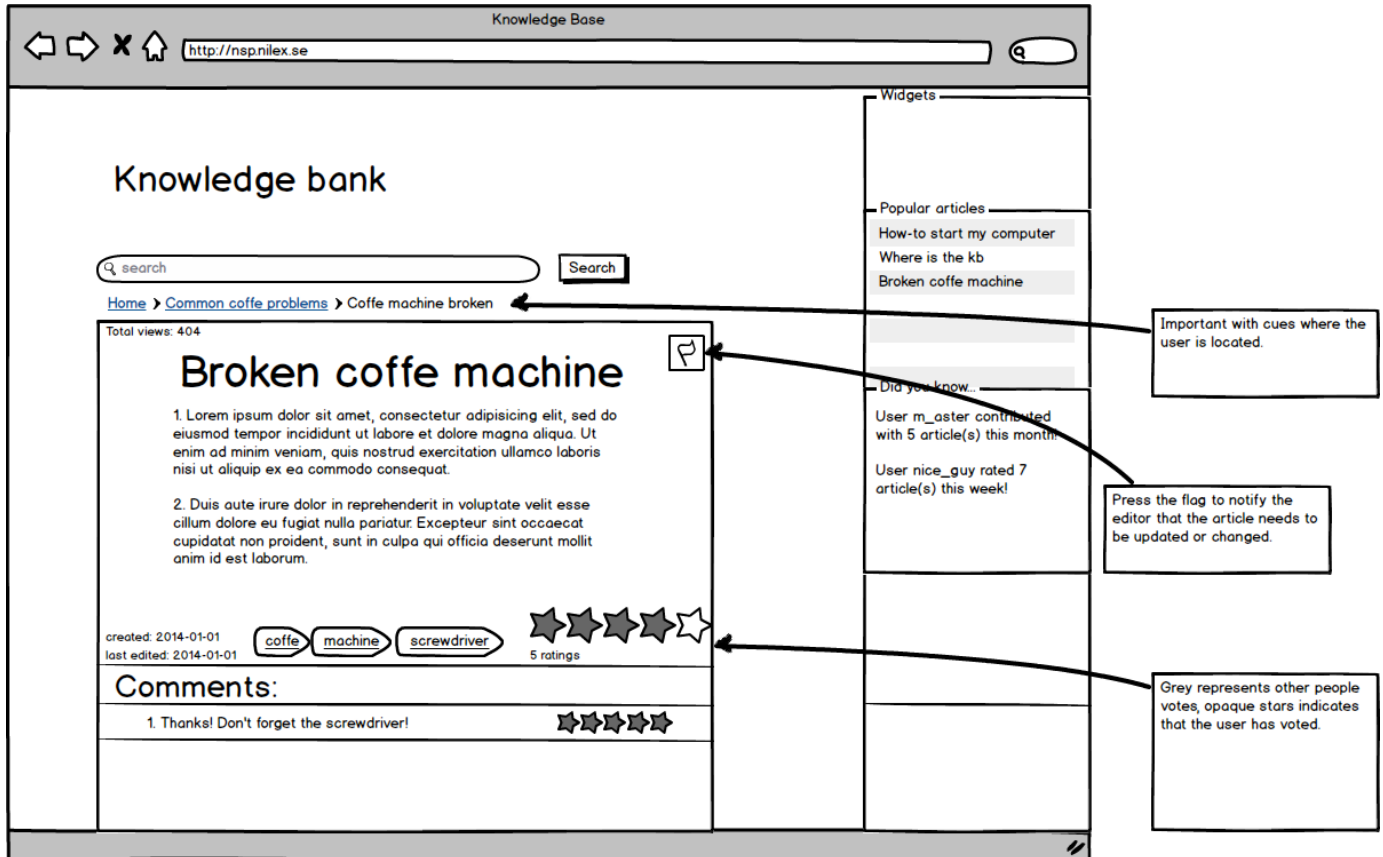
Figure 3.29 Mockup from inside an article

The final of the first three mockups was the design from within an article. For simple navigation to the categories, that the article is placed within was a navigation link bar included above the article. In this view was also the associated tags represented by the same label icons like in the result list.

A new design choice in comparison to the old Nilex knowledge bank was to remove one of the star scales to only have one handling the same functionality. Before the user has voted is the combined average of the previous votes of represented by a transparent grey. After the user has voted will the grey disappear and be replaced by golden stars for the same amount of that the user voted for. The user will therefore not be able to see the total average after voting. The idea was to remove the uncertainty of which star scale the user should use for rating and make the interface cleaner.

Included as well was the functionality of rating comments that users contributed to the article by a star scale. Naturally would a higher score mean that the comment consists of vital information in addition to the article. A button with a flag in the top right corner of the article indicates a functionality of notifying the creator of the article that something needs to be changed.

Presenting these mockups to the supervisor, the head of development, and the product manager of Nilex started a discussion of further work. Two primary points were brought up during these discussions:

- Include design from the agent's perspective inside the helpdesk with the usage of tags.
- Make interface objects as tags and rating clearer of their meaning.

People that answered the background survey were foremost agents so the conclusion of improvements is mainly for the environment they are working in. The next step was therefore to create storyboards for the interface where the creation of article and tags are made. As for other interface objects like buttons and links must additional research be performed to find viable solutions. Concepts like color coding and functionality from the old knowledge bank were important subjects to look closer at.

Second iteration

To further make use of the proposed tag taxonomy of thesaurus was the functionality of related articles introduced. Related articles combine similar tags with categories to create the list of items.

One important functionality is to be able to notify the author of the article needs to be updated. The idea was that either can the content be completely wrong or it isn't up to the correct standard and that those two cases needs to be separated. Two different dedicated buttons were therefore proposed for those cases when the article needs to be redone from scratch. The first proposal was mentioned in the first mockup of an article where there was a flag icon for reporting. This was indicated after testing to be somewhat unclear and was changed to a red exclamation mark with a text of "report" under it. Clicking that would create a popup where the user can write the reason for the report.

The rating is an indication for when the standard of the article isn't good enough. The proposed design was therefore that if the user gave it three stars or less would a popup appear where the user could be more specific about the problem.
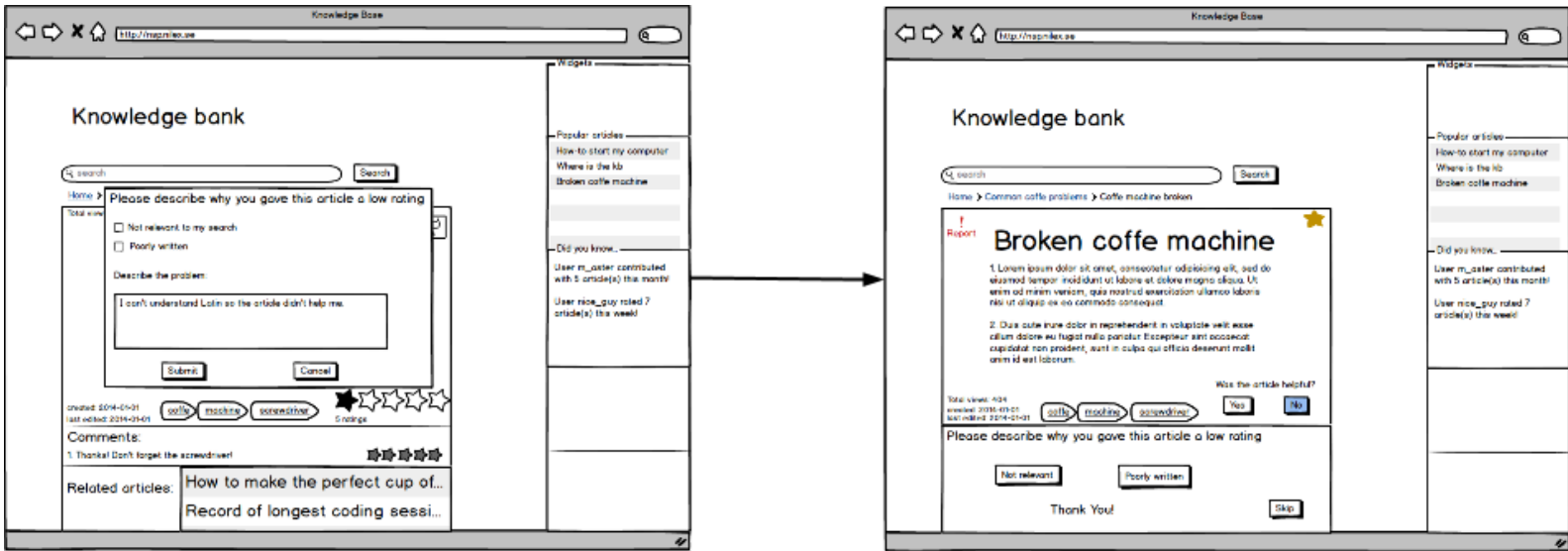
Figure 3.30 From a popup solution to a none popup solution.

Both these report systems had at first a popup window for feedback but this isn't a good modern web solution. Pop-ups are seen less and less nowadays as their appearances are intruding and hard to fit in to the overall design of a site. Mobile phones and tablets doesn't handle pop-ups very good either which made the choice of redesign necessary. The proposed solution was to make the window appear naturally below the article and pushing the existing content downwards. This would make the information to no longer be intruding while keeping the overall design format (see figure 3.30).

The rating system was also a debatable concept which went through some redesigns. The main problem being that it can be hard to declare an opinion of the article by selecting an arbitrary number of stars. For example is it hard to say what differs an article for receiving two or three stars. The final design replaced the star scale with a question with two choices. The asked question could depend on what the managing part of the KB is interested in but an example is if the user found the article helpful with a yes or no alternatives. The star scale is still a good visual representation of the rating so it was kept intact in the result list but on the premise that the percentage sum of all the positive answers was represented by corresponding amount of marked stars.

Create article from incident
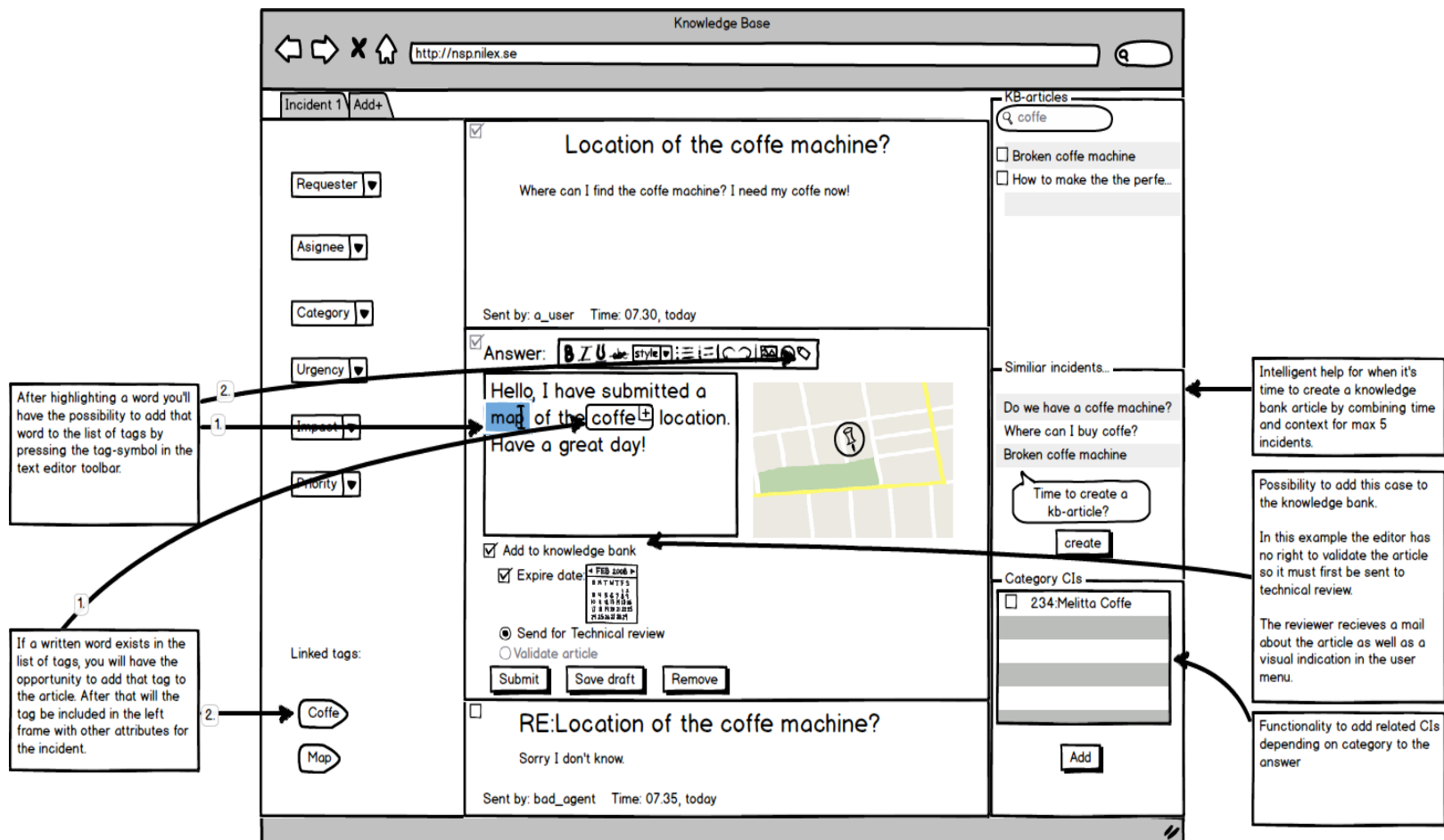
Figure 3.31 Mockup from inside an ticket

From an agents perspective is the interaction when creating an article with tags the main feature to look closer at (see figure 3.31). The idea was that the addition of tags could happen in two different ways depending on if the tag existed in the system previously or not. When the agent writes the answer to the incident could written words that exist in the database of tags be highlighted with an option of adding them to the article. If the agent wants to add a word that doesn't exist could the agent manually highlight it and use the text editor toolbar for adding it to the article as well as making it available system wide for future use.

The possibility of adding this incident to the knowledge bank is much clearer with this design as there is a text that explicit asks the agent about it with a checkbox. Thereafter can the agent pick an expiration date for the article from a calendar that looks like an ordinary date

picker. In this mockup example must the agent send the article forward to a technical reviewer as the agent doesn't have the rights for immediate publishing.

The linked tags are displayed as label icons in the bottom left sidebar to make sure that it's easy to see which tags that have been linked. These tags aides in the creation of content to the widgets in the right sidebar. The top most one with the title of "KB-article", short for knowledge bank article, searches through the knowledge bank after articles containing the same tag. If a published article can help solving the incident can the agent check that one as an answer to the issue without the need of writing a specific answer. In the middle of the sidebar is a widget called similar incidents. By combining the linked tags, together with a recent time frame, is the purpose to find incidents that may be solved with a knowledge bank article. When pressing the button to create an article is there a list of choices to be made. A frame will appear where the agent can choose what content from what specific incident that is of interest. This includes incident information, proposed answers and comments from other parties. That information will thereafter be exported to a view where a new article is created.

The final widget is a way of linking configuration items (CIs), often physical possessions in the real world, to the incident. These CIs are also tagged which make the combination to an incident with the same tags a simple task.

Tag handler

Figure 3.32 Mockup from the tag handler

To manage the tags is it essential and therefore was a configuration page created where actions such as add, delete, merge and rename are available (see figure 3.32). A list was

proposed where the agent marks which tags to edit. Add indicates creating a new tag, deletes indicates removing a tag completely throughout the system, merge indicates combining two or more tags that means the same or are misspelled to a set definition and rename means what it says.

3.5 Testing

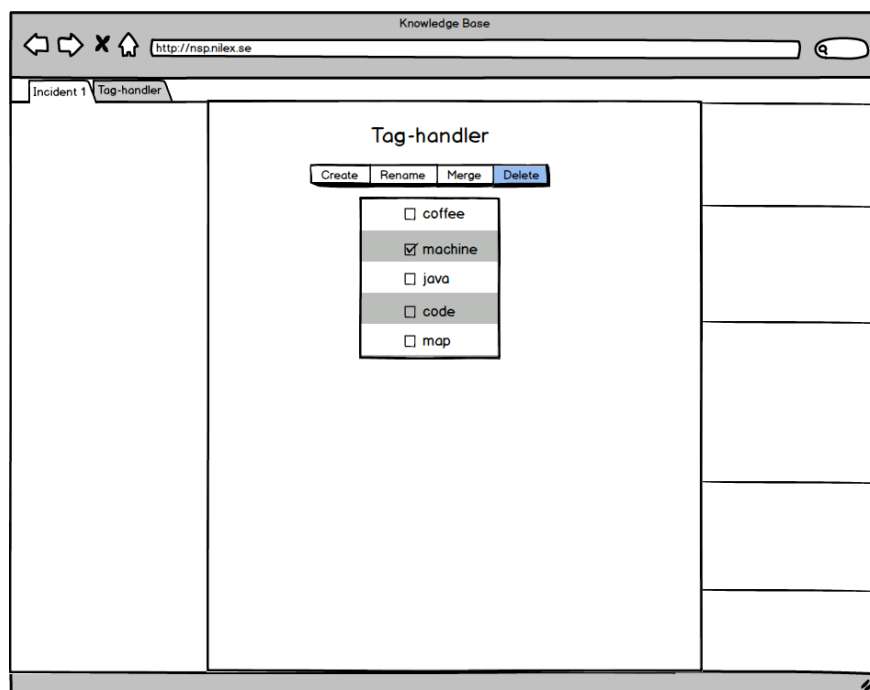The testing was done with the basis of testing the hypothesis. A choice was made to recreate the prototypes of storyboards to interactive prototypes. The reason being that this gives an opportunity to change design before the final implementation. To make the prototypes interactive was the same tool for creating the storyboards "Balsamiq mockups" used with the added functionality of making buttons and other objects clickable. To test the hypothesis was two variants created, a knowledge bank with tagging functionality and one without. Test participants were people with previous experience of the Nilex software and the current knowledge bank but no insight to this project. The reason for that selection was to get more experienced and insightful opinions on the design improvements. The Nilex system is pretty complex which means that doing the test with users with no previous experience of the ticket system would have to learn much more related to the system. This wouldn't have been of interest for this research and very time consuming.

The tests were done in a meeting room environment where a laptop computer was used together with a pointing device of a mouse. In the testing room were two persons, one as the role of a test operator and one tester. A test session started with the test operator explaining the research and what test would be about. The participants were explained to that it was the design that was to be tested and not themselves and that they had to opportunity to leave the test session at any point if they wanted to, all to follow good testing practices. They were also encouraged to ask questions and think aloud if something was unclear or had other opinions about the design. In the first part of the test was the mission to complete some tasks without the tagging functionality. As they progressed through the missions noted the test operator opinions and problems that occurred during their run. Thereafter were the same missions executed but this time with tagging functionality. The functionality of creating tags from an agent perspective was also tested here to see if the proposed method was intuitive. Other proposals to the knowledge bank that wasn't related to tags were also tested for additional possible KB improvements.

After the tests were done did each tester complete a survey about their experience from the test. They had to choose between if they preferred a knowledge bank with or without tags and if they could see it being used more frequently with such functionality. The answers were between 1-5 on a scale or a choice of two opinions. The last question was a curiosity if there was any other opinion or thoughts about the knowledge bank which could be answered by text. The answers to that question was then combined with the notes written by the test operator during the test as several testers forgot or had no more opinion to share.

3.5.1 Result of the test

The pilot test resulted in that two scenarios switched places and that a question in the survey was skipped. Otherwise seemed liked everything was well understood so the real tests could be undergoing. Each participants summary from "thinking aloud" is presented down below.

1.
- The interface is very "grey:ish"
- Want to be able to change language.
- Doesn't see too much difference with or without tags.
- From the agents perspective is marking a word as a tag from the toolbar unclear. The icon must be highlighted in some way or otherwise will they never find that option. Right-click would be a better choice.
- Thinks that the "similiar incidents" widget will frighten some agents and isn't inviting to experimentation with it. Other agents will "spam" create articles because they don't understand how to use it.
- Additions to the "similiar incident" widget popup to have add all and remove all.
- The tag settings should have a search function.

2.
- Some uncertainties in the answer from "helpful or not" button as it isn't a perfect match.
- An agent shouldn't have the rights to choose if there should be comments or not.

3.
- Answers for the "helpful or not" button aren't matching what the tester wants to report.
- Wants to be able to add more tags in the search field together with free text.
- The label icon for tags can be misinterpreted as an arrow pointing in a hierarchy system. Can be made clearer.
- From the agents perspective to be able to remove tags that have been linked.
- Add tags directly from linked tags.
- Tag handler should be put within a submenu for the global settings.
- Need to be able to search within tag handler.
- Delete a tag is bit scary since it's uncertain how the whole system will handle it.

4.
- Misses toolbar button for adding tags. Wants to right-click to add or search within the toolbar.
- Tags should be able to handle translations.

5.
- Misses toolbar button for adding tags.

6.
- Feels it's redundancy with both report-button and "helpful or not"-button. Just wants to be able to write freely from the "helpful or not" answers.
- Tags feels a bit "messy", especially in the result list. Wants to be able to show/hide them. But it's okay to show the tags inside an article.
- Many search functionalities from the previous knowledge bank is missing.
- Wants to be able to right-click a marked word for linking a tag to an article.
- Need to be able to search tags within tag handler

General observations
- Many testers draws no attention immediately to the widget sidebar to solve the first mission. First after navigating back and forward for a period is it noted.
- Many people wants to write a comment before using any of the report functionalities.

Post-test comments
- Functionality of automatic tagging by summarize written words and from that choose potential tags.
- KB should be handled by a specific part and not by an agent. It is too much responsibility for the agent.
- It's much about laziness using tags. Once the first hurdle is handled it is pretty much straight forward.

The post-test questions were choices between one to five if the tester agreed with the allegation.

Hur intuitiv upplevdes kunskapsbanken?

1	0	0 %
2	0	0 %
3	2	25 %
4	5	63 %
5	1	13 %

Figure 3.33 One indicated low intuitiveness and five indicated high intuitiveness

Hur upplevdes widgetsen?

1	0	0 %
2	0	0 %
3	3	38 %
4	1	13 %
5	4	50 %

Figure 3.34 One indicated very negative and five indicated very positive

Vad är din allmänna åsikt om taggar?

1	0	0 %
2	0	0 %
3	0	0 %
4	2	25 %
5	6	75 %

Figure 3.35 One indicated very negative and five indicated very positive

Hur funkar kombinationen med taggar och kunskapsbanken?

1	0	0 %
2	0	0 %
3	0	0 %
4	3	38 %
5	5	63 %

Figure 3.36 One indicated very negative and five indicated very positive

Taggar är en funktionalitet för att öka användandet av kunskapsbanken.

1	0	0 %
2	0	0 %
3	1	13 %
4	2	25 %
5	5	63 %

Figure 3.37 One indicated "I disagree very much" and five indicated "I agree very much"

Taggar underlättar att få relevant information vid sökning.

1	0	0 %
2	0	0 %
3	0	0 %
4	3	38 %
5	5	63 %

Figure 3.38 One indicated "I disagree very much" and five indicated "I agree very much"

Vilken variant med kunskapsbank föredrog du?

— Utan taggar [0]
— Ingen skillna [1]
Med taggar [7] —

Med taggar	7	88 %
Utan taggar	0	0 %
Ingen skillnad	1	13 %
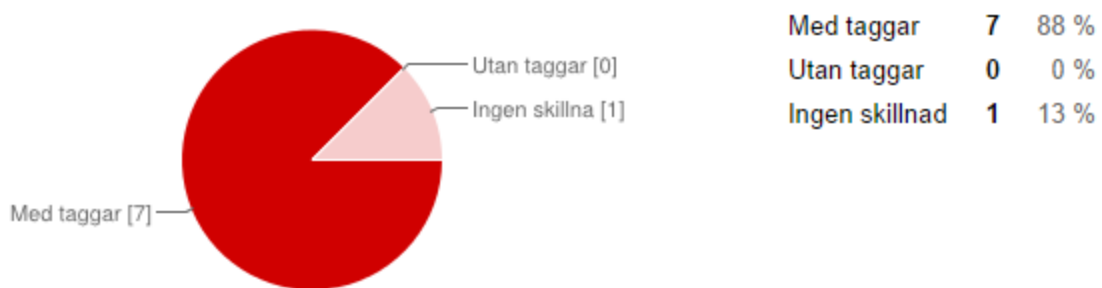
Figure 3.39 What kind of knowledge bank the testers preferred

3.5.2 Discussion

According to these test results seem tags as a concept works very well for a knowledge bank. The majority of the testers could see the benefits of using it and thought that it would improve the usage of the knowledge bank. The widgets gave a mixed reception which

indicates that more work is necessary. One of the most important aspects is that it's trivial to include tags when creating articles or otherwise they won't be used. It was hard to find the icon for adding a new word as a tag from the toolbar. Several users instead wanted to be able to right click a word and add it from a popup menu. The reason for this proposed solution is probably due to the habit of using the old Nilex system. To be noted is that the toolbar will be a component in the new system that commonly will be used for text input. Therefore would it be advantageous if tag-adding could be included there as well. Worth to be noted is that these design proposals were built upon objects that are available in the program for making mockups. A visual artist would be able to create objects that would represents the looks of a final solution better. In any case must the visibility problem be taken seriously before the future implementation phase.

4 Future Work

The management

Both the theory and results from performed methods suggests that the organization aspect of setting up and manage a knowledge bank is vital. This study did not perform any particular amount of work into it because of time and scope constraints. Figuring out what roles, documentation and other practises that need to be upheld are some of the future important work to be done. Most likely will it depend on the type and size of the organization to specify what investments are reasonable.

Virtual Agent

Even if the idea with virtual agents was crossed out is there still potential for further investigation in this area. Users with small technical knowledge may embrace the knowledge bank easier if the interaction resemblance an ordinary conversation. That will probably require the implementation of the virtual agent to have a good visual representation and artificial intelligence to be seen as natural component and not interfering. You don't want the implementation to end up like the infamous type of virtual agent that was Microsoft Word's Clippy. Its way of continually asking the users if they needed help with writing the document was seen as very irritating [38].

Implementation
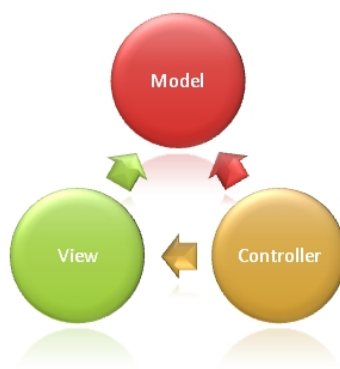
Figure 4.1 The MVC Programming model

A future step is also to implement the proposed knowledge bank functionality with tagging to Nilex latest software, the NSP, built on the Microsoft framework of ASP.NET MVC. This framework's purpose is to create advanced web pages and other web applications. MVC

(model view controller) is a programming model where three processes symbolize the system. The model in MVC represents the logic for different types of data collections in the application. For example are calls to store and get data from a database done here. The purpose of the controller is to handle user interaction by, for example, take the user to a correct page after pushing a button. The view is what is represented on the screen and is often taken from a model. With this triangular separation are complex problems an easier task to handle. Some early backend implementations have been done but this process will continue in due time.

5 Final Discussion

5.1 Widgets

Widget is an interesting feature for presentation of information. Its role may not be essential for performing tasks but instead to be seen more as an aid. People that tested the design proposals had a mixed experience of the widgets. As they were relegated to the side bar did several testers at first miss their existence both in the incident view and the knowledge bank view. The widgets aren't in the main focus of the interface so first after a minute or so were they acknowledge. There are many possibilities for different types of widgets except of the ones proposed; only the imagination is the limit. As for the ones in the knowledge bank, "popular articles" and "did you know", was "popular articles" the one that received most appreciation. A further spin off could be adding "popular tags" in the style of "tag clouds", where the most popular gets the biggest space and font size to indicate it.

The idea behind "similar incident" in the agents perspective seems feasible but the execution of it was one of the more troublesome areas. Conclusion to be drawn is that the agent must be able to choose what content to be included so the resulting article isn't just a document filled with unstructured information. At the same time must the widget be easy-to-use so even the most frighten agent dares to use it. This trade-off is somewhat evened out by making all the options appear after interaction.

5.2 Tags

At first were the tags only supposed to be a search feature for knowledge bank articles but its usage area was broaden after discussion with Nilex employees. Conclusion was that tags would benefit other entities in the system, such as tickets and physical inventorial items, which required higher demands on the functionality. Many web implementations of tags are built in a social system which often means that the end user controls and define the tags. Meanings can then come adrift which makes it hard to administrate. Another potential problem with social tags, noted in the research about "last.fm" [11], is that popular things are better defined by tags in comparison to unpopular things. In the worst case could this mean that articles concerning uncommon problems are harder to find as their tags are vague. Nilex's knowledge bank won't have a social tag system and therefore requires the person in charge of managing the tags to understand its end users thought process. That can be achieved by performing user researches so some of the benefits of social tags can be incorporated to a closed system.

6 Conclusion

The conclusions that can be drawn from the testing results in regards to the hypotheses:

- A knowledge bank with a tagging feature is preferable against a knowledge bank without a tagging feature.

Most of the testing participants were pleased with the tagging feature and preferred it in comparison to a knowledge bank without it. There were several design choices that may needs to be looked further into before the final implementation but the primary idea with tags seems reasonable. The biggest problem being that the tags takes too much visual space so the interface feels cluttered.

- A tagging feature can improve the usage of a knowledge bank.

If the users can with ease find the information they are looking for is the reason for the knowledge bank fulfilled. That seems to be the case when the articles are tagged as it helps sorting out what is a useful article.

- A tagging feature can help a service desk agent understand when it's time to create knowledge bank article.

It's simple to sort out tickets that may relate to each other If they share the same tags. Widgets were created with this is mind to make the agent notice that a knowledge bank article may ease the burden on the helpdesk.

References

[1]Nilex, "About us", [Online], URL:http://www.nilexcorp.com/about-us [Accessed 13/08/2014]

[2]Cartlidge, Alison(2012):"An Introductory Overview
of ITIL® 2011", URL:
https://www.best-management-practice.com/gempdf/itSMF_An_Introductory_Overview_of_ITIL_V3.pdf [Accessed 12/08/2014]

[3] Arraj, Valerie (2013),"ITIL®: the basics",
URL:http://www.best-management-practice.com/gempdf/ITIL_The_Basics.pdf [Accessed 12/08/2014]

[4] APC Group, "ITIL® V3 Foundation Bridging Certificate Course", [ONLINE], URL:
http://www.alc-group.com/images/ul/itil_v3_cms.gif [Accessed 21/09/2014]

[5] Alavi, Maryam, and Dorothy E. Leidner. "Review: Knowledge management and knowledge management systems: Conceptual foundations and research issues."page 35, MIS quarterly (2001),URL:https://flora.insead.edu/fichiersti_wp/inseadwp1999/99-34.pdf [Accessed 12/08/2014]

[6] Koenig, E.D Michael (2012):"What is KM? Knowledge Management Explained",
URL:http://www.kmworld.com/Articles/Editorial/What-Is-.../What-is-KM-Knowledge-Management-Explained-82405.aspx [Accessed 12/08/2014]

[7] Durham, Mary. (2004). Three Critical Roles for Knowledge Management Workspaces. In M.E.D. Koenig & T. K. Srikantaiah (Eds.), "Knowledge Management: Lessons Learned: What Works and What Doesn't." (pp. 23-36). Medford NJ: Information Today, for The American Society for Information Science and Technology.

[8] Rouse, Margaret. "Definition knowledge base." SearchCRM(2007),
URL:http://searchcrm.techtarget.com/definition/knowledge-base [Accessed 12/08/2014]

[9] Allen, Aprill, "A simple guide to creating a knowledge base." Dev Logic Pty(2010), [Ebook]

[10] Golder, Scott A., and Bernardo A. Huberman. "Usage patterns of collaborative tagging systems." Journal of information science 32.2 (2006): 198-208.,
URL:http://arxiv.org/ftp/cs/papers/0508/0508082.pdf [Accessed 13/08/2014]

[11] Lamere, Paul. "Social tagging and music information retrieval." Journal of New Music Research 37.2
(2008):101-114,:http://www.vigliensoni.com/McGill/CURSOS/2009_09/MUMT621/Final%20Project/References/lamere_social_tagging.pdf [Accessed 12/08/2014]

[12] Wikipedia, [Online], URL:http://www.wikipedia.org/ [Accessed 13/08/2014]

[13] Voss, Jakob. "Collaborative thesaurus tagging the Wikipedia way." arXiv preprint cs/0604036 (2006). URL:http://arxiv.org/ftp/cs/papers/0604/0604036.pdf [Accessed 12/08/2014]

[14] Tognazzini, Bruce, "First Principles of Interaction Design (Revised & Expanded)" (2014), URL:http://asktog.com/atc/principles-of-interaction-design/ [Accessed 13/08/2014]

[15]Interaction Design Assoication, "Definition of IxD", [Online], URL:http://www.ixda.org/about/ixda-mission [Accessed 13/08/2014]

[16]Norman, Donald A. The design of everyday things. Basic books, 2002.

[17] Karpoff,Jonathan, "GDD – Goal-Driven Design", [Online], URL:http://jonkarpoff.com/gdd-goal-driven-design/ [Accessed 20/08/2014]

[18] Maier, Andrew, "Complete Beginner's Guide to Interaction Design" (2009), URL:http://www.uxbooth.com/articles/complete-beginners-guide-to-interaction-design [Accessed 13/08/2014]

[19]Baskerville, Philip,"informal meeting" (2013), URL:http://www.skillmaker.edu.au/2013/08/06/informal-meeting/ [Accessed 12/08/2014]

[20] Reyes, Ayra Christine, "Formal and Informal Business Meetings" (2011), URL:http://prezi.com/xopvw66lqblg/formal-and-informal-business-meetings/[Accessed 12/08/2014

[21] Cavanagh,Chris, "Meetings " (2004), URL:http://studentweb.usq.edu.au/home/W0013448/Meeting%20Administration/Meeting_Notes.htm [Accessed 12/08/2014]

[22] Lauesen, Soren. Software requirements: styles and techniques. Pearson Education, 2002

[23] Usability.gov,"Usability Testing", [Online], URL:http://www.usability.gov/how-to-and-tools/methods/usability-testing.html [Accessed 13/08/2014]

[24] Enerson, Meryl, "Paired Comparison Usability Testing" (2012), URL:http://www.softwaremag.com/content/ContentCT.asp?P=3418 [Accessed 13/08/2014]

[25] Nielsen,Jakob, "Thinking Aloud: The #1 Usability Tool" (2012), URL:http://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/ [Accessed 13/08/2014]

[26] O'Brien, Rory, "An Overview of the Methodological Approach of Action Research" (1998), URL:http://www.web.ca/robrien/papers/arfinal.html [Accessed 12/08/2014]

[27] Comaround,"Zero Level support", [Online], URL:http://www.comaround.se/konceptet/zero-level-support [Accessed 12/08/2014]

[28] RightAnswers, "WEB SELF-SERVICE", [Online],URL:http://www.rightanswers.com/products/customer-experience/web-self-service/ [Accessed 12/08/2014]

[29] Zendesk, [Online],URL: https://www.zendesk.com/product/tour [Accessed 12/08/2014]

[30] FrontRange, "HEAT Self Service Quick Tip" (2013), [Video], URL:http://www.youtube.com/watch?v=wmHHwleHNNo#t=107 [Accessed 12/08/2014]

[31] Google, "Google Schoolar", [Online], URL:http://scholar.google.se/ [Accessed 13/08/2014]

[32]LU university, "LUBsearch", [Online], URL:http://www.lub.lu.se/soeka/lubsearch.html [Accessed 13/08/2014]

[33] Mori, Masahiro, Karl F. MacDorman, and Norri Kageki. "The uncanny valley [from the field]." Robotics & Automation Magazine, IEEE 19.2 (2012): 98-100.

[34]Shneiderman, Shneiderman Ben, and Catherine Plaisant. "Designing the user interface 5th edition." (2009).

[35]Usability.gov, "Prototyping", [Online], URL:http://www.usability.gov/how-to-and-tools/methods/prototyping.html [Accessed 13/08/2014]

[36] Balsamiq, "Balsamiq mockups", [Online], URL:http://balsamiq.com/products/mockups/ [Accessed 13/08/2014]

[37] Sharon, Tomer, "Storyboard", [Online] URL:http://www.usabilitybok.org/storyboard [Accessed 12/08/2014]

[38] Gentilviso, Chris, "The 50 Worst Inventions" (2010), URL:http://content.time.com/time/specials/packages/article/0,28804,1991915_1991909_1991755,00.html [Accessed 13/08/2014]

Appendix A

Questionnaire

The survey was created as the following image

KnowledgeBank

*Obligatorisk

Where is your organisation located? *
Specify country

What is the size of your organisation? *
How many employees.

How often do you use a knowledge bank? *
○ Never.
○ Few times each year.
○ Few times each month.
○ Few times each week.
○ Few times each day.

How often do you rate an article in a knowledge bank? *
○ Never.
○ Few times each year.
○ Few times each month.
○ Few times each week.
○ Few times each day.

Do you primarly create knowledge bank articles from user-submitted incidents or from scratch? *
○ I don't create articles.
○ User-submitted incident.
○ From scratch.

If you have a common problem or question, do you foremost turn to a written source or a knowledgeable person for answer? *

◯ Written source.

◯ Knowledgeable person.

If you have a more technically complicated problem or question, do you foremost turn to a written source or a knowledgeable person for answer? *

◯ Written source.

◯ Knowledgeable person.

What do you think would make your company and/or your customers to use a knowledge bank more frequently?

Answer in English or Swedish is fine.

```


```

Skicka

Skicka aldrig lösenord med Google Formulär

Appendix B

Requirements

System

A1. The systems main purpose is to collect knowledge articles and present them for the user by searching or direct links.

A2. The systems functionality and layout should be acceptable by either a mobile device or a computer screen.

A3. There should be different user roles with different rights in the system.

Search

A4. The knowledge bank shall have a input search field for tags or other search terms.

A5. While typing in the search field a list should appear, like auto-complete, with tags that corresponds to the typed input combination.

A6. The result from searching should generate a list with relevant articles based foremost on tag and thereafter rating and last time edited.

A6.1 Those articles in the search result which exist in the internal knowledge bank are presented in a folder structure.

A6.2 Every article is summarized in the result list by title, rating, tags, time for creation as well as time for last edit.

A7. After relevant articles from the internal article database, relevant external sources are presented.

A7.1 The external sources should be presented in such a way that they can be differentiated from the articles in the internal database.

A7.2 Possibility to specify what external sources, e.g. Microsoft TechNet, ComAround, RightAnswers etc, to be included in the search result by check boxes.

A7.3 An external source is displayed in the same way as an internal article to use the same rating system, tags and so on.

A8. If the system shall handle special search inputs, an explanation how to use them should be available.

A9. If you hover the mouse cursor over an article, a small popup should appear with a summary of the article.

Article

A10. Every article shall have a title and a body text.

A11. You should be able to rate an article.

A12. Articles should be able to have tags that are keywords for what the article is about.

A13. You should be able to report an article if something needs to be updated.

A13.1 The article writer shall receive a notification about the event.

A14. You should be able to see total views on an article.

A15. You should be able to write a comment for an article.

A16. You should be able to rate a comment for an article.

A17. You should be able to see in a tree or folder-structure where you are located on the site.

A18. A user rating of 3 or less should result in a popup window with a questionnaire of the reason why the rating was low.

A19. Link at the bottom of the page to other articles with the same tag(s) ranked by rating.

A20. If the article was created through an incident the incident should be linked from the article.

Statistics

A21. For each event a user does as create, edit, rate and so on, save that event for the user.

A22. The event statistics for a prominent user should be displayed in the knowledge bank summary as achievements or interesting facts.

Agent and article creator perspective

A23. An agent should be able to create a knowledge bank article from within a incident report.

A24. The article creator should be able to use scripts when creating an article.

A24.1 The article creator should be able to embed youtube-clips with specific time frames.

A24.2 The article creator should be able to display addresses as points on a map by linking to an external map service.

A25. The article creator should be able to specify what tags should be linked with the article.

A25.1 When writing a word that is in the list of existing tags, the article creator should have the opportunity to include that tag in the article.

A25.2 When highlighting a word in the article the creator should be able to add that word to the list of current tags.

A26. The article creator should be able to specify what user role should be able to see the article.

A27. If the permission allows it, an article should be able to be shared through email, twitter and facebook.

A28. In the sidebar different widgets will be displayed with smart connections to the knowledge bank.

A28.1 The incidents that are suggested as "similar incidents" in the side-widget are retrieved by automatic-searching tags together with similar time-periods.

A28.2 By searching in the widget-sidebar knowledge bank articles containing the same search terms are displayed by primarily user-ranking and thereafter time period of creation.

A28.3 One widget shall list configuration items that shares the same category to be able to link that item to the incident.

A29. Depending on the rights of the user creating the article it may have to be sent to be reviewed by a person with technical knowledge.

A29.1 The reviewer receives a notification by email and will continuously receive reminders until article has been dealt with.

A30. There should be an option to determine when the article is obsolete.

A31. You should be able to manage the tags in a options menu with all the common edit commands.

Appendix C

Survey from the test

KnowledgeBank Test

Beskrivning av formulär

Ålder*

[dropdown ▼]

Kön*

○ Man
○ Kvinna

Hur intuitiv upplevdes kunskapsbanken?*
Betydelsen av knappar, ikoner, symboler osv

	1	2	3	4	5	
Låg intuitivitet	○	○	○	○	○	Hög intuitivitet

Hur upplevdes widgetsen? *

	1	2	3	4	5	
Väldigt negativt	○	○	○	○	○	Väldigt positivt

Vad är din allmänna åsikt om taggar?*

	1	2	3	4	5	
Mycket Dåligt	○	○	○	○	○	Mycket Bra

Hur funkar kombinationen med taggar och kunskapsbanken?*

	1	2	3	4	5	
Mycket dåligt	○	○	○	○	○	Mycket bra

Taggar är en funktionalitet för att öka användandet av kunskapsbanken.[*]

       1   2   3   4   5

Håller absolut inte med. ○ ○ ○ ○ ○ Håller absolut med.

Taggar underlättar att få relevant information vid sökning.[*]

       1   2   3   4   5

Håller absolut inte med. ○ ○ ○ ○ ○ Håller absolut med.

Vilken variant med kunskapsbank föredrog du?[*]
○ Med taggar
○ Utan taggar
○ Ingen skillnad

Andra kommentarer om kunskapsbanken/taggar: